

PATCH 1

Patch Asterisk per gestione hangup nei trasferimenti

di seguito i risultati del diff dei seguenti file:

- diff -cB channels/chan_sip.c.orig channels/chan_sip.c > /root/chan_sip.c.diff
- diff -cB main/channel.c.orig main/channel.c > /root/channel.c
- diff -cB include/asterisk/channel.h.orig include/asterisk/channel.h > /root/channel.h

chan_sip.c.diff

```
*** chan_sip.c.orig 2010-02-04 15:42:02.000000000 +0100
--- chan_sip.c 2010-02-04 15:43:52.000000000 +0100
*****
*** 14127,14134 ****
    /* Make sure that the masq does not free our PVT for the old call */
    if (! earlyreplace && ! oneleggedreplace )
    !     ast_set_flag(&p->refer->refer_call->flags[0],
SIP_DEFER_BYE_ON_TRANSFER);    /* Delay hangup */
    !
    /* Prepare the masquerade - if this does not happen, we will be gone */
    if(ast_channel_masquerade(replacecall, c))
        ast_log(LOG_ERROR, "Failed to masquerade C into Replacecall\n");
--- 14127,14134 ----
    /* Make sure that the masq does not free our PVT for the old call */
    if (! earlyreplace && ! oneleggedreplace )
    !     /* ast_set_flag(&p->refer->refer_call->flags[0],
SIP_DEFER_BYE_ON_TRANSFER);*/
    !     ast_set_flag(&p->refer->refer_call->flags[0],
AST_FLAG_KEEP_PVT_AT_HANGUP);
    /* Prepare the masquerade - if this does not happen, we will be gone */
    if(ast_channel_masquerade(replacecall, c))
        ast_log(LOG_ERROR, "Failed to masquerade C into Replacecall\n");
```

channel.c

```
*** channel.c.orig 2010-02-04 15:39:11.000000000 +0100
--- channel.c 2010-02-04 15:41:43.000000000 +0100
*****
*** 1270,1277 ****
    ast_channel_unlock(chan);
    if (chan->tech_pvt) {
    !     ast_log(LOG_WARNING, "Channel '%s' may not have been hung up
properly\n", chan->name);
    !     free(chan->tech_pvt);
    }
    if (chan->sched)
```

```

--- 1270,1283 ----
    ast_channel_unlock(chan);
    if (chan->tech_pvt) {
!       /* ast_log(LOG_WARNING, "Channel '%s' may not have been hung up
properly\n", chan->name);
!       free(chan->tech_pvt); */
!       if (ast_test_flag(chan, AST_FLAG_KEEP_PVT_AT_HANGUP)) {
!           ast_log(LOG_DEBUG, "Channel '%s' pvt stays around, even
though the channel is freed.\n", chan->name);
!           } else {
!               ast_log(LOG_WARNING, "Channel '%s' may not have been hung up
properly\n", chan->name);
!               free(chan->tech_pvt);
!           }
    }
    if (chan->sched)

```

channel.h

```

*** channel.h.orig  2010-02-04 15:35:29.000000000 +0100
--- channel.h      2010-02-04 15:37:41.000000000 +0100
*****
*** 496,501 ****
--- 496,503 ----
    AST_FLAG_OUTGOING =      (1 << 10),
    /*! This channel is being whispered on */
    AST_FLAG_WHISPER =      (1 << 11),
+   /*! Do NOT destroy PVT even if it's around after hangup! */
+   AST_FLAG_KEEP_PVT_AT_HANGUP = (1 << 12),
    /*! A DTMF_BEGIN frame has been read from this channel, but not yet an
END */
    AST_FLAG_IN_DTMF =      (1 << 12),
    /*! A DTMF_END was received when not IN_DTMF, so the length of the digit
is

```

PATCH NUMERO 2

Patch per la gestione del Q.SIG e dei trasferimenti su euroisdn

Nel file **libpri/pri_facility.h**

cambiare il valore di ROSE_DIVERTING_LEG_INFORMATION2 da 0x15 a 0x0f

Q.SIG (USA) usa il valore HEX 0x15 per la funzione, mentre euroisdn (almeno il nostro) HEX 0x0f (15 DEC)

From:

<https://wiki.unitn.it/> - **Wiki UniTn**

Permanent link:

<https://wiki.unitn.it/pub:patch-asterisk>



Last update: **2010/05/05 14:23**