                  Media Gateway Control Protocol Fax Package

Status of This Memo

   This memo provides information for the Internet community. It does
   not specify an Internet standard of any kind. Distribution of this
   memo is unlimited.

Abstract

   This document defines a Media Gateway Control Protocol (MGCP) package
   to support fax calls. The package allows for fax calls to be
   supported in two different ways. The first one utilizes ITU-T
   Recommendation T.38 for fax relay under the control of the Call
   Agent. The second one lets the gateway decide upon a method for fax
   transmission as well as handle the details of the fax call without
   Call Agent involvement.

Table of Contents

1. Introduction

   This document defines a Media Gateway Control Protocol (MGCP)
   [RFC3435] package that enables MGCP controlled gateways to support
   fax calls. The package enables fax calls to be supported in two
   different ways. The first one utilizes ITU-T Recommendation T.38
   using either UDP Transport Layer (UDPTL) or TCP (see [T38]) for fax
   relay under the control of the Call Agent. The second one lets the
   gateway decide upon a method for fax transmission as well as handle
   the details of the fax call without Call Agent involvement.

   The fax package definition is provided in Section 2, and in Section 3
   we provide three call flow examples showing how to use it. Security
   considerations are found in Section 4, followed by the IANA
   considerations and references.

1.1. Conventions Used in This Document

   The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT",
   "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this
   document are to be interpreted as described in BCP 14, RFC-2119
   [RFC2119].

2. Fax Package Definition

   A package is defined for fax. The package defines new
   LocalConnectionOptions, events, and connection parameters as detailed
   below:

   Package Name: FXR
   Package Version: 0

2.1. LocalConnectionOptions

   A new Fax LocalConnectionOptions (LCO) parameter is defined for fax
   handling. The Call Agent supplies this fax LCO to indicate the
   desired fax handling procedure to the Media Gateway. The fax LCO
   contains a list of desired fax handling procedures ordered by
   preference, with the most desired procedure listed first. When the
   parameter is explicitly included in a command, the gateway MUST be
   able to use at least one of the listed procedures for the command to
   succeed. Currently, the list can indicate one or more of the
   following procedures (see Sections 2.1.1 to 2.1.4 for further details
   on these):

   * T.38 Strict:
     Use T.38 [T38] with either UDPTL or TCP for fax relay and have the
     Call Agent control it. Assuming the procedure can be used (see
     Section 2.1.1), a switch to T.38 procedures will be initiated upon
     fax detection, and a "t38(start)" event will be generated (see
     Section 2.2). This mode requires an indication of T.38 support
     from the remote side in order to be used, as described further in
     Section 2.1.1.

   * T.38 Loose:
     Identical to T.38 Strict procedure, except that an indication of
     T.38 support from the remote side is not required for the procedure
     to be used.

   * Off:
     Do not invoke any special procedure for fax, except for echo
     cancellation adjustment and possibly switching to another codec.

   * Gateway:
     Let the gateway control and decide how to handle fax calls without
     Call Agent involvement. This includes the case where the gateway
     does not do anything special for fax; hence, by definition this
     procedure can always be supported. If the gateway invokes a
     special procedure upon detection of fax, it will generate a
     "gwfax(start)" event to inform the Call Agent of this (see Section
     2.2). The Call Agent SHOULD then refrain from issuing potentially
     conflicting commands to the gateway until the gateway ends its
     special fax handling procedure.

     A gateway that ends up not being able to invoke any special
     procedure for fax will generate a "nopfax(start)" event (see
     Section 2.2) upon detection of fax.

   The set of possible values (i.e., procedures) for the fax LCO is
   extensible. The prefix "x-", which indicates an optional extension,
   and the prefix "x+", which indicates a mandatory extension, are
   reserved for vendor-specific use.

   In CreateConnection commands, the fax LCO value defaults to
   "gateway". In ModifyConnection commands, the fax LCO value defaults
   to its current value on the connection. Thus, if
   LocalConnectionOptions are omitted or if the fax LCO is not included
   in a ModifyConnection command, the previous fax LCO value for the
   connection is retained without affecting the outcome of the command;
   consequently, the gateway may now not apply any special procedure to
   fax. If the Call Agent wants to ensure that a command succeeds only
   when a fax procedure is applied, the command needs to include the fax
   LCO explicitly.

      As an example of this, assume that the CreateConnection command
      successfully specified the use of "T.38 Strict", and a
      ModifyConnection command is now received without the fax LCO but
      with a RemoteConnectionDescriptor indicating no support for T.38.
      In this case, the ModifyConnection command will succeed, but T.38
      procedures will no longer be invoked upon fax detection (a
      "nopfax" event will be generated). Had the Call Agent instead
      included the fax LCO set to "T.38 Strict", the command would have
      failed.

   If multiple fax parameter values are provided, the gateway MUST
   choose one of the procedures specified according to the order in
   which they are supplied, except as follows:

   1. If "gateway" would have been selected and it would have resulted
      in no special procedure being applied, and

   2. if there are procedures other than "off" that are specified after
      "gateway" (e.g., "t38"),

   then the gateway MUST use the most preferred of those subsequent
   procedures that can be supported. If none of those subsequent
   procedures can be supported, the gateway reverts to not invoking any
   special procedure for fax. Please refer to Section 2.1.4 for further
   details on determining which procedures can be supported.

   The fax LCO parameter is encoded as the keyword "fx" (prefixed with
   the package name per [RFC3435]), followed by a colon and then a
   semicolon separated list of values, where T.38 Strict is encoded as
   "t38", T.38 Loose is encoded as "t38-loose", gateway is encoded as
   "gw", and off is encoded as "off".

   The following example illustrates the use of PCMU or G.729 for audio
   encoding, and T.38 Strict fax relay (preferred) or gateway control
   for fax:

      L: a:PCMU;G729, fxr/fx:t38;gw

   It should be noted that MGCP allows the CreateConnection command to
   omit both LocalConnectionOptions and RemoteConnectionDescriptor,
   thereby letting the gateway decide upon the media parameters to use.
   When the T.38 fax package is supported, the gateway could thus choose
   to do either audio or T.38 fax relay in such cases. Most likely, the
   Call Agent requires one or the other to be used, and hence it SHOULD
   NOT omit both LocalConnectionOptions and RemoteConnectionDescriptor
   in CreateConnection commands.

   When auditing capabilities, the fax LCO may be returned with a
   semicolon-separated list of supported fax handling parameters. The
   values "t38", "t38-loose", "off", and "gw" MAY be omitted from such a
   list as they are always implied. Gateways that implement additional
   parameters SHOULD return these additional parameters when
   capabilities are audited, as illustrated by the following example:

      A: a:image/t38, fxr/fx:mypar, ...

   In the following subsections, we provide additional detail on the
   above-defined fax procedures.

2.1.1. T.38 Procedure (Strict or Loose)

   When a gateway is instructed to use one of the T.38 procedures
   (strict or loose), also known as Call Agent controlled T.38 mode, the
   "m=" line in the Session Description Protocol (SDP) returned will not
   indicate use of UDPTL-based or TCP-based T.38 (unless the gateway was
   also instructed to use "image/t38" for the media stream). Any other
   entity seeing this SDP will not know whether or not T.38 is supported
   and hence whether it is safe to attempt a switch to T.38 upon fax
   detection. To remedy this dilemma, capability information for T.38
   (if supported) using the SDP Simple Capability Declaration extensions
   [RFC3407] MUST be included. Other capability information is included
   as well, regardless of whether the Call Agent authorized use of those
   in the connection handling command. A subsequent attempt to actually
   use these may of course not succeed, e.g., because the Call Agent LCO
   does not allow them to be used. The following example illustrates
   the RFC 3407 [RFC3407] capability descriptor--note the inclusion of
   both current (audio) and latent (T.38) capabilities, as specified in
   RFC 3407 [RFC3407]:

       m=audio 3456 RTP/AVP 18
       a=sqn: 0
       a=cdsc: 1 audio RTP/AVP 18
       a=cdsc: 2 image udptl t38

   For a list of T.38 related parameters to be included in the SDP,
   please refer to T.38 Annex D [T38].

   Upon fax detection, a gateway that has successfully been instructed
   to use one of the T.38 procedures will:

   1. Initiate the T.38 fax relay procedure and mute the media channel
      in both the send and receive direction (unless the media channel
      is already using T.38).

   2. Generate a "t38(start)" event.

   3. Await further instructions from the Call Agent in order to
      initiate the actual media change (unless the media channel is
      already using T.38).

   The Call Agent instructs the gateway to perform the media change by
   sending it a ModifyConnection command with "image/t38" listed as the
   encoding method in the LocalConnectionOptions (receipt of a
   ModifyConnection command without LocalConnectionOptions but with a
   RemoteConnectionDescriptor containing an "m=" line with the MIME type
   "image/t38" would achieve the same). Per the normal MGCP codec
   negotiation procedures (see [RFC3435] Section 2.6), if a

RemoteConnectionDescriptor was included as well, it needs to include
an "m=" line with "image/t38" as an acceptable media format in order
for the command to succeed. The gateway may choose between the UDPTL
and TCP transport protocols at its own discretion subject to the
normal MGCP codec negotiation procedures (in practice, TCP-based
implementations are currently rare).

If a RemoteConnectionDescriptor was not included with the
ModifyConnection command sent to a gateway that initiated the T.38
procedure, it is possible (in fact likely), that the last received
RemoteConnectionDescriptor did not include an "m=" line listing
"image/t38" as an acceptable media format. In that case, the
endpoint cannot send T.38 media to the other side. The endpoint MUST
instead wait for an updated RemoteConnectionDescriptor that contains
"image/t38" as an acceptable media format and a supported transport
protocol (UDPTL or TCP). The T.38 fax procedure continues when an
acceptable RemoteConnectionDescriptor is received. An acceptable
RemoteConnectionDescriptor contains an "m=" line with the "image/t38"
MIME type (using the normal SDP syntax) and a supported transport
protocol (UDPTL or TCP). If the fax call fails (e.g., due to a fax
timeout) while waiting for either the Call Agent to instruct the
gateway to switch to "image/t38" or for an acceptable
RemoteConnectionDescriptor, a "t38(stop)" or a "t38(failure)" event
MUST be generated. When the T.38 procedure ends, a "t38(stop)" or
"t38(failure)" event MUST be generated.

Finally, the Call Agent may need to abort a T.38 procedure that is in
progress. This can for example be done when the remote side is
unable to switch to T.38, and a fallback to fax passthrough using an
audio codec is attempted. The Call Agent instructs the endpoint to
abort an in-progress T.38 procedure by use of the "off" fax LCO as
illustrated below:

    L: fxr/fx:off

We now define "time t38init" as the point in time where the T.38
procedure was initiated, and "time t38abort" as the point in time
where the Call Agent aborts an in-progress T.38 procedure. If the
Call Agent at time t38abort instructs or enables the endpoint to
revert to one or more codecs that were in use just prior to time
t38init, the endpoint SHOULD use media stream parameters that mimic
the most recent LocalConnectionDescriptor issued before time t38init.
For example, IP-address and UDP port, payload formats used and their
payload type mapping, should all be the same as before time t38init.
This will enable the fallback to be as rapid as possible. A
LocalConnectionDescriptor is returned as usual, i.e., only if one or

more parameters changed since the last LocalConnectionDescriptor
issued (e.g., if a T.38 LCD was issued or a transport address in the
audio LCD was changed).

2.1.2. Gateway Procedure

A gateway using the gateway procedure, also known as Gateway
controlled mode, may initiate special fax handling upon detecting a
fax call. The details of this special fax handling are outside the
scope of this document. However, in order to use any special fax
handling, support for it MUST be negotiated with the other side by
passing and recognizing relevant parameters via the
LocalConnectionDescriptor and RemoteConnectionDescriptor (this
includes the use of RTP-based T.38). If the other side has not
indicated support for the special fax handling desired, the gateway
MUST NOT attempt to initiate it. When special fax handling is
initiated, a "gwfax(start)" event MUST be generated, thereby enabling
the Call Agent to differ between the Call Agent and gateway
controlled mode while still being informed about the actual change to
fax. When the special gateway handling of fax ends, a "gwfax(stop)"
or "gwfax(failure)" event MUST be generated.

2.1.3. Off Procedure

A gateway using the "off" procedure will not invoke any special fax
procedures, e.g., T.38, when detecting a fax. However, the gateway
may still adjust local echo cancellation and/or switch to an
alternative codec as needed. Also, a "nopfax(start)" event MUST be
generated; a corresponding "stop" event, however, will not.

Generating a "stop" event would imply that the gateway had to infer
when the fax call ends, which involves processing the media stream.
However, when using the "off" mode, such processing is not expected
to occur.

2.1.4. Mode Operation

For each of the above modes, the RemoteConnectionDescriptor provides
information on what procedure(s) the other side supports. The
following rules are used to determine which procedure to use:

1. Whatever the Call Agent specified in the Fax
   LocalConnectionOptions for the current command MUST be adhered to.
   If the gateway cannot satisfy any of the options, the command
   fails (error code 532 -- unsupported value(s) in
   LocalConnectionOptions is RECOMMENDED).

   2. If both Fax LocalConnectionOptions and a
      RemoteConnectionDescriptor are provided, the procedure selected
      MUST be supported by both sides -- this is currently only an issue
      for "T.38 Strict". A procedure can be satisfied by the remote
      side if:

      * the relevant MIME media type, e.g., "image/t38", is included in
        the "m=" line in the RemoteConnectionDescriptor, or

      * the relevant MIME media type is included as a capability (see
        [RFC3407]) in the RemoteConnectionDescriptor.

      If the gateway cannot select any of the procedures in the Fax
      LocalConnectionOptions, the command fails (error code 532 is
      RECOMMENDED). Note that "T.38 Loose", "gateway", and "off" -- by
      definition -- can always be supported by an implementation that
      supports this package, irrespective of what the
      RemoteConnectionDescriptor indicates.

   3. If the Call Agent did not include any Fax LocalConnectionOptions
      or a RemoteConnectionDescriptor with the command, the gateway MUST
      continue using whichever procedure it is currently using.

   4. If the Call Agent did not include any Fax LocalConnectionOptions,
      but a RemoteConnectionDescriptor was included, the gateway MUST
      follow rule 2 in selecting a procedure. In so doing, the default
      Fax LocalConnectionOptions, i.e., "gateway" in CreateConnection,
      or the current value in ModifyConnection, MUST be used. In the
      case of ModifyConnection, the outcome of the command does not
      depend on the gateway being able to select one of these "default"
      procedures (as described in Section 2.1). Note that this is not
      an issue for the CreateConnection command, since the default value
      can always be supported by definition.

   5. A previously received RemoteConnectionDescriptor does not affect
      what procedure can be selected. Only a RemoteConnectionDescriptor
      supplied with the current command affects the procedure selection.
      However, in order to send media of a given type (e.g.,
      "image/t38"), the most recently received
      RemoteConnectionDescriptor MUST include a corresponding media
      line.

The following examples illustrate the use of the above rules:

Per rule 1, a gateway that only supports standard T.38 fax relay will fail a command that only contains the fax option "mypar", whereas it will succeed a command that contains "t38-loose", "gw", "off", or no fax LCO. A command that only contained "t38", i.e., use of T.38 in "strict" mode, may or may not succeed (depending on the RemoteConnectionDescriptor).

A gateway supporting T.38 that receives a CreateConnection command with the fax handling LCO set to "t38" and a RemoteConnectionDescriptor with neither a T.38 capability nor a T.38 media stream will fail per rule 2. Had the fax handling LCO included either "t38-loose", "gw" or "off", the command would have succeeded, and any of the procedures included could have been selected.

Assume a gateway supporting T.38 has successfully executed a CreateConnection command with fax handling set to "t38" (i.e., strict). If the gateway now receives a ModifyConnection command without a fax handling LCO but with a RemoteConnectionDescriptor that has neither a T.38 capability nor a media stream with "image/t38", the command will succeed (since rule 1 has no effect in that case). However, per rule 2 and 4, there will not be any T.38 procedure in place. Had the Call Agent instead included a fax handling LCO set to "t38" again, the command would have failed per rule 2.

Finally, it should be noted that a switch to T.38 can be initiated by either one or both of the originating and terminating gateways and hence implementations MUST be prepared to handle this. This includes the case where both sides initiate the switch, which for example can occur when the originating fax generates Calling Tone (CNG) and the terminating fax detects V.21 fax preamble (see [T30]) before the switch to T.38 has been performed on the terminating side.

2.1.5. Detecting a Fax Call

A fax call can be detected by several different means (e.g., V.21 fax preamble, T.30 CNG tone, or V.8 signals) depending on the fax transmission method being used. Implementations of this package MUST at a minimum detect a fax call based on V.21 fax preamble.

Triggering based on T.30 CNG tone MAY be done; this is generally considered acceptable for G3 and lower fax speeds. However, when used with T.38 version 2 or earlier, it will impact V.34 high-speed fax. The reason is that T.38 version 2 (and earlier) does not support the V.8 ANSam and CM signals used with V.34 fax, and hence the V.34 faxes will downspeed to G3 (14.400 bps) or lower when using T.38 version 2 (or earlier). Also, a few rare cases of modems

generating T.30 CNG tones for non-fax calls have been reported; such
modems would generate a false trigger for fax. As a consequence of
the above, it is RECOMMENDED that implementations of this package
that support T.30 CNG-based fax detection provide a configuration
option to disable it for T.38 version 2 (or earlier).

2.1.6. Considerations for Determining Which Procedures to Request

   It is important to understand the implications of using any one of
   the above defined procedures. Furthermore, multiple alternative
   procedures can be requested, however not all combinations make sense.
   In this section, we elaborate on both of these issues.

   Use of the T.38 Strict mode is ideal in an environment where it is
   known that other endpoints generate RFC 3407 [RFC3407] capability
   descriptions with T.38 fax relay information. If a
   RemoteConnectionDescriptor without T.38 fax relay capabilities is
   received in such an environment, it is known that the other side does
   not support T.38, and hence an unsuccessful attempt to switch to T.38
   (which in turn may lead to a failed fax call) can be avoided. If it
   is not known whether other endpoints support the RFC 3407 [RFC3407]
   capability descriptors, the trade-off is less clear. The advantage
   is that a switch to T.38 will only be attempted if it is known that
   the other side supports it, but endpoints that do not indicate
   support for T.38 may still support it; however, T.38 will not be used
   with these, which in turn may lead to unnecessary fax failures with
   low-bandwidth codecs or lossy networks.

   Use of the T.38 loose mode involves the same considerations as for
   T.38 Strict, however the pros and cons are reversed. If a peer
   endpoint does not support T.38, the T.38 loose mode will still
   attempt to switch to T.38 (and fail), which in turn may lead to a
   failed fax call. On the other hand, if the peer endpoint does not
   support the RFC 3407 [RFC3407] capability descriptors, but the peer
   endpoint does in fact support T.38, T.38 would still be used with
   this mode.

   In summary, there is no single good answer to the use of either T.38
   Strict or T.38 loose mode; it depends on the capabilities of the
   endpoints involved as well as the trade-off between potentially
   letting fax calls fail due to lack of capability indications (where
   T.38 is otherwise supported) versus potentially letting fax calls
   fail due to an unsuccessful switch to T.38 (because T.38 in fact was
   not supported). It should be noted that Call Agents may have means
   beyond RFC 3407 [RFC3407] capability descriptors to determine if a
   peer endpoint supports T.38 or not. For example, when SIP is used as
   the signaling protocol with other peers (e.g., Call Agents or other
   SIP devices), the SIP OPTIONS method can be used to learn whether

T.38 is supported. Also, if the Call Agent allows use of
high-bandwidth codecs with redundancy when support for T.38 is not
indicated, fax calls may still succeed without the use of T.38, even
in networks with non-negligible packet loss.

When the gateway controlled mode is selected, there will only be
special fax handling if the two peer endpoints support the same fax
handling method; note that the details of the actual method is
entirely up to the vendor. Also note that if the two peer endpoints
do not support the same method for fax handling or if the method is
not indicated in the SDP exchanged, there will be no special fax
handling in place. Furthermore, the Call Agent will not be aware
that this is the case until the fax transmission starts and a
"nopfax(start)" event is generated.

The off mode is straightforward; there will be no special procedure
in place for fax handling, except for the usual handling of echo
cancellation and possibly a change to a higher bandwidth codec.

Having looked at the individual procedures in more detail, we now
elaborate on some of the combinations of procedures that may be
requested:

* T.38 strict:
  If the T.38 strict procedure is placed after the T.38 loose or the
  off procedure (both of which can always be supported), it will not
  be selected. Apart from this, it makes little sense to request
  both T.38 strict and T.38 loose.

* T.38 loose:
  The T.38 loose procedure can always be supported, so any procedure
  specified after T.38 loose will not be selected.

* Gateway:
  The gateway controlled procedure can always be supported. If the
  gateway controlled procedure would have resulted in no special fax
  procedure and further options (except off) are provided, those
  procedures will be attempted. If neither of those procedures can
  be supported, there will be no special fax procedure in place.

* Off:
  The off procedure can always be supported. Any procedure specified
  after this one will not be selected.

2.2. Events and Signals

   The following events are defined in support of the above:

Symbol	Definition	R	S Duration
gwfax	Gateway controlled fax	x	
nopfax	No special fax handling	x	
t38	T.38 fax relay	x	

   The definitions of the individual events are provided in the
   following subsections.

2.2.1. Gateway Controlled Fax (gwfax)

   The "gateway controlled fax" event occurs when the gateway handled
   fax procedure either starts, stops, or fails. The event is encoded
   as "gwfax", and the following event parameters, which apply to
   ObservedEvents only, are defined:

   * start:
     Gateway controlled fax procedure was initiated. The Call Agent
     SHOULD refrain from issuing media handling instructions to the
     gateway until either a "gwfax(stop)" or "gwfax(failure)" event is
     generated.

   * stop:
     Gateway controlled fax procedure ended and the gateway did not
     detect any errors. Note that this does not necessarily imply a
     successfully transmitted fax. It merely indicates that the gateway
     controlled fax procedure has ended and the procedure itself did not
     encounter any errors. Media parameters for the connection are as
     before the gateway handled fax procedure started.

   * failure:
     The gateway controlled fax procedure ended abnormally. Some kind
     of problem was encountered in the gateway controlled fax procedure,
     and the procedure ended. Media parameters are as before the
     gateway handled fax procedure started.

   One of the above parameters will be present when the event is
   reported. The "gwfax" event MAY be parameterized with additional
   parameters in ObservedEvents, however it is RECOMMENDED that one of
   the above parameters is the first parameter supplied. Unknown
   parameters MUST be ignored.

The following example illustrates the encoding of the "gwfax" event:

```
O: fxr/gwfax(start)
O: fxr/gwfax(stop, foobar)
```

2.2.2. No Special Fax Handling (nopfax)

The "no special fax handling" event occurs when there is no special
fax handling procedure in place and a fax call is detected. This can
happen either because no special fax handling procedure was requested
(including "off") or negotiation resulted in no special fax handling
procedure being supported. The event is encoded as "nopfax", and the
following event parameter, which applies to ObservedEvents only, is
defined:

* start:
  No special fax handling procedure is in place, however a fax call
  is now detected. The Call Agent may have to issue further commands
  in order to ensure a successful fax call (e.g., switch to another
  codec).

The above parameter will be present when the event is reported. The
"nopfax" event MAY be parameterized with additional parameters on
ObservedEvents, however it is RECOMMENDED that the above parameter is
the first parameter supplied. Unknown parameters MUST be ignored.
Note that this event currently cannot be parameterized with "stop" or
"failure" as it only detects the beginning of a fax call.

The following example illustrates the encoding of the "nopfax" event:

```
O: fxr/nopfax(start)
```

2.2.3. T.38 Fax Relay (t38)

The "T.38 fax relay" event occurs when one of the T.38 fax relay
procedures (strict or loose) either starts, stops, or fails. The
event is encoded as "t38", and the following event parameters, which
apply to ObservedEvents only, are defined:

* start:
  A fax call was detected on the endpoint and the Call Agent
  controlled T.38 fax relay procedure was initiated. The Call Agent
  SHOULD modify each side of the connection to start using the
  "image/t38" media format, unless they already do. Note that, as
  long as use of the Call Agent controlled T.38 relay procedure is in
  effect, the event will be generated upon fax call detection,
  irrespective of the current encoding method on any connections on
  the endpoint (incl. "image/t38"). The "t38(start)" event MUST be

generated at most once by the endpoint per fax call, regardless of
whether or not it is requested again in a subsequent requested
events list.

* stop:
  Call Agent controlled T.38 fax relay procedure ended and the
  gateway did not detect any errors. Note that this does not
  necessarily imply a successfully transmitted fax. It merely
  indicates that the Call Agent controlled T.38 fax relay procedure
  has ended and the procedure itself did not encounter any errors.
  The Call Agent may want to modify the media parameters for each
  side of the connection. Note that, in contrast to the gateway
  controlled fax procedure case, media parameters such as codecs do
  not automatically revert to their values before the start of the
  fax call; however, echo cancellation and silence suppression do per
  the procedures in [RFC3435] Section 2.3.5. The "t38(stop)" event
  MUST NOT be generated unless a corresponding "t38(start)" event for
  the fax call in question was generated earlier.

* failure:
  Call Agent controlled T.38 fax relay procedure ended abnormally.
  Some kind of problem in the Call Agent controlled T.38 fax relay
  procedure was encountered, and the procedure ended. The Call Agent
  may want to modify the media parameters for each side of the
  connection. Note that, in contrast to the gateway controlled fax
  procedure case, media parameters such as codecs do not
  automatically revert to their state before the start of the fax
  call; however, echo cancellation and silence suppression do per the
  procedures in [RFC3435] Section 2.3.5. The "t38(failure)" event
  MUST NOT be generated unless a corresponding "t38(start)" event for
  the fax call in question was generated earlier.

One of the above parameters will be present when the event is
reported. The "t38" event MAY be parameterized with additional
parameters, however it is RECOMMENDED that one of the above
parameters is the first parameter supplied. Unknown parameters MUST
be ignored.

The following example illustrates the encoding of the "t38" event:

    O: fxr/t38(start)
    O: fxr/t38(stop, foobar)

2.3. Connection Parameters

The connection parameters for the connection, that measures packets
and octets sent and received, MUST include packets and octets for fax
handling as well. Interarrival jitter and average transmission delay

calculation however MAY NOT be performed while fax is in progress,
e.g., if T.38 is used. In such cases, the interarrival jitter and
average transmission delay calculations are simply suspended until
calculations can resume, e.g., by changing back to an RTP-based media
stream.

In addition to these connection parameters, the fax package defines
the following connection parameters, which gateways MAY support:

Number of fax pages sent (PGS):

   The cumulative number of fax pages sent by the endpoint for the
   life of the connection. The parameter is encoded as "PGS", and
   the value supplied is a string of up to nine decimal digits.

Number of fax pages received (PGR):

   The cumulative number of fax pages received by the endpoint for
   the life of the connection. The parameter is encoded as "PGR",
   and the value supplied is a string of up to nine decimal digits.

The following example illustrates the use of these parameters:

   P: FXR/PGS=3, FXR/PGR=0, PS=1245, OS=62345, ...

2.4. Negotiation of T.38 Parameters

T.38 Annex D [T38] defines a number of T.38 parameters that can be
negotiated in the SDP. Currently, T.38 does not specify procedures
for how each of these parameters is negotiated or in particular
whether each side has to use the same value. Therefore, we
considered adding such definitions and procedures here. However, it
is expected that T.38 will rectify the above, which could lead to
conflicting definitions and procedures. To avoid that, we instead
assume the existence of an offer/answer [RFC3264] section for T.38,
where T.38 Annex D parameters are classified as either declarative or
negotiated, and we then provide guidelines for how to map such
definitions and procedures to the MGCP fax package defined here.

MGCP does not specify use of the offer/answer model but instead
operates with the concept of connection handling commands (e.g.,
CreateConnection and ModifyConnection) that may include a
RemoteConnectionDescriptor (SDP) and in turn may generate a
LocalConnectionDescriptor (SDP) in their response.

When an MGCP endpoint receives a CreateConnection command without a
RemoteConnectionDescriptor, it should follow the corresponding T.38
procedures for generating an initial offer and return the resulting
SDP in its LocalConnectionDescriptor.

When an MGCP endpoint receives a CreateConnection command with a
RemoteConnectionDescriptor, it should follow the corresponding T.38
procedures for receiving an initial offer and generating an answer to
it. The resulting SDP is returned in the LocalConnectionDescriptor.

When an MGCP endpoint receives a ModifyConnection command with a
RemoteConnectionDescriptor, it cannot determine whether this
corresponds to an answer to an initial offer or to a new offer. This
is not an issue for declarative parameters since they can be
specified independently in either direction. Negotiated parameters,
however, require some consideration:

When an offerer receives an answer to a previous offer, the
negotiation has completed and the parameters negotiated can no longer
be changed with this offer/answer exchange. The negotiated
parameters may be subject to certain validation checks. Conversely,
when an answerer receives an offer, the negotiation is open and the
answerer may change some of the offered negotiated parameters. Since
the MGCP endpoint does not know which situation it is in, it cannot
perform the "offerer" validation checks. Likewise, in order to
ensure that any required negotiation actually takes place, it needs
to process an incoming SDP as an offer. If the SDP in fact does
correspond to an offer, then this is obviously correct behavior.
However, if the SDP corresponds to an answer, and one or more
negotiated parameters did change, then this will result in a new SDP.
The Call Agent may or may not contain sufficient intelligence to
determine whether or not this new SDP needs to result in another
offer/answer exchange.

    For example, if the initial offer (in response to a
    CreateConnection without SDP) contained fax version 2, and the
    answer (in response to a CreateConnection with SDP) contained fax
    version 0, then the corresponding ModifyConnection command (with
    SDP) will result in an updated SDP with fax version also set to
    zero. If this was the only change in the updated SDP, a new
    offer/answer exchange would not be needed. Note that this example
    does not imply that it is generally considered a good idea for
    Call Agents to parse SDP in order to determine whether or not new
    offer/answer exchanges are needed.

Finally, a ModifyConnection without SDP that generates an SDP needs
to be considered. The SDP generated may either correspond to an
initial offer/answer exchange or a subsequent offer/answer exchange.

The endpoint should generate SDP as if it was part of a subsequent
offer/answer exchange. If the Call Agent does not desire such
semantics, it can simply create a new connection instead.

2.5. Implementation Considerations

2.5.1. Media IP Address and Port for T.38

When an endpoint is instructed to change to or from T.38 for a media
stream, it SHOULD continue using the same IP address and port the
media stream is currently using, since this will minimize any Quality
of Service, Network Address Translator (NAT), and Firewall
interactions from the change. However, if an endpoint has a good
reason, it MAY choose not to follow this recommendation.

When an endpoint uses the same port for RTP audio and T.38 with
either UDPTL or TCP, packets of one type (e.g., T.38) may be received
while expecting packets of another type (RTP audio). Since there is
explicit signaling to indicate which type is expected at any given
point in time, this does not introduce any new problems. In other
words, the receiver does not operate as a demultiplexer with a need
to determine if a given packet received is an RTP audio packet or a
T.38 UDPTL/TCP packet. The receiver simply processes incoming
packets as usual. If T.38 packets are expected, then incoming
packets are validated against T.38, and if RTP audio packets are
expected, then incoming packets are validated against RTP.

2.5.2. Case Sensitivity

IANA has registered the uppercase string "UDPTL" as the transport
protocol identifier to be used for UDP-based T.38. However, the
examples provided in Recommendation T.38, as well as most (if not
all) current implementations, use the lowercase string "udptl"
instead. Implementations conforming to this package SHOULD generate
the lowercase string "udptl" and accept the lowercase, uppercase, and
mixed upper/lowercase strings as being equivalent.

The attribute "T38MaxBitRate" was once incorrectly registered with
IANA as "T38maxBitRate" (lower-case "m"). In accordance with T.38
examples and common implementation practice, the form "T38MaxBitRate"
SHOULD be generated by implementations conforming to this package.

In general, it is RECOMMENDED that implementations of this package
accept lowercase, uppercase, and mixed upper/lowercase encodings of
all the T.38 attributes.

2.5.3. Boolean Indicator After T.38 Parameters

   Some implementations incorrectly use a colon (':') followed by a
   number (zero or one) after the attributes T38FaxFillBitRemoval,
   T38FaxTranscodingMMR, and T38FaxTranscodingJBIG. Implementations
   that receive such erroneous encodings MAY interpret the value ":0" as
   lack of support for the option and all other values as support for
   the option.

3. Call Flow Examples

   In this section, we provide three example call flows. The first one
   illustrates a T.38 fax call under Call Agent control on both the
   originating and terminating side. The second one illustrates the use
   of multiple and different options on the two sides. The third one
   illustrates the interaction with a SIP endpoint.

3.1. Call Agent Controlled T.38 Strict

   In this example, both sides are under strict T.38 Call Agent control.
   We assume the originating and terminating Call Agents communicate via
   the Session Initiation Protocol (SIP) [RFC3261]. Furthermore, the
   originating fax machine does not generate CNG tone, which is typical
   of early (i.e., pre-1993) fax machines.

```
    ----------------------------------------------------------------
   | #|     GW-o      |     CA-o      |     CA-t      |     GW-t      |
   |==|==============|==============|==============|==============|
   | 1|           <-|CRCX          |              |              |
   | 2|     200(sdp-o)|->            |              |              |
   | 3|              |  INVITE(sdp-o)|->            |              |
   | 4|              |              |    CRCX(sdp-o)|->            |
   | 5|              |              |           <-|200 (sdp-t)   |
   | 6|              |              |  <-|200(sdp-t)|              |
   | 7|           <-|MDCX(sdp-t)   |              |              |
   | 8|         200|->            |              |              |
   |--|--------------|--------------|--------------|--------------|
   | 9|              |              |              |  <- ANS/     |
   |  |              |              |              |     T.30 CED |
   |10|              |              |              |  <- V.21 fax |
   |  |              |              |              |     preamble |
   |11|              |              |           <-|NTFY(t38 start)|
   |12|              |              |         200|->            |
   |13|              |              |    MDCX(t38)|->            |
   |14|              |              |           <-|200(sdp-t2)   |
   |15|              |              |  <- INVITE(sdp-t2)|          |
   |16|           <-|MDCX(sdp-t2)  |              |              |
   |17|     200(sdp-o2)|->          |              |              |
   |18|              | 200(sdp-o2)|->            |              |
   |19|              |              |  MDCX(sdp-o2)|->            |
   |20|              |              |           <-|200           |
   |21|  V.21 fax ->|              |              |              |
   |  |  preamble    |              |              |              |
   |22|NTFY(t38 start)|->           |              |              |
   |23|           <-|200           |              |              |
   |24|           <-|RQNT(T38 event)|             |              |
   |25|         200|->            |              |              |
   |--|--------------|--------------|--------------|--------------|
   |26|              |              |              |  (fax ends)  |
   |27|              |              |           <-|NTFY(t38 stop)|
   |28|              |              |         200|->            |
   |29|NTFY(t38 stop) |->           |              |              |
   |30|           <-|200           |              |              |
    ----------------------------------------------------------------
```

   Step 1:

   The Call Agent issues a CreateConnection command to the gateway,
   instructing it to use PCMU media encoding and to use the strict Call
   Agent controlled T.38 procedure. Consequently, the Call Agent asks
   the gateway to notify it of the "t38" event:

      CRCX 1000 ds/ds1-1/1@gw-o.example.net MGCP 1.0
      C: 1
      L: a:PCMU, fxr/fx:t38
      M: recvonly
      R: fxr/t38
      X: 1

   Step 2:

   The gateway acknowledges the command and includes SDP with codec
   information and RFC 3407 [RFC3407] capability information:

      200 1000 OK
      I:1

      v=0
      o=- 25678 753849 IN IP4 192.0.2.1
      s=-
      c=IN IP4 192.0.2.1
      t=0 0
      m=audio 3456 RTP/AVP 0
      a=sqn: 0
      a=cdsc: 1 audio RTP/AVP 0 18
      a=cdsc: 3 image udptl t38

   Step 3:

   The originating Call Agent sends a SIP INVITE message with the SDP to
   the terminating Call Agent.

   Step 4:

   The terminating Call Agent issues a CreateConnection command to the
   terminating gateway, instructing it to use PCMU media encoding and to
   use the strict Call Agent controlled T.38 procedure. Consequently,
   the Call Agent asks the gateway to notify it of the "t38" event:

```
CRCX 2000 ds/ds1-1/2@gw-t.example.net MGCP 1.0
C: 2
L: a:PCMU, fxr/fx:t38
M: sendrecv
R: fxr/t38
X: 20

v=0
o=- 25678 753849 IN IP4 192.0.2.1
s=-
c=IN IP4 192.0.2.1
t=0 0
m=audio 3456 RTP/AVP 0
a=sqn: 0
a=cdsc: 1 audio RTP/AVP 0 18
a=cdsc: 3 image udptl t38
```

   Step 5:

   The terminating gateway supports T.38, and the
   RemoteConnectionDescriptor included indicates that the other side
   supports T.38 as well, so the strict T.38 Call Agent controlled
   procedure requested can be used. The terminating gateway sends back
   a success response with its SDP, which also includes capability
   information:

```
200 2000 OK
I:2

v=0
o=- 25678 753849 IN IP4 192.0.2.2
s=-
c=IN IP4 192.0.2.2
t=0 0
m=audio 1296 RTP/AVP 0
a=sqn: 0
a=cdsc: 1 audio RTP/AVP 0 18
a=cdsc: 3 image udptl t38
```

   Step 6:

   The terminating Call Agent sends back a SIP 200 OK response to the
   originating Call Agent, which in turn sends a SIP ACK (not shown).

   Step 7:

   The originating Call Agent in turn sends a ModifyConnection command
   to the originating gateway:

```
      MDCX 1001 ds/ds1-1/1@gw-o.example.net MGCP 1.0
      C: 1
      I: 1
      M: sendrecv

      v=0
      o=- 25678 753849 IN IP4 192.0.2.2
      s=-
      c=IN IP4 192.0.2.2
      t=0 0
      m=audio 1296 RTP/AVP 0
      a=sqn: 0
      a=cdsc: 1 audio RTP/AVP 0 18
      a=cdsc: 3 image udptl t38
```

   The ModifyConnection command does not repeat the
   LocalConnectionOptions sent previously. As far as fax handling is
   concerned, the gateway therefore attempts to continue using the
   current fax handling procedure, i.e., strict Call Agent controlled
   T.38. Since the capability information indicates the other side
   supports T.38, the gateway will in fact be able to use the strict
   Call Agent controlled T.38 procedure. Had there not been any support
   for T.38 in the RemoteConnectionDescriptor, then this command would
   still have succeeded, however there would be no special fax handling
   procedure (since strict mode could not be supported).

   Step 8:

   The gateway acknowledges the command. At this point, a call is
   established using PCMU encoding, and if a fax call is detected, the
   Call Agent controlled T.38 procedure will be initiated.

Steps 9-11:

A fax call now occurs. The T.30 CED tone (a.k.a. V.25 ANS) is sent
-- in this case, it is simply passed through the current PCMU
encoding. Since both fax and modem calls can start with this
sequence, it is not possible to determine that this is a fax call
until step 10, where the V.21 fax preamble is detected.

The gateway was instructed to apply the Call Agent controlled T.38
procedure for fax calls, so it begins to mute audio, generates the
"t38(start)" event, and notifies the Call Agent:

```
NTFY 2500 ds/ds1-1/2@gw-t.example.net MGCP 1.0
O: fxr/t38(start)
X: 20
```

Step 12:

The Call Agent acknowledges the Notify command:

```
200 2500 OK
```

Step 13:

The Call Agent then instructs the terminating gateway to use the
"image/t38" MIME type instead:

```
MDCX 2002 ds/ds1-1/2@gw-t.example.net MGCP 1.0
C: 2
I: 2
L: a:image/t38
R: fxr/t38
X: 21
```

Step 14:

The gateway changes to T.38 and sends back a success response with
updated SDP:

```
200 2002 OK

v=0
o=- 25678 753850 IN IP4 192.0.2.2
s=-
c=IN IP4 192.0.2.2
t=0 0
m=image 1296 udptl t38
a=sqn: 0
a=cdsc: 1 audio RTP/AVP 0 18
a=cdsc: 3 image udptl t38
```

Note that since the gateway's current RemoteConnectionDescriptor (as
opposed to the LocalConnectionDescriptor returned here) does not list
"image/t38" as a valid encoding method, the terminating gateway is
still muting the media and is now waiting for an updated
RemoteConnectionDescriptor with "image/t38".

Step 15:

The terminating Call Agent sends a re-INVITE to the originating Call
Agent with the updated SDP.

Step 16:

The originating Call Agent then sends a ModifyConnection command to
the originating gateway:

```
MDCX 1003 ds/ds1-1/1@gw-o.example.net MGCP 1.0
C: 1
I: 1

v=0
o=- 25678 753850 IN IP4 192.0.2.2
s=-
c=IN IP4 192.0.2.2
t=0 0
m=image 1296 udptl t38
a=sqn: 0
a=cdsc: 1 audio RTP/AVP 0 18
a=cdsc: 3 image udptl t38
```

Step 17:

The originating gateway changes to T.38 and sends back a success
response with updated SDP:

    200 1003 OK

    v=0
    o=- 25678 753850 IN IP4 192.0.2.1
    s=-
    c=IN IP4 192.0.2.1
    t=0 0
    m=image 3456 udptl t38
    a=sqn: 0
    a=cdsc: 1 audio RTP/AVP 0 18
    a=cdsc: 3 image udptl t38

Step 18:

The originating Call Agent sends a SIP 200 OK response with the
updated SDP to the terminating Call Agent, which in turn sends a SIP
ACK (not shown).

Step 19:

The terminating Call Agent sends a ModifyConnection with the updated
SDP to the terminating gateway:

    MDCX 2003 ds/ds1-1/2@gw-t.example.net MGCP 1.0
    C: 2
    I: 2

    v=0
    o=- 25678 753850 IN IP4 192.0.2.1
    s=-
    c=IN IP4 192.0.2.1
    t=0 0
    m=image 3456 udptl t38
    a=sqn: 0
    a=cdsc: 1 audio RTP/AVP 0 18
    a=cdsc: 3 image udptl t38

Step 20:

The terminating gateway sends back a success response:

    200 2003 OK

Since the terminating gateway now has a RemoteConnectionDescriptor
with "image/t38" as valid media, it can start exchanging T.38 with
the originating gateway.

Steps 21, 22:

The originating endpoint detects V.21 fax preamble. Even though the
endpoint is already using "image/t38" for media, it generates a
"t38(start)" event and notifies the Call Agent.

    NTFY 3500 ds/ds1-1/1@gw-o.example.net MGCP 1.0
    O: fxr/t38(start)
    X: 1

Steps 23, 24:

The Call Agent acknowledges the Notify command, then issues a new
request for notification of the "t38" event.

    200 3500 OK
    .
    RQNT 1004 ds/ds1-1/1@gw-o.example.net MGCP 1.0
    R: fxr/t38
    X: 2

Step 25:

The gateway acknowledges the command.

    200 1004 OK

Steps 26, 27:

When the fax ends, a "t38(stop)" event is generated by the
terminating endpoint, which is notified to the Call Agent:

    NTFY 2501 ds/ds1-1/2@gw-t.example.net MGCP 1.0
    O: t38(stop)
    X: 21

Step 28:

The Call Agent acknowledges the Notify command:

    200 2501 OK

Step 29:

The originating endpoint also generates a "t38(stop)" event, which is
notified to the Call Agent:

    NTFY 3502 ds/ds1-1/1@gw-o.example.net MGCP 1.0
    O: t38(stop)
    X: 2

Step 30:

The Call Agent acknowledges the Notify command:

    200 3502 OK

The fax call is now over. The Call Agent may now decide to change
back to a voice codec, delete the connection, or do something
different.

3.2. Multiple and Different Options

   In this example, the originating gateway is instructed to use the
   gateway procedure, whereas the terminating gateway is given a choice
   between the gateway procedure and the strict t38 procedure.
   Furthermore, the originating fax machine is generating CNG tone.

```
    ---------------------------------------------------------------------
   | #|    GW-o       |    CA-o       |    CA-t       |    GW-t       |
   |==|===============|===============|===============|===============|
   | 1|            <- |CRCX           |               |               |
   | 2|    200(sdp-o) |->             |               |               |
   | 3|               |  INVITE(sdp-o)|->             |               |
   | 4|               |               |    CRCX(sdp-o)|->             |
   | 5|               |               |            <- |200 (sdp-t)    |
   | 6|               |            <- |200(sdp-t)     |               |
   | 7|            <- |MDCX(sdp-t)    |               |               |
   | 8|        200    |->             |               |               |
   |--|---------------|---------------|---------------|---------------|
   | 9|       CNG ->  |               |               |               |
   |10|               |               |               |<- ANS/T.30 CED|
   |11|               |               |               |<- V.21 fax    |
   |  |               |               |               |    preamble   |
   |12|               |               |            <- |NTFY(t38 start)|
   |13|               |               |        200 |->                |
   |14|               |               |    MDCX(t38)|->               |
   |15|               |               |            <- |200(sdp-t2)    |
   |16|               |            <- |INVITE(sdp-t2) |               |
   |17|            <- |MDCX(sdp-t2)   |               |               |
   |18|    200(sdp-o2)|->             |               |               |
   |19|               |   200(sdp-o2) |->             |               |
   |20|               |               |   MDCX(sdp-o2)|->             |
   |21|               |               |            <- |200            |
   |--|---------------|---------------|---------------|---------------|
   |22|               |               |               |   (fax ends)  |
   |23|               |               |            <- |NTFY(t38 stop) |
   |24|               |               |        200 |->                |
    ---------------------------------------------------------------------
```

Step 1:

The Call Agent issues a CreateConnection command to the gateway,
instructing it to use PCMU media encoding and to use the gateway
procedure. Consequently, the Call Agent asks the gateway to notify
it of the "gwfax" event:

```
CRCX 1000 ds/ds1-1/1@gw-o.example.net MGCP 1.0
C: 1
L: a:PCMU, fxr/fx:gw
M: recvonly
R: fxr/gwfax
X: 1
```

Step 2:

The gateway acknowledges the command and includes SDP with codec
information and capability information:

```
200 1000 OK
I:1

v=0
o=- 25678 753849 IN IP4 192.0.2.1
s=-
c=IN IP4 192.0.2.1
t=0 0
m=audio 3456 RTP/AVP 0
a=sqn: 0
a=cdsc: 1 audio RTP/AVP 0 18
a=cdsc: 3 image udptl t38
a=X-FaxScheme: 123
```

We assume the gateway supports some other fax scheme, and it
indicates this by including an attribute "X-FaxScheme: 123".

Step 3:

The originating Call Agent sends a SIP INVITE message with the SDP to
the terminating Call Agent.

Step 4:

The terminating Call Agent issues a CreateConnection command to the
terminating gateway, instructing it to use PCMU media encoding and to
use either the gateway procedure or the strict Call Agent controlled
T.38 procedure. Consequently, the Call Agent asks the gateway to
notify it of both the "gwfax" and "t38" events:

```
CRCX 2000 ds/ds1-1/2@gw-t.example.net MGCP 1.0
C: 2
L: a:PCMU, fxr/fx:gw;t38
M: sendrecv
R: fxr/t38, fxr/gwfax
X: 20

v=0
o=- 25678 753849 IN IP4 192.0.2.1
s=-
c=IN IP4 192.0.2.1
t=0 0
m=audio 3456 RTP/AVP 0
a=sqn: 0
a=cdsc: 1 audio RTP/AVP 0 18
a=cdsc: 3 image udptl t38
a=X-FaxScheme: 123
```

Step 5:

The terminating gateway does not support any special gateway fax
handling; however, it does support T.38, and the
RemoteConnectionDescriptor included indicates that the other side
supports T.38 as well, so the strict T.38 Call Agent controlled
procedure requested can be honored. The terminating gateway sends
back a success response with its SDP, which also includes capability
information:

```
200 2000 OK
I:2

v=0
o=- 25678 753849 IN IP4 192.0.2.2
s=-
c=IN IP4 192.0.2.2
t=0 0
m=audio 1296 RTP/AVP 0
a=sqn: 0
a=cdsc: 1 audio RTP/AVP 0 18
a=cdsc: 3 image udptl t38
```

Step 6:

The terminating Call Agent sends back a SIP 200 OK response to the
originating Call Agent, which in turn sends a SIP ACK (not shown).

Step 7:

The originating Call Agent in turns sends a ModifyConnection command
to the originating gateway:

    MDCX 1001 ds/ds1-1/1@gw-o.example.net MGCP 1.0
    C: 1
    I: 1
    M: sendrecv

    v=0
    o=- 25678 753849 IN IP4 192.0.2.2
    s=-
    c=IN IP4 192.0.2.2
    t=0 0
    m=audio 1296 RTP/AVP 0
    a=sqn: 0
    a=cdsc: 1 audio RTP/AVP 0 18
    a=cdsc: 3 image udptl t38

The ModifyConnection command does not repeat the
LocalConnectionOptions sent previously. As far as fax handling is
concerned, the gateway therefore attempts to continue using the
current fax handling, i.e., the gateway procedure. The SDP
information returned however does not indicate support for the "X-
FaxScheme: 123", and hence the originating gateway will not invoke
any special fax handling procedure for this call.

Step 8:

The gateway acknowledges the command. At this point, a call is
established using PCMU encoding, and if a fax call is detected, no
special fax handling procedure will occur.

Steps 9-12:

A CNG tone is generated by the originating fax, thereby indicating a
fax call. If the gateway was using either of the T.38 modes, or if
it had negotiated support for a special gateway handling procedure
with the other side, a "t38(start)" or "gwfax(start)" event would now
have been generated and the switch to T.38 (or special gateway
handling) could start. However, since the negotiation with the
terminating gateway resulted in the originating gateway not doing
anything special for fax, no such event is generated. Instead, the
"nopfax(start)" event is now generated, but since the Call Agent has
not requested this event, it is not detected and hence not reported
to the Call Agent. Consequently, the CNG tone is simply passed
through the current PCMU encoding without the (originating) Call
Agent being aware of the fax call.

Subsequently, the T.30 CED tone (a.k.a. V.25 ANS) occurs -- in this
case, it is also simply passed through the current PCMU encoding.
Since both fax and modem calls can start with this sequence, it is
not possible to determine that this is a fax call until step 11,
where the V.21 fax preamble is detected.

The terminating gateway is using the Call Agent controlled T.38
procedure for fax calls, so it begins to mute audio, generates the
"t38(start)" event, and notifies the Call Agent:

    NTFY 2500 ds/ds1-1/2@gw-t.example.net MGCP 1.0
    O: fxr/t38(start)
    X: 20

Step 13:

The Call Agent acknowledges the Notify command:

    200 2500 OK

Step 14:

The Call Agent then instructs the terminating gateway to use the
"image/t38" MIME type instead:

    MDCX 2002 ds/ds1-1/2@gw-t.example.net MGCP 1.0
    C: 2
    I: 2
    L: a:image/t38
    R: fxr/t38
    X: 21

    Step 15:

    The gateway changes to T.38 and sends back a success response with
    updated SDP:

        200 2002 OK

        v=0
        o=- 25678 753850 IN IP4 192.0.2.2
        s=-
        c=IN IP4 192.0.2.2
        t=0 0
        m=image 1296 udptl t38
        a=sqn: 0
        a=cdsc: 1 audio RTP/AVP 0 18
        a=cdsc: 3 image udptl t38

    Note that since the terminating gateway's last received
    RemoteConnectionDescriptor (as opposed to the
    LocalConnectionDescriptor returned here) did not list "image/t38" as
    a valid encoding method, the terminating gateway is still muting the
    media and is now waiting for an updated RemoteConnectionDescriptor
    with "image/t38".

    Step 16:

    The terminating Call Agent sends a re-INVITE to the originating Call
    Agent with the updated SDP.

    Step 17:

    The originating Call Agent then sends a ModifyConnection command to
    the originating gateway:

        MDCX 1003 ds/ds1-1/1@gw-o.example.net MGCP 1.0
        C: 1
        I: 1

        v=0
        o=- 25678 753850 IN IP4 192.0.2.2
        s=-
        c=IN IP4 192.0.2.2
        t=0 0
        m=image 1296 udptl t38
        a=sqn: 0
        a=cdsc: 1 audio RTP/AVP 0 18
        a=cdsc: 3 image udptl t38

Step 18:

The originating gateway changes to T.38 and sends back a success
response with updated SDP:

```
200 1003 OK

v=0
o=- 25678 753850 IN IP4 192.0.2.1
s=-
c=IN IP4 192.0.2.1
t=0 0
m=image 3456 udptl t38
a=sqn: 0
a=cdsc: 1 audio RTP/AVP 0 18
a=cdsc: 3 image udptl t38
```

Step 19:

The originating Call Agent sends a SIP 200 OK response with the
updated SDP to the terminating Call Agent, which in turn sends a SIP
ACK (not shown).

Step 20:

The terminating Call Agent sends a ModifyConnection with the updated
SDP to the terminating gateway:

```
MDCX 2003 ds/ds1-1/2@gw-t.example.net MGCP 1.0
C: 2
I: 2

v=0
o=- 25678 753850 IN IP4 192.0.2.1
s=-
c=IN IP4 192.0.2.1
t=0 0
m=image 3456 udptl t38
a=sqn: 0
a=cdsc: 1 audio RTP/AVP 0 18
a=cdsc: 3 image udptl t38
```

Step 21:

The terminating gateway sends back a success response:

    200 2003 OK

Since the terminating gateway now has a RemoteConnectionDescriptor
with "image/t38" as valid media, it can start exchanging T.38 with
the originating gateway.

Steps 22, 23:

When the fax ends, a "t38(stop)" event is generated, which is
notified to the Call Agent:

    NTFY 2501 ds/ds1-1/2@gw-t.example.net MGCP 1.0
    O: t38(stop)
    X: 21

Step 24:

The Call Agent acknowledges the Notify command:

    200 2501 OK

The fax call is now over. The Call Agent may now decide to change
back to a voice codec, delete the connection, or do something
different.

3.3. Interaction with SIP Endpoints

   In this example, we show interaction with a SIP endpoint that does
   not support the RFC 3407 [RFC3407] capability descriptors. To
   accommodate such endpoints, the T.38 loose mode is being used (at the
   risk of initiating T.38 to an endpoint that does not support it).
   Once again, the originating fax does not generate CNG tone.

```
    ------------------------------------------------------------------
   | #|     GW-o      |     CA-o      |    SIP-UA-t    |     fax       |
   |==|===============|===============|===============|===============|
   | 1|            <-|CRCX           |               |               |
   | 2|     200(sdp-o)|->             |               |               |
   | 3|               |   INVITE(sdp-o)|->            |               |
   | 4|               |             <-|200(sdp-t)     |               |
   | 5|               |           ACK|->              |               |
   | 6|            <-|MDCX(sdp-t)    |               |               |
   | 7|          200|->             |               |               |
   |--|---------------|---------------|---------------|---------------|
   | 8|               |               |               | <- ANS/       |
   |  |               |               |               |    T.30 CED   |
   | 9|               |               |               | <- V.21 fax   |
   |  |               |               |               |    preamble   |
   |10|               |             <-|INVITE(sdp-t2) |               |
   |11|            <-|MDCX(sdp-t2)   |               |               |
   |12|    200(sdp-o2)|->             |               |               |
   |13|               |  200(sdp-o2)|->              |               |
   |14|               |             <-|ACK            |               |
   |15|   V.21 fax ->|               |               |               |
   |  |   preamble    |               |               |               |
   |16|NTFY(t38 start)|->             |               |               |
   |17|            <-|200            |               |               |
   |18|            <-|RQNT(T38 event)|               |               |
   |19|          200|->             |               |               |
   |--|---------------|---------------|---------------|---------------|
   |20|               |               |               | (fax ends)    |
   |21|               |             <-|BYE            |               |
   |22|               |          200|->              |               |
   |23|NTFY(t38 stop) |->             |               |               |
   |24|            <-|200            |               |               |
    ------------------------------------------------------------------
```

Step 1:

The Call Agent issues a CreateConnection command to the gateway,
instructing it to use PCMU media encoding and to use the loose Call
Agent controlled T.38 procedure. Consequently, the Call Agent asks
the gateway to notify it of the "t38" event:

```
CRCX 1000 ds/ds1-1/1@gw-o.example.net MGCP 1.0
C: 1
L: a:PCMU, fxr/fx:t38-loose
M: recvonly
R: fxr/t38
X: 1
```

Step 2:

The gateway acknowledges the command and includes SDP with codec
information and RFC 3407 [RFC3407] capability information:

```
200 1000 OK
I:1

v=0
o=- 25678 753849 IN IP4 192.0.2.1
s=-
c=IN IP4 192.0.2.1
t=0 0
m=audio 3456 RTP/AVP 0
a=sqn: 0
a=cdsc: 1 audio RTP/AVP 0 18
a=cdsc: 3 image udptl t38
```

    Step 3:

    The originating SIP User Agent (UA) sends a SIP INVITE message with
    the SDP to the terminating Call Agent (not all SIP details shown
    here):

        INVITE sip:bob@biloxi.example.com SIP/2.0
        ...
        Content-Type: application/sdp
        Content-Length: 167

        v=0
        o=- 25678 753849 IN IP4 192.0.2.1
        s=-
        c=IN IP4 192.0.2.1
        t=0 0
        m=audio 3456 RTP/AVP 0
        a=sqn: 0
        a=cdsc: 1 audio RTP/AVP 0 18
        a=cdsc: 3 image udptl t38

    Step 4:

    The terminating SIP User Agent sends back a SIP 200 OK response (not
    all SIP details shown) to the originating Call Agent:

        SIP/2.0 200 OK
        ...
        Content-Type: application/sdp
        Content-Length: 100

        v=0
        o=- 25678 753849 IN IP4 192.0.2.2
        s=-
        c=IN IP4 192.0.2.2
        t=0 0
        m=audio 1296 RTP/AVP 0

    Note that the terminating SIP User Agent does not use the RFC 3407
    [RFC3407] capability descriptor to indicate support for (or lack of
    support for) T.38.

   Step 5:

   The originating Call Agent receives the SIP 200 response and sends a
   SIP ACK message to the terminating SIP UA.

   Note that the Call Agent does not know whether the peer entity
   supports T.38. In order to figure this out, the Call Agent could
   send a SIP OPTIONS request to the terminating SIP UA, requesting it
   to return its capabilities (not shown). Note that this can of course
   be done towards any SIP peer, e.g., if the other side was a Call
   Agent speaking SIP it could be done there too.

   Step 6:

   The originating Call Agent in turns sends a ModifyConnection command
   to the originating gateway:

      MDCX 1001 ds/ds1-1/1@gw-o.example.net MGCP 1.0
      C: 1
      I: 1
      M: sendrecv

      v=0
      o=- 25678 753849 IN IP4 192.0.2.2
      s=-
      c=IN IP4 192.0.2.2
      t=0 0
      m=audio 1296 RTP/AVP 0

   The ModifyConnection command does not repeat the
   LocalConnectionOptions sent previously. As far as fax handling is
   concerned, the gateway therefore attempts to continue using the
   current fax handling procedure, i.e., loose Call Agent controlled
   T.38. The T.38 loose procedure can always be supported, and hence a
   switch to T.38 will be attempted if the originating gateway detects a
   fax call.

   Step 7:

   The gateway acknowledges the command. At this point, a call is
   established using PCMU encoding, and if a fax call is detected, the
   Call Agent controlled T.38 procedure will be initiated.

Steps 8, 9:

A fax call now occurs. The T.30 CED tone (a.k.a. V.25 ANS) is
sent--in this case, it is simply passed through the current PCMU
encoding. Since both fax and modem calls can start with this
sequence, it is not possible to determine that this is a fax call
until step 9, where the V.21 fax preamble is detected.

Step 10:

The terminating SIP UA does in fact support T.38 and, upon detecting
the fax call, attempts to change to T.38. Consequently, it sends a
re-INVITE to the originating Call Agent with an updated SDP
indicating a switch to T.38.

```
INVITE sip:ca@ca-o.example.net SIP/2.0
...
Content-Type: application/sdp
Content-Length: 100

v=0
o=- 25678 753850 IN IP4 192.0.2.2
s=-
c=IN IP4 192.0.2.2
t=0 0
m=image 1296 udptl t38
```

Step 11:

The originating Call Agent then sends a ModifyConnection command to
the originating gateway:

```
MDCX 1003 ds/ds1-1/1@gw-o.example.net MGCP 1.0
C: 1
I: 1

v=0
o=- 25678 753850 IN IP4 192.0.2.2
s=-
c=IN IP4 192.0.2.2
t=0 0
m=image 1296 udptl t38
```

   Step 12:

   The originating gateway changes to T.38 and sends back a success
   response with updated SDP:

```
200 1003 OK

v=0
o=- 25678 753850 IN IP4 192.0.2.1
s=-
c=IN IP4 192.0.2.1
t=0 0
m=image 3456 udptl t38
a=sqn: 0
a=cdsc: 1 audio RTP/AVP 0 18
a=cdsc: 3 image udptl t38
```

   Step 13:

   The originating Call Agent sends a SIP 200 OK response with the
   updated SDP to the terminating SIP User Agent:

```
SIP/2.0 200 OK
...
Content-Type: application/sdp
Content-Length: 167

v=0
o=- 25678 753850 IN IP4 192.0.2.1
s=-
c=IN IP4 192.0.2.1
t=0 0
m=image 3456 udptl t38
a=sqn: 0
a=cdsc: 1 audio RTP/AVP 0 18
a=cdsc: 3 image udptl t38
```

   Step 14:

   The terminating SIP User Agent receives the SIP 200 and sends a SIP
   ACK.

   Since the terminating SIP User Agent now has a
   RemoteConnectionDescriptor with "image/t38" as valid media, it can
   start exchanging T.38 with the originating gateway (and vice versa).

Steps 15, 16:

The originating endpoint detects V.21 fax preamble. Even though the
endpoint is already using "image/t38" for media, it generates a
"t38(start)" event and notifies the Call Agent.

```
NTFY 3500 ds/ds1-1/1@gw-o.example.net MGCP 1.0
O: fxr/t38(start)
X: 1
```

Steps 17, 18:

The Call Agent acknowledges the Notify command and issues a new
(piggybacked) request for notification of the T38 event.

```
200 3500 OK
.
RQNT 1004 ds/ds1-1/1@gw-o.example.net MGCP 1.0
R: fxr/t38
X: 2
```

Step 19:

The gateway acknowledges the command.

```
200 1004 OK
```

Steps 20-22:

When the fax ends, the terminating SIP UA decides to tear down the
call and hence sends a SIP BYE message, which the Call Agent responds
to with a SIP 200.

Step 23:

The originating endpoint also generates a "t38(stop)" event, which is
notified to the Call Agent:

```
NTFY 3502 ds/ds1-1/1@gw-o.example.net MGCP 1.0 O: t38(stop) X: 2
```

   Step 24:

   The Call Agent acknowledges the Notify command:

      200 3502 OK

   The fax call is now over. The Call Agent may now decide to change
   back to a voice codec, delete the connection, or do something
   different.

4. Security Considerations

   The MGCP fax package itself is not known to introduce any new
   security concerns. However, implementers should note that T.38 media
   is currently transported over UDP (UDPTL) or TCP in the clear and
   without any integrity protection. If for example security services
   are in place to protect RTP media streams, these will thus not be in
   effect for the T.38 media stream. If such lack of security is a
   concern, the fax LocalConnectionOptions allowing T.38 in this package
   SHOULD NOT be used, i.e., the "off" (or a new secure extension) fax
   LocalConnectionOption should be used.

5. IANA Considerations

   IANA has registered the following MGCP package:

      Package Title   Name   Version
      -------------   ----   -------
      Fax   FXR   0

6. Normative References

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

   [RFC3435] Andreasen, F. and B. Foster, "Media Gateway Control
             Protocol (MGCP) Version 1.0", RFC 3435, January 2003.

   [T38]   ITU-T Recommendation T.38, "Procedures for real-time Group
             3 facsimile communication over IP networks", March 2002.

   [RFC3407] Andreasen, F., "Session Description Protocol (SDP) Simple
             Capability Declaration", RFC 3407, October 2002.

7. Informative References

   [T30]   ITU-T Recommendation T.30, "Procedures for document
              facsimile transmission in the general switched telephone
              network", July 2003.

   [RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston,
              A., Peterson, J., Sparks, R., Handley, M., and E. Schooler,
              "SIP: Session Initiation Protocol", RFC 3261, June 2002.

   [RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model
              with Session Description Protocol (SDP)", RFC 3264, June
              2002.

Acknowledgements

   Several people have contributed to the development of the MGCP fax
   package. In particular, the author would like to thank Bill Foster,
   Paul Jones, Gary Kelly, Rajesh Kumar, Dave Horwitz, Hiroshi Tamura,
   Rob Thompson, and the CableLabs PacketCable NCS focus team for their
   contributions.

Authors' Addresses

   Flemming Andreasen
   Cisco Systems
   499 Thornall Street, 8th Floor
   Edison, NJ 08837

   EMail: fandreas@cisco.com

   David Hancock
   CableLabs
   858 Coal Creek Circle
   Louisville, CO 80027

   EMail: d.hancock@cablelabs.com