Internet Engineering Task Force
INTERNET DRAFT
<draft-huitema-megaco-mgcp-firewall-00.txt>
February 16, 1999

Christian Huitema
Flemming Andreasen
Bellcore

Expires: August 16, 1999

Media Gateway Control Protocol (MGCP) Support for Packet Relays

Christian Huitema, Flemming Andreasen

Status of this document

Abstract

The Media Gateway Control Protocol (MGCP) organizes the communication between a Media Gateway controller, or call agent, and a Media Gateway, e.g. a Voice over IP gateway or a Network Access Server. MGCP is defined in a companion document [1]. This document explains how MGCP can be used to handle "packet relays", such as firewalls. It contains an introduction, two example call flows, and a discussion of some open questions.

1. Introduction

The Media Gateway Control Protocol can be used to control different types of endpoints, such as "residential gateways", "trunking gateways" or "packet relays." As stated in the MGCP specification,

> A packet relay endpoint is a specific form of conference bridge, that typically only supports two connections. Packets relays can be found in firewalls between a protected and an open network, or in transcoding servers used to provide interoperation between incompatible gateways, for example gateways that do not support compatible compression algorithms, or gateways that operate over different transmission networks such as IP and ATM.

```
                        +-------
+---------------------+ |
|Packet relay endpoint|  2 connections
+---------------------+ |
                        +-------
```

It has been incorrectly stated that the "single endpoint" model adopted by MGCP would not support these relays. This document will show that on the contrary, MGCP does indeed support this mode of operation, by showing the use of the protocol on two example call flows. We will then discuss the specific issues posed by the support of packet relays, and how MGCP could be modified to better support these relays.

2. Call flows

The following two call flows are example of usage of packet relays in a VoIP environment. In our examples, the packet relay is a firewall that protects a local area network. We will in fact go through two detailed call flows:

* a call from a PSTN user to a residential gateway (RGW), through a trunking gateway (TGW) and a firewall (FW).

* a call from an H.323 terminal to a residential gateway (RGW), through a firewall (FW).

In the PSTN case, we assume that the same call agent controls all the network elements. We choose this hypothesis to avoid having to represent in our diagrams the actual exchanges between call agents, for which several solutions are available.

2.1. Call from a TGW to a firewalled RGW

```
      ___________________________________________________________
     | CO |  SS7/|  TGW |     CA    |   FW  |   RGW |     Usr   |
     |____|______|______|___________|_______|_______|___________|
     | IAM|   -> |      |           |       |       |           |
     |    |  IAM |  - - |     ->    |       |       |           |
     |    |      |   <- |    CRCX   |       |       |           |
     |    |      |  Ack |     ->    |       |       |           |
     |    |      |      |   CRCX(1) |   ->  |       |           |
     |    |      |      |     <-    |   Ack |       |           |
     |    |      |      |   CRCX(2) |   ->  |       |           |
     |    |      |      |     <-    |   Ack |       |           |
     |    |      |      |  CRCX+RQNT|   - - |   ->  |    ring   |
     |    |      |      |     <-    |   - - |   Ack |           |
     |    |      |      |   MDCX(2) |   ->  |       |           |
     |    |      |      |     <-    |   Ack |       |           |
     |    |   <- |  - - |     ACM   |       |       |           |
     | <- |  ACM |      |           |       |       |           |
     |    |      |      |           |       |       |  off-hook |
     |    |      |      |     <-    |   - - |  NTFY |           |
     |    |      |      |    NTRQ   |   - - |   ->  |           |
     |    |      |      |     <-    |   - - |   Ack |           |
     |    |      |   <- |    MDCX   |       |       |           |
     |    |      |  Ack |     ->    |       |       |           |
     |    |   <- |  - - |     ANM   |       |       |           |
     | <- |  ANM |      |           |       |       |           |
     |____|______|______|___________|_______|_______|___________|
     |    |      |      |     <-    |   - - |  NTFY |   on-hook |
     |    |      |      |     Ack   |   - - |   ->  |           |
     |    |      |      |   DLCX(1) |   ->  |       |           |
     |    |      |      |     <-    |  perf |       |           |
     |    |      |      |           |   data|       |           |
     |    |      |      |   DLCX(2) |   ->  |       |           |
     |    |      |      |     <-    |  perf |       |           |
     |    |      |      |           |  data |       |           |
     |    |      |      |  DLCX+RQNT|   - - |   ->  |           |
     |    |      |      |     <-    |   - - |  perf |           |
     |    |      |      |           |       |  data |           |
     |    |      |   <- |    DLCX   |       |       |           |
     |    |      |  Perf|           |       |       |           |
     |    |      |  data|     ->    |       |       |           |
     |    |   <- |  - - |     REL   |       |       |           |
     | <- |  REL |      |           |       |       |           |
     | RLC|   -> |      |           |       |       |           |
     |    |  RLC |  - - |     ->    |       |       |           |
     |____|______|______|___________|_______|_______|___________|
```

This diagram shows the various exchange of messages during a call from a telephone user on the circuit-switched PSTN to a residential user connected to a residential Gateway through a firewall. During these exchanges the Call Agent uses MGCP to control the TGW, the firewall and the residential gateway.

Upon reception of the IAM message, the Call Agent immediately sends a CreateConnection request to the trunking gateway to connect to the incoming trunk, creating a connection:

```
        CRCX 1237 card23/21@trgw-7.example.net MGCP 0.1
        C: A3C47F21456789F0
        L: p:10, a:PCMU
        M: recvonly
```

The trunking gateway immediately acknowledges the creation, sending back the identification of the newly created connection and the session description used to receive audio data:

```
        200 1237 OK
        I: FDE234C8

        v=0
        c=IN IP4 128.96.41.1
        m=audio 3456 RTP/AVP 0
```

The SDP announcement, in our example, specifies the address at which the gateway is ready to receive audio data (128.96.41.1), the transport protocol (RTP), the RTP port (3456) and the audio profile (AVP).

The Call Agent, having seized the incoming trunk, must now reserve a "pass-through" conenction in the firewall. It does so by first opening an "external" connection to a "packet-relay" endpoint in the firewall:

```
        CRCX 1238 relay/$@fw.example.net MGCP 0.1
        C: A3C47F21456789F0
        L: p:10, a:PCMU
        M: sendrecv

        v=0
        c=IN IP4 128.96.41.1
        m=audio 3456 RTP/AVP 0
```

This message carries the session description returned by the ingress gateway. The firewall acknowledges the creation of the connection, and the selection of a relay endpoint:

```
        200 1238 OK
        Z:relay/17@fw.example.net
        I:32F345E2

        v=0
        c=IN IP4 128.96.63.25
        m=audio 1296 RTP/AVP 0
```

The call agent will then request the creation of a second connection on the same endpoint on the firewall, in order to build up the "protected" leg of the call:

```
        CRCX 1239 relay/17@fw.example.net MGCP 0.1
        C: A3C47F21456789F0
        L: p:10, a:PCMU
        M: sendrecv
```

The firewall acknowledges the creation of this connection:

```
        200 1239 OK
        I:32F345E3

        v=0
        c=IN IP4 128.96.63.25
        m=audio 1296 RTP/AVP 0
```

In our example, the firewall responds with exactly the same session description that was used for the first command; it could indeed used a different description if it wanted to use, for example, different ports.

The call agent can now send a connection creation request to the residential gateway, together with an embedded Notification Request.

```
        CRCX 1240 relay/17@fw.example.net MGCP 0.1
        C: A3C47F21456789F0
        L: p:10, a:PCMU
        M: sendrecv
        X: 0123456789B1
        R: hd
        S: rg

        v=0
        c=IN IP4 128.96.63.25
        m=audio 1296 RTP/AVP 0
```

The residential gateway will acknowledge the connection command, sending in the session description its own parameters such as address, ports and

RTP profile:

```
        200 1240 OK
        I:12345678

        v=0
        c=IN IP4 10.0.13.17
        m=audio 1296 RTP/AVP 0
```

The Call Agent will relay the information to the firewall using a ModifyConnection command:

```
        MDCX 1241 relay/17@fw.example.net MGCP 0.1
        C: A3C47F21456789F0
        I:32F345E3
        M: sendrecv

        v=0
        c=IN IP4 10.0.13.17
        m=audio 1296 RTP/AVP 0
```

The firewall immediately acknowledges the modification:

```
        200 1241 OK
```

At this stage, the Call Agent has established a half-duplex transmission path. The residential gateway is instructed to look for an off-hook event, and to report it. The Call Agent sends an address complete message (ACM) to the calling switch. (We assume that this switch will generate ringing tones for the calling user.)

When the gateway notices the off hook event, it sends a Notify command to the Call Agent:

```
        NTFY 2001 endpoint/1@rgw-2567.example.net MGCP 0.1
        X: 0123456789B0
        O: hd
```

The Call Agent immediately acknowledges that notification.

```
        200 2001 OK
```

The Call Agent now asks the residential gateway to send a Notify command on the occurrence of an on-hook event. It does so by sending a NotificationRequest to the residential gateway:

```
        RQNT 1242 endpoint/1@rgw-2567.example.net MGCP 0.1
        X: 0123456789B1
        R: hu
```

The gateway acknowledges that command:

200 1242 OK

In parallel, the Call Agent will send a ModifyConnection command to the trunking gateway, to place the connection in full duplex mode. This commands, in our example, also informs the trunking gateway of the session data of the firewall.

```
        MDCX 1243 card23/21@trgw-7.example.net MGCP 0.1
        C: A3C47F21456789F0
        I:FDE234C8
        M: sendrecv

        v=0
        c=IN IP4 128.96.63.25
        m=audio 1296 RTP/AVP 0
```

The trunking gateway will acknowledge that command:

```
        200 1243 OK
```

The Call Agent can now send an answer message (ANM) to the calling switch.

After some time, the Call Agent will have to tear down the call. In our example, this is triggered by the residential user, who hangs up. The Notify command is sent to the Call Agent:

```
        NTFY 2005 endpoint/1@rgw-2567.example.net MGCP 0.1
        X: 0123456789B1
        O: hu
```

The Call Agent acknowledges the notification.

```
        200 2005 OK
```

It will then a DeleteConnection command to the trunking gateway, another to the residential gateway, and two delete connection commands for the two connections that were established on the firewall. These two commands can be grouped on a single message. The command sent to the residential gateway also carries a notification request, to prepare the gateway for the next call.

```
         DLCX 1244 endpoint/1@rgw-2567.example.net MGCP 0.1
         C:A3C47F21456789F0
         I:32F345E2
         X:0123456789B3
         R:hd

         DLCX 1245 card23/21@trgw-7.example.net MGCP 0.1
         C:A3C47F21456789F0
         I:FDE234C8

         DLCX 1246 relay/17@fw.example.net MGCP 0.11
         C:A3C47F21456789F0
         I:32F345E2
         .
         DLCX 1247 relay/17@fw.example.net MGCP 0.11
         C:A3C47F21456789F0
         I:32F345E3
```

The gateways and the firewall will respond with a message that should include a "call parameters" header fields:

```
        250 1244 OK
        P: PS=1245, OS=62345, PR=780, OR=45123, PL=0, JI=12, LA=6

        250 1245 OK
        P: PS=790, OS=45700, PR=1230, OR=61875, PL=15, JI=27, LA=48

        250 1246 OK
        P: PS=1245, OS=62345, PR=780, OR=45123, PL=10, JI=27, LA=48

        250 1247 OK
        P: PS=780, OS=45123, PR=1245, OR=62345, PL=0, JI=12, LA=6
```

The four commands are processed in parallel. As soon as it receives the acknowledgement of the Delete Connection command from the trunking gateway, the Call Agent sends an ISUP "Release" message to the adjacent switch, and then waits for a "Release confirmed" message.

Both gateways, at this point, are ready for the next call.

2.2. Call from a residential gateway (RGW) to an H.323 user

```
________________________________________________________________________
|    Usr    |   RGW  |  FW |       CA     |    H.323  |    Usr   |  GK |
|___________|________|_____|______________|___________|__________|_____|
|           |    <-  |  - -|      RQNT    |           |          |     |
|           |   Ack  |  - -|       ->     |           |          |     |
|           |        |     |              |           |          |     |
| Off-hook  |  Notify|  - -|       ->     |           |          |     |
|           |    <-  |  - -|      Ack     |           |          |     |
| (D.tone)  |    <-  |  - -|      RQNT    |           |          |     |
|           |   Ack  |  - -|       ->     |           |          |     |
|           |        |     |              |           |          |     |
|  Digits   |  Notify|  - -|       ->     |           |          |     |
|    <-     |   - -  |  Ack|              |           |          |     |
|  (prog.)  |    <-  |  - -|   CRCX+RQNT  |           |          |     |
|           |   Ack  |  - -|       ->     |           |          |     |
|           |        |  <- |    CRCX(1)   |           |          |     |
|           |        |  Ack|       ->     |           |          |     |
|           |        |  <- |    CRCX(2)   |           |          |     |
|           |        |  Ack|       ->     |           |          |     |
|           |    <-  |  - -|       MDCX   |           |          |     |
|           |   Ack  |  - -|       ->     |           |          |     |
|           |        |     |  (processing)|           |          |     |
|           |        |     |    TCP-SYN   |     ->    |          |     |
|           |        |     |       <-     |  SYN, ACK |          |     |
|           |        |     |    Set-up+   |           |          |     |
|           |        |     |   faststart  |     ->    |          |     |
|           |        |     |              |     ARQ   |   - - -  |  -> |
|           |        |     |              |     <-    |   - - -  |  ACF|
|           |        |     |       <-     |  alerting |    ring  |     |
|           |        |     |    TCP ACK   |     ->    |          |     |
|(ring back)|    <-  |  - -|      RQNT    |           |          |     |
|           |   Ack  |  - -|       ->     |           |          |     |
|           |        |     |       <-     |  connect +|  off hook|     |
|           |        |     |              |  faststart|          |     |
|           |        |     |    TCP ACK   |     ->    |          |     |
|           |        |  <- |   MDCX+RQNT  |           |          |     |
|           |        |  Ack|       ->     |           |          |     |
|           |    <-  |  - -|      RQNT    |           |          |     |
|           |   Ack  |  - -|       ->     |           |          |     |
| dialtone  |        |     |  (call est.) |           |          |     |
|___________|________|_____|______________|___________|__________|_____|
 ____________________________________________________________________
|    Usr   |   RGW  |     FW    |    CA   |   H.323  |    Usr  |  GK |
|__________|________|___________|_________|__________|_________|_____|
|  on hook |  Notify|     - -   |    ->   |          |         |     |
|    <-    |   - -  |     Ack   |         |          |         |     |
|    <-    |   - -  |  DLCX+RQNT|         |          |         |     |
| perf data|   - -  |     ->    |         |          |         |     |
|          |        |     <-    |  DLCX(1)|          |         |     |
|          |        |  perf data|    ->   |          |         |     |
|          |        |     <-    |  DLCX(2)|          |         |     |
|          |        |  perf data|    ->   |          |         |     |
|          |        |           |  Rel. C.|     ->   |         |     |
|          |        |           |  TCP-FIN|     ->   |         |     |
|          |        |           |    <-   |  FIN, ACK|         |     |
|          |        |           |  TCP ACK|     ->   |         |     |
|          |        |           |         |  (signal)|  On-hook|     |
|          |        |           |         |    DRQ   |   - - - |  -> |
|          |        |           |         |     <-   |   - - - |  DCF|
|__________|________|___________|_________|__________|_________|_____|
```

During these exchanges the MGCP is used by the call agent to control the residential gateway and the firewall. The call will be routed to an **H.323 entity. The first command is a notification request, sent by the** call agent to the residential gateway. The request will consist of the following lines:

```
        RQNT 1201 endpoint/1@rgw-2567.example.net MGCP 0.1
        N: ca@ca1.example.net:5678
        X: 0123456789AB
        R: hd
```

The gateway, at that point, is instructed to look for an off-hook event, and to report it. It will first acknowledge the request, repeating in the acknowledgement message the transaction id that the call agent attached to the query.

```
        200 1201 OK
```

Note that this command is not actually simultaneous with the call. It can be issued long before the actual call, for example when the gateway is turned on, or after the end of a previous call.

When the off hook event is noticed, the gateway sends a Notify command to the call agent:

```
           NTFY 2001 endpoint/1@rgw-2567.example.net MGCP 0.1
           N: ca@ca1.example.net:5678
           X: 0123456789AB
           O: hd
```

The call agent immediately acknowledges that notification.

```
           200 2001 OK
```

The call agent examines the services associated to an off hook action (it could take special actions in the case of a direct line). In most cases, it will send a notification request, asking for digits. The current example provides the gateway with a digit map, and requests the gateway to play a dialtone:

```
        RQNT 1202 endpoint/1@rgw-2567.example.net MGCP 0.1
        N: ca@ca1.example.net:5678
        X: 0123456789AC
        R: hu, [0-9#*T](D)
        D: (0T|00T|[1-7]xxx|8xxxxxxx|#xxxxxxx|*xx|91xxxxxxxxxx|9011x.T)
        S: dl
```

The gateway immediately acknowledges that request.

```
        200 1202 OK
```

The gateway will start accumulating digits according to that digit map. When it has noticed a sufficient set of values, it will notify the observed string to the call agent:

```
           NTFY 2002 endpoint/1@rgw-2567.example.net MGCP 0.1
           N: ca@ca1.example.net:5678
           X: 0123456789AC
           O: 9,1,9,7,3,8,2,9,4,2,6,6
```

The call agent immediately acknowledges that notification.

```
           200 2002 OK
```

At this stage, the call agent seizes the incoming circuit, creating a connection. It will also send together with that creation request a notification request, to stop collecting digits yet continue watch for an on-hook transition:

```
           CRCX 1204 endpoint/1@rgw-2567.example.net MGCP 0.1
           C: A3C47F21456789F0
           L: p:10, a:PCMU
           M: recvonly
           X: 0123456789AD
           R: hu
```

The gateway immediately acknowledges the creation, sending back the identification of the newly created connection and the session description used to receive audio data:

```
           200 1204 OK
           I:  FDE234C8

           v=0
           c=IN IP4 10.11.12.13
           m=audio 3456 RTP/AVP 0
```

The call agent must now create two connections on the firewall. We will assume in this example that the call agent manages the packet endpoints on the firewall, and is thus able to select an endpoint without using the "wildcard" command. When this is the case, the two create connection commands can be sent in parallel, or grouped in a single packet:

```
           CRCX 1205 relay/17@fw.example.net MGCP 0.1
           C: A3C47F21456789F0
           L: p:10, a:PCMU
           M: recvonly

           v=0
           c=IN IP4 10.11.12.13
           m=audio 3456 RTP/AVP 0
           .
           CRCX 1206 relay/17@fw.example.net MGCP 0.1
           C: A3C47F21456789F0
           L: p:10, a:PCMU
           M: sendrecv
```

The firewall acknowledges the two creations, possibly in a single message:

```
200 1205 OK
I:  12345678

v=0
c=IN IP4 128.96.41.1
m=audio 1234 RTP/AVP 0
.
200 1206 OK
I:  12345679

v=0
c=IN IP4 128.96.41.1
m=audio 3456 RTP/AVP 0
```

The call agent, having seized the incoming circuit and "opened a hole" in the firewall, proceeds with call routing. In parallel, it will complete the set-up of the residential gateway by sending a modify command to establish a full duplex connection with the firewall:

```
MDCX 1207 endpoint/1@rgw-2567.example.net MGCP 0.1
C: A3C47F21456789F0
I:  FDE234C8
M: sendrecv

v=0
c=IN IP4 128.96.41.1
m=audio 1234 RTP/AVP 0
```

The residential gateway will acknowledge the modification:

```
200 1207 OK
```

Using local databases, it determines that the dialed digits (912018294266) correspond to a H.323 entity. It will set up a TCP-IP connection and send an H.225/Q.931 "set-up" message to the H.323 entity. In this message, the "faststart" element carries a set of open logical channel proposals, derived from the SDP description received from the calling gateway:

```
        faststart-1 OpenLogicalChannel ::= {
           forwardLogicalChannelNumber   1,
           forwardLogicalChannelParameters {
              dataType   g711Ulaw64k 160, -- 20 ms G.711 frame
              multiplexParameters
                 h2250LogicalChannelParameters H2250LogicalChannelParameters  {
                     sessionID 1,
                     silenceSuppression   FALSE
                 }
           }
        }

        faststart-2 OpenLogicalChannel ::= {
           forwardLogicalChannelNumber   2,
           forwardLogicalChannelParameters {
              dataType              nullData, -- pro forma
              multiplexParameters   none
           },
           reverseLogicalChannelParameters   {
              dataType   g711Ulaw64k 160, -- 20 ms frame
              multiplexParameters
                 h2250LogicalChannelParameters H2250LogicalChannelParameters {
                    sessionID 2,
                    mediaChannel unicastAddress iPAddress {
                       network  '80602901'H, -- 128.96.41.1
                       tsapIdentifier 3456 -- port
                    },
                    mediaControlChannel unicastAddress iPAddress {
                       network  '80602901'H, -- 128.96.41.1
                       tsapIdentifier 3457 -- port
                    },
                    silenceSuppression FALSE
                 }
           }
        }
```

The H.323 alerts the user, and sends an H.225/Q.931 "alerting" message. On reception of this message, the call agent will send a notification request that instruct the RGW to generate ringback tones:

```
           RQNT 1208 endpoint/1@rgw-2567.example.net MGCP 0.1
           X: 0123456789AE
           R: hu
           S: G/rt
```

The residential gateway acknowledges this request:

```
           200 1208 OK
```

When the called user accepts the call, the H.323 entity sends an H.225/Q.931 "set-up" message to the call agent. If the H.323 entity accepted the fast start procedure, the "faststart" parameter will contain the confirmation of the open logical channel creations in the two directions of communication:

```
        faststart-1 OpenLogicalChannel ::= {
           forwardLogicalChannelNumber   1,
           forwardLogicalChannelParameters {
              dataType   g711Ulaw64k 160, -- 20 ms frame
              multiplexParameters h2250LogicalChannelParameters {
                 sessionID 1,
                 mediaChannel unicastAddress iPAddress {
                                 network  '80603F19'H,
                                 tsapIdentifier 3456, -- port
                 } ,
                 mediaControlChannel unicastAddress iPAddress {
                                 network  '80603F19'H,
                                 tsapIdentifier 3457, -- port
                 } ,
                 silenceSuppression   FALSE
              }
          }
        }
        faststart-2 OpenLogicalChannel ::= {
           forwardLogicalChannelNumber   2,
           forwardLogicalChannelParameters {
              dataType   g711Ulaw64k 160, -- 20 ms frame
                    multiplexParameters h2250LogicalChannelParameters  {
                        sessionID 2,
                        silenceSuppression   FALSE
                    }
              }
        }
```

The call agent will send a notification request to the residential gateway, to suppress the ring back tone:

```
           RQNT 1209 endpoint/1@rgw-2567.example.net MGCP 0.1
           X: 0123456789AF
           R: hu
```

The call agent also sends, in parallel, a connection modification request to the firewall, in order to establish a full duplex connection. The SDP payload, in that request, is derived from the parameters of the logical channel for transmission from the gateway to the H.323 entity.

```
           MDCX 1210 relay/17@fw.example.net MGCP 0.1
           C: A3C47F21456789F0
           I: 12345679
           M: sendrecv

           v=0
           c=IN IP4 128.96.63.25
           m=audio 1296 RTP/AVP 0
```

The firewall and the gateway will acknowledge these request:

```
           200 1209 OK

           200 1210 OK
```

At this point, the connection is established.

In our example, the call is terminated when the calling party hangs up. This triggers a Notify command to the call agent:

```
           NTFY 2005 endpoint/1@rgw-2567.example.net MGCP 0.1
           X: 0123456789AF
           O: hu
```

After this notification, the call agent should send an acknowledgement:

```
           200 2005 OK
```

The call agent will then clear the call, by sending a delete connection request to the calling gateway. This request is combined with a notification request, to be ready to detect the next call issued by the residential gateway:

```
           DLCX 1211 endpoint/1@rgw-2567.example.net MGCP 0.1
           C: A3C47F21456789F0
           I: FDE234C8
           X: 0123456789B0
           R: hd
```

The gateway will respond with a message that should include a "call parameters" header field:

```
           250 1211 OK
           P: PS=1245, OS=62345, PR=780, OR=45123, PL=0, JI=11, LA=5
```

In parallel, the call agent also sends two delete connection commands to the firewall, in order to clear the firewall connections:

```
           DLCX 1212 relay/17@fw.example.net MGCP 0.1
           C: A3C47F21456789F0
           I:  12345678
           .
           DLCX 1213 relay/17@fw.example.net MGCP 0.1
           C: A3C47F21456789F0
           I:  12345679
```

The firewall will send reports in response to these commands:

```
           250 1211 OK
           P: PS=780, OS=45123, PR=1245, OR=62345, PL=0, JI=11, LA=5
           .
           250 1211 OK
           P: PS=1245, OS=62345, PR=780, OR=45123, PL=10, JI=27, LA=48
```

The call agent will in parallel sends an H.225/Q.931 "release complete" message to the H.323 entity. It will then tear down the TCP-IP connection. The gateway, at this point, is ready for the next call. The H.323 user will be ready as soon as the H.323 entity notices that the phone is back on hook.

3. Open issues

The preceding call flows demonstrate the adequacy of MGCP to handle "packet to packet" gateways, of which firewalls are a prime example. But the discussions of the packet-relay endpoint leads to three open issues, which may or may not require enhancement to the base protocol:

* How do we handle packet reformating ?

* How do we handle the change of medias, such as going from ATM/AAL1, or ATM/AAL2, to IP/UDP/RTP, and how do we specify which interface should be used ?

* How do we handle several connection simultaneously ?

We will debate these issues in the following paragraphs.

3.1. Packet reformating

Packet reformatting occurs when one side of the packet relay uses a different formatting than the other side, such as G.711 versus G.723.1. MGCP can easily be used to specify that such a transformation is required. For example, in the RGW-to-H.323 example, we could have used different parameters for the two create connection commands sent to the firewall:

```
           CRCX 1205 relay/17@fw.example.net MGCP 0.1
           C: A3C47F21456789F0
           L: p:10, a:PCMU
           M: recvonly

           v=0
           c=IN IP4 10.11.12.13
           m=audio 3456 RTP/AVP 0
           .
           CRCX 1206 relay/17@fw.example.net MGCP 0.1
           C: A3C47F21456789F0
           L: p:10, a:G.723
           M: sendrecv
```

The first command specifies a G.711 encoding with mu-law samples (PCMU, content type 0), while the second specifies a G.723 encoding (content type 4). The firewall, through these commands, is effectively asked to compress the G.711 packets coming from the RGW endpoint and to reformat them as G.723 packets before transmission to the H.323 terminal, and vice versa.

The problem, here, does not lie in the expressiveness of the connection creation and modification commands, but rather in the difficulty of assessing the capabilities of the packet relay:

* When used in pure relay mode, without reformatting, the relay can obviously support any compression algorithm.

* When reformatting is required, the relay needs to be able to execute the algorithm, which requires both knowledge and computing power.

If we wanted to use the protocol to express the capacities of the relay, we would need to distinguish between these two modes.

3.2. Changes of media, multiple interfaces

The change of media is a variation of the packet reformatting problem. It occurs when an end to end connection spans different types of network, such as:

* ATM and IP,

* ATM AAL1 and ATM AAL2,

* Frame-relay and IP,

* Frame-relay and ATM,

* Private and Public IP network.

The problem here is in fact similar to the generic problem of gateways that are connected to several networks: when we create a connection, we have to specify over which network the connection shall be created. There are two extreme ways to solve this problem:

* On a circuit switched network, the network is specified by specifying the circuit over which the connection will be set. In MGCP, this done by specifying an "endpoint."

* On a fully connected packet network, the network is specified through its type, that essentially defines a set of reachable destinations.

The call flows that we presented here fall in the second category. The firewall is asked to set-up "an IP connection", on the assumption that all of the firewall addresses can be reached equally efficiently from the interior and exterior network. That solution may however not be generic enough, and we may need to extend the possibilities of MGCP.

For example, we could have needed to specify that some connections were "interior" and other were "exterior." Today, the only parameter used to specify the outgoing network is the "network type" in the "local connection options". This parameter could be extended to specify not only a type of connectivity but also:

- An addressing domain, such as "IPv4" or "UNI",

- An actual address, or an address prefix, such as "192.96.41.1" or "10.11.12/24",

- In some cases, a port number.

Such extensions could be useful not only to packet relays, but also to the control of large "multi-network" gateways.

3.3. Multiple commands

At several points in our call flows example, we have had to create or delete several connections at the same time. The delete connections requests could always be grouped in a single packet, but in at least one case, the call agent had to wait the answer to a "wildcard" create connection request before a second connection creation request could be issued.

One way to solve this could be to create special purpose commands, that would for example "create two connections" in one command, with the obvious idea that these two commands would work on the same endpoint. But the creation of new types of embedded commands is not very generic.

In fact, we should consider replacing the special purpose embedding facilities, that allow for the encapsulation of a "notification request" or an "endpoint configuration" in a connection creation or modification command, by a more generic mechanism, such as the one suggested by Nancy Greene in an e-mail to the MEGACO list.

Instead of

```
        CRCX 1204 endpoint/1@rgw-2567.whatever.net MGCP 0.1
        C: A3C47F21456789F0
        L: p:10, a:G.711;G.726-32
        M: recvonly
        X: D/0123456789AD
        R: hu
        S:
```

where R: and S: indicate an embedded NotificationRequest, Nancy proposed the following:

```
        CRCX 1204 endpoint/1@rgw-2567.whatever.net MGCP 0.1
        C: A3C47F21456789F0
        L: p:10, a:G.711;G.726-32
        M: recvonly
        Embedded-RQNT
        X: D/0123456789AD
        R: hu
        S:
```

The embedded NotificationRequest is clearly indicated with the line

```
Embedded-RQNT
```

The embedded NotificationRequest shares the same transactionId, endpointId, and MGCP version number as the CreateConnection, so they are not repeated. (Actually we may want to have syntax that would allow a different version number for the embedded RQNT.)

Pushing a good idea one step further, we can extend the concept of embedding so that we can embed arbitrary command, separated by a marker whose syntax would be:

```
        embedded-marker = "Embedded-" MGCPVerb [endpointName]
```

The embedded command would share the same transactionId, and MGCP version number as the main command. If no endpointName is specified, the embedded command would share the endpointName of the main connection. As in the current version of MGCP, the embedding will imply fate sharing: if one command fails, they will all fail - embedding the commands means either they all are executed or none of them are. This would allow us, in our first example, to embed the two create connection commands, as in:

```
        CRCX 1238 relay/$@fw.example.net MGCP 0.1
        C: A3C47F21456789F0
        L: p:10, a:PCMU
        M: sendrecv

        v=0
        c=IN IP4 128.96.41.1
        m=audio 3456 RTP/AVP 0

        Embedded-CRCX
        C: A3C47F21456789F0
        L: p:10, a:PCMU
        M: sendrecv
```

In fact, there may be a need to go one step further than this, and to also allow for the various attribute values (such as endpointName) to be derived from the response of the main command. This would allow us, for example, to specify the endpoint configuration in a "two endpoint" create connection command, as in:

```
        CRCX 1234 trunk/1/$@gw2.example.net
        C: 1234567890abcdef
        M: sendrecv
        Z2: trunk/3/$@gw2.example.net
        Embedded-EPCF =Z
        B: e:A
        Embedded-EPCF =Z2
        B: e:A
```

Obviously, any such modification to the protocol should first be discussed on the MEGACO mailing list.

4. Acknowledgements

We want to thank here the many reviewers who helped design the MGCP flows, notably Ike Elliott and Chip Sharp. The proposal to extend the embedding capabalities of MGCP was suggested by Nancy Greene.

5. References

[0] Arango, M., A. Dugan, I. Elliott, C. Huitema, S. Pickett, "Media Gateway Control Protocol (MGCP)", work in progress.

[1] ITU-T, Recommendation Q.761, "FUNCTIONAL DESCRIPTION OF THE ISDN USER PART OF SIGNALLING SYSTEM No. 7", (Malaga-Torremolinos, 1984; modified at Helsinki, 1993)

[2] ITU-T, Recommendation Q.762, "GENERAL FUNCTION OF MESSAGES AND SIGNALS OF THE ISDN USER PART OF SIGNALLING SYSTEM No. 7", (Malaga-Torremolinos, 1984; modified at Helsinki, 1993)

*   ITU-T, Recommendation H.323, "VISUAL TELEPHONE SYSTEMS AND EQUIPMENT FOR LOCAL AREA NETWORKS WHICH PROVIDE A NON-GUARANTEED QUALITY OF SERVICE."

*   ITU-T, Recommendation H.225, "Call Signaling Protocols and Media Stream Packetization for Packet Based Multimedia Communications Systems."

- * ITU-T, Recommendation H.245, "LINE TRANSMISSION OF NON-TELEPHONE SIGNALS."

- * Handley, Shulzrinne, Schooler, Rosenberg, "SIP: Session Initiation Protocol", work in progress

6. Authors' Addresses

Christian Huitema
Bellcore
MCC 1J236B
445 South Street
Morristown, NJ 07960
U.S.A.

Phone: +1 973-829-4266
EMail: huitema@bellcore.com

Flemming Andreasen
Bellcore
RRC-1M223
444 Hoes Lane
Piscataway, NJ 08854-4157
U.S.A.

Phone: +1 732 699-7351
EMail: fandreas@notes.cc.bellcore.com