

E&M 디지털 CAS 시그널링을 위한 EM_PARK 문제 해결

목차

[소개](#)

[사전 요구 사항](#)

[요구 사항](#)

[사용되는 구성 요소](#)

[배경 이론](#)

[표기 규칙](#)

[문제](#)

[솔루션](#)

[가짜 대담](#)

[관련 정보](#)

소개

Cisco 2600, 3600 및 MC3810 라우터 플랫폼의 디지털 E&M 시그널링에서는 일부 T1/E1 시간 슬롯이 EM_PARK 상태에서 중단될 수 있습니다. 이는 **show voice call summary** 명령을 실행할 때 표시됩니다. 이 문서에서는 이 문제를 해결하는 방법에 대해 설명합니다.

이 출력은 일부 시간 슬롯이 EM_PARK 상태에 있음을 보여줍니다. EM_PARK 상태의 시간 슬롯은 음성 통화에 사용되지 않습니다.

```
Router#show voice call summary
PORT      CODEC      VAD      VTSP STATE      VPM STATE
=====
1/0:0.1   -         -         -               EM_ONHOOK
1/0:0.2   -         -         -               EM_PARK
1/0:0.3   -         -         -               EM_PARK
1/0:0.4   -         -         -               EM_ONHOOK
1/0:0.5   -         -         -               EM_ONHOOK
```

사전 요구 사항

요구 사항

이 문서에 대한 특정 요건이 없습니다.

사용되는 구성 요소

이 문서의 정보는 다음 소프트웨어 및 하드웨어 버전을 기반으로 합니다.

- 하드웨어—Cisco 2600, Cisco 3600, Cisco VG200 및 MC3810 라우터
- 소프트웨어 - 모두

이 문서의 정보는 특정 랩 환경의 디바이스를 토대로 작성되었습니다. 이 문서에 사용된 모든 디바이스는 초기화된(기본) 컨피그레이션으로 시작되었습니다. 현재 네트워크가 작동 중인 경우, 모든 명령어의 잠재적인 영향을 미리 숙지하시기 바랍니다.

배경 이론

예를 들어, T1 CAS에서 링크 시작 신호, PBX가 오프후크되면 원격 대상에서 통화에 응답할 때까지 라우터/게이트웨이 측 시간 슬롯이 유희(EM_ONHOOK) 상태로 유지됩니다. 원격 대상에서 통화에 응답하면 라우터 시간 슬롯 상태가 EM_OFFHOOK로 변경됩니다.

통화가 연결되지 않으면 라우터/게이트웨이가 인밴드 재주문음을 발신자에게 재생합니다. 라우터 측의 채널 상태가 여전히 EM_ONHOOK이므로 라우터가 채널을 끊을 수 없습니다. 발신자가 전화를 끊은 후 PBX는 채널 상태를 오프후크에서 온후크로 변경해야 합니다.

PBX는 전환전환의 도움을 받아 온후크 메시지를 보내지 않는 경우가 있습니다. 라우터에는 이러한 "가짜 대답"에 대한 해결 방법이 있습니다. 거짓 응답 방법이 없으면 채널이 무기한 EM_PARK 상태로 정지됩니다. 자세한 내용은 [가짜 응답](#) 섹션을 참조하십시오.

참고: 음성 게이트웨이 라우터 새시가 올바르게 전기로 접지되지 않은 경우 일부 T1 채널의 EM_PARK 상태에서 통화를 보류할 수 있습니다. 전기 접지에 대한 자세한 내용은 하드웨어 설치 가이드를 참조하십시오.

표기 규칙

문서 규칙에 대한 자세한 내용은 [Cisco 기술 팁 표기 규칙을 참고하십시오](#).

문제

시간 슬롯이 EM_PARK 상태에 머물러 있는 주된 이유는 두 가지입니다.

- DSP(Digital Signal Processor)가 잘못되어 하드웨어 또는 소프트웨어 문제가 발생했습니다.
- PSTN 스위치/PBX는 라우터에 지속적인 오프후크 신호를 전송하며 이를 해제하지 않습니다.

솔루션

이 문제의 해결 방법은 다음과 같습니다.

시스템의 시간 슬롯이 EM_PARK 상태로 고정되어 있으면 DSPs를 확인합니다. DSP를 확인하려면 [Cisco 2600/3600 Series 라우터용 NM-HDV의 DSP 문제 해결](#)을 참조하십시오.

DSP가 활성 상태이면 PSTN 스위치/PBX 또는 Cisco IOS®에 문제가 있을 수 있습니다(라우터/게이트웨이가 가짜 응답 절차를 시작하지 않음). 자세한 내용은 [가짜 응답](#) 섹션을 참조하십시오.

가짜 대답

Cisco 라우터/게이트웨이는 재주문음을 재생하는 동안 PBX에서 온후크로 시간 슬롯을 설정해야 한다는 사실을 알고 나면 기본값인 30초([timeouts wait-release](#) 및 [timeouts call-disconnect](#) 명령을 사

용하여 이 값을 변경합니다) 기다립니다.

이 문제가 발생하지 않으면 라우터는 시간 슬롯을 EM_PARK 상태로 이동하고 10초의 기간 동안 다른 타이머를 시작합니다. 10초 후에도 PBX가 계속 고착되지 않으면 라우터가 PBX를 재사용합니다. 라우터는 1초 동안 가짜 응답을 전송한 다음 후크합니다.

라우터가 가짜 응답 신호를 전송하면 라우터는 5분의 또 다른 타이머를 시작합니다. PBX가 온후크 상태가 되면 타이머가 중지되고 라우터가 시간 슬롯을 EM_ONHOOK 상태로 전환합니다. 그렇지 않으면, 5분 후에 1초라는 가짜 응답 신호를 보냅니다. 라우터는 PBX가 온후크 상태가 될 때까지 이 프로세스를 반복합니다. 라우터가 PBX에 강제로 통화를 지웁니다.

참고: 이 응답 전환은 실제 통화가 지워지므로 계정 레코드로 업데이트되지 않습니다. 그러나 PBX는 이를 하나의 대답으로 이해하며 사용자는 1초 통화에 대해 요금이 부과될 수 있습니다.

EM_PARK 상태의 시간 슬롯과 연결된 DSP가 활성 상태이고 문제가 지속되면 `debug vpm all 및 debug vtsp all` 명령을 실행하여 Cisco IOS가 가짜 응답을 보내려고 하는지 확인합니다.

참고: 디버그를 5분 이상 실행해야 합니다.

참고: 대부분의 경우 DSP가 불량일 경우 라우터가 가짜 응답 해결 방법을 수행하지 않습니다. 자세한 내용은 [Cisco 2600/3600 Series 라우터용 NM-HDV의 DSP 문제 해결](#)을 참조하십시오.

이 디버그 출력은 EM_PARK에서 시간 슬롯이 고정되는 방법과 가짜 응답 해결 방법이 작동하는 방식을 보여줍니다.

```
Jan 11 17:19:00.767: htsp_dsp_message: SEND/RESP_SIG_STATUS: state=0xC timestamp
=44262 systime=31305235
Jan 11 17:19:00.767: htsp_process_event:
[4/1:1(10), EM_ONHOOK, E_DSP_SIG_1100]em_onhook_offhook htsp_setup_ind
!--- Offhook signal is received from the switch. Jan 11 17:19:00.767: [4/1:1(10)]
get_local_station_id calling num= calling name= calling time=01/11 17:19 Jan 11 17:19:00.767:
vtsp_tsp_call_setup_ind (sdb=0x62BB7B14, tdm_info=0x0, tsp_info=0x62BB4050, calling_number=
calling_oct3 = 0x0, called_number= called_oct3 = 0x81, oct3a=0x0): peer_tag=0 Jan 11
17:19:00.767: : ev.clg.clir is 0 ev.clg.clid-transparent is 0 ev.clg.null_orig_clg is 1
ev.clg.calling_translated is false Jan 11 17:19:00.767: htsp_timer - 3000 msec Jan 11
17:19:00.767: vtsp_do_call_setup_ind Jan 11 17:19:00.767: vtsp_allocate_cdb,cdb 0x62DCEA70 Jan
11 17:19:00.767: vtsp_do_call_setup_ind: Call ID=112722, guid=62DC4230 Jan 11 17:19:00.767:
vtsp_do_call_setup_ind: type=0, under_spec=1640890368, name=, id0=10, id1=1, id2=25038,
calling=, called= subscriber=RegularLine Jan 11 17:19:00.767: vtsp_do_normal_call_setup_ind Jan
11 17:19:00.771: cc_api_call_setup_ind (vdbPtr=0x62BB7FA0, callInfo={called=
,called_oct3=0x81,calling=,calling_oct3=0x0,calling_oct3a=0x0,calling_xlated=fal
se,subscriber_type_str=RegularLine,fdest=0,peer_tag=0, prog_ind=3},callID=0x62DC 40DC) Jan 11
17:19:00.771: cc_api_call_setup_ind type 1 , prot 0 Jan 11 17:19:00.771: vtsp_insert_cdb,cdb
0x62DCEA70 Jan 11 17:19:00.771: vtsp_open_voice_and_set_params Jan 11 17:19:00.771:
dsp_close_voice_channel: [4/1:1:32995] packet_len=8 channel_id=3 packet_id=75 Jan 11
17:19:00.771: dsp_open_voice_channel_20: [4/1:1:32995] packet_len=16 channel_id=3 packet_id=74
alaw_ulaw_select=0 associated_signaling_channel=130 time_slot=2 serial_port=0 Jan 11
17:19:00.771: vtsp_modem_proto_from_cdb: cap_modem_proto 1073741824 Jan 11 17:19:00.771:
vtsp_modem_proto_from_cdb: cap_modem_proto 1073741824 Jan 11 17:19:00.771: dsp_encap_config:
[4/1:1:32995] packet_len=30 channel_id=3 packet_id=92 TransportProtocol 2 t_ssrc=0x0 r_ssrc=0x0
t_vpxcc=0x0 r_vpxcc=0x0 sid_support=1, tse_payload=65535, seq_num=0x0, redundancy=0 Jan 11
17:19:00.771: dsp_set_playout_delay Jan 11 17:19:00.771: dsp_set_playout: [4/1:1:32995]
packet_len=18 channel_id=3 packet_id=76 mode=1 initial=60 min=40 max=200 fax_nom=300
dsp_set_playout_delay_config Jan 11 17:19:00.771: dsp_set_playout_config Jan 11 17:19:00.771:
mode 0, init 60, min 40, max 200 playout default Jan 11 17:19:00.771:
dsp_set_playout_config:mode 0, init 60, min 40, max 200 Jan 11 17:19:00.771:
```

dsp_set_payout_config: [4/1:1:32995] packet_len=18 channel_id=3 packet_id=76 mode=1 initial=60 min=40 max=200 fax_nom=300 Jan 11 17:19:00.771: dsp_echo_canceler_control: echo_cancel: 1 Jan 11 17:19:00.771: dsp_echo_canceler_control: [4/1:1:32995] echo_cancel 1, disable_hpf 0, flags=0x0, threshold=-21 Jan 11 17:19:00.771: dsp_echo_canceler_control: [4/1:1:32995] packet_len=12 channel_id=3 packet_id=66 flags=0x0, threshold=-21 Jan 11 17:19:00.771: set_gains: FXx/E&M: msg->message.set_codec_gains.out_gain=0 Jan 11 17:19:00.771: dsp_set_gains: [4/1:1:32995] packet_len=12 channel_id=3 packet_id=91 in_gain=0 out_gain=0 Jan 11 17:19:00.771: dsp_vad_enable: [4/1:1:32995] enable: packet_len=12 channel_id=3 packet_id=78 thresh=-38 Jan 11 17:19:00.771: cc_process_call_setup_ind (event=0x62E63ACC) Jan 11 17:19:00.771: >>>>CCAPI handed cid 32995 with tag 0 to app "DEFAULT" Jan 11 17:19:00.771: sess_appl: ev(24=CC_EV_CALL_SETUP_IND), cid(32995), disp(0) Jan 11 17:19:00.771: sess_appl: ev(SSA_EV_CALL_SETUP_IND), cid(32995), disp(0) Jan 11 17:19:00.771: ssaCallSetupInd Jan 11 17:19:00.771: ccCallSetContext (callID=0x80E3, context=0x62DFBCF0) Jan 11 17:19:00.771: ssaCallSetupInd cid(32995), st(SSA_CS_MAPPING),oldst(0), ev (24)ev->e.evCallSetupInd.nCallInfo.finalDestFlag = 0 Jan 11 17:19:00.771: ccCallSetupAck (callID=0x80E3) Jan 11 17:19:00.771: ccGenerateTone (callID=0x80E3 tone=8) Jan 11 17:19:00.771: ccCallReportDigits (callID=0x80E3, enable=0x1) Jan 11 17:19:00.771: vtsp_report_digit_control: enable=1: digit reporting enabled Jan 11 17:19:00.771: cc_api_call_report_digits_done (vdbPtr=0x62BB7FA0, callID=0x80E3, disp=0) Jan 11 17:19:00.771: : vtsp_get_digit_timeouts Jan 11 17:19:00.771: sess_appl: ev(52=CC_EV_CALL_REPORT_DIGITS_DONE), cid(32995), disp(0) Jan 11 17:19:00.771: cid(32995)st(SSA_CS_MAPPING)ev (SSA_EV_CALL_REPORT_DIGITS_DONE) oldst(SSA_CS_MAPPING)cfid(-1)csz(0)in(1)fDest(0) Jan 11 17:19:00.771: ssaReportDigitsDone cid(32995) peer list: (empty) Jan 11 17:19:00.771: ssaReportDigitsDone callid=32995 Enable succeeded Jan 11 17:19:00.771: ccGenerateTone (callID=0x80E3 tone=8) Jan 11 17:19:00.771: vtsp:[4/1:1:32995, S_SETUP_INDICATED, E_CC_SETUP_ACK] Jan 11 17:19:00.775: act_setup_ind_ack Jan 11 17:19:00.775: vtsp_modem_proto_from_cdb: cap_modem_proto 0 Jan 11 17:19:00.775: vtsp_modem_proto_from_cdb: cap_modem_proto 0 Jan 11 17:19:00.775: dsp_encap_config: [4/1:1:32995] packet_len=30 channel_id=3 packet_id=92 TransportProtocol 2 t_ssrc=0x0 r_ssrc=0x0 t_vpxcc=0x0 r_vpxcc=0x0 sid_support=1, tse_payload=65535, seq_num=0x0, redundancy=0 Jan 11 17:19:00.775: dsp_voice_mode: [4/1:1:32995] cdb 62DCEA70, cdb->codec_params.modem 2, inband_detect flags 0x21 Jan 11 17:19:00.775: map_dtmf_relay_type--digit relay mode: 2 Jan 11 17:19:00.775: dsp_voice_mode: [4/1:1:32995] packet_len=24 channel_id=3 packet_id=73 coding_type=1 voice_field_size=160 VAD_flag=0 echo_length=256 comfort_noise=1 inband_detect=33 digit_relay_mode=2 AGC_flag=0act_setup_ind_ack: modem_mode = 0, fax_relay_on = 1 Jan 11 17:19:00.775: act_setup_ind_ack(): dsp_dtmf_mode() dsp_dtmf_mode(VTSP_TONE_DTMF_MODE) Jan 11 17:19:00.775: dsp_dtmf_mode: [4/1:1:32995] packet_len=10 channel_id=3 packet_id=65 dtmf_or_mf=0 Jan 11 17:19:00.775: vtsp_timer: 31305236 Jan 11 17:19:00.775: vtsp:[4/1:1:32995, S_DIGIT_COLLECT, E_CC_GEN_TONE] Jan 11 17:19:00.775: act_gen_tone Jan 11 17:19:00.775: dsp_cp_tone_off: [4/1:1:32995] packet_len=8 channel_id=3 packet_id=71 Jan 11 17:19:00.775: dsp_cp_tone_on: [4/1:1:32995] packet_len=38 channel_id=3 packet_id=72 tone_id=4 n_freq=2 freq_of_first=350 freq_of_second=440 amp_of_first=5514 amp_of_second=5514 direction=1 on_time_first=65535 off_time_first=0 on_time_second=0 off_time_second=0 Jan 11 17:19:00.775: vtsp:[4/1:1:32995, S_DIGIT_COLLECT, E_CC_GEN_TONE] Jan 11 17:19:00.775: act_gen_tone Jan 11 17:19:00.775: dsp_cp_tone_off: [4/1:1:32995] packet_len=8 channel_id=3 packet_id=71 Jan 11 17:19:00.775: dsp_cp_tone_on: [4/1:1:32995] packet_len=38 channel_id=3 packet_id=72 tone_id=4 n_freq=2 freq_of_first=350 freq_of_second=440 amp_of_first= 5514 amp_of_second=5514 direction=1 on_time_first=65535 off_time_first=0 on_time4_second=0 off_time_second=0 Jan 11 17:19:00.775: htsp_process_event: [4/1:1(10), EM_WAIT_SETUP_ACK, E_HTSP_SETUP_ACK]em_wait_setup_ack_get_ack Jan 11 17:19:00.775: htsp_timer_stop Jan 11 17:19:00.775: htsp_timer2 - 172 msec Jan 11 17:19:00.947: htsp_process_event: [4/1:1(10), EM_WAIT_SETUP_ACK, E_HTSP_EVENT_TIMER2]em_wait_prewink_timer **Jan 11 17:19:00.947: em_offhook (0)[recEive and transMit4/1:1(10)] set signal st**
ate = 0x8em_onhook (200)[recEive and transMit4/1:1(10)] set signal state = 0x0
!--- A wink of duration 200 msec is sent out to the switch. Jan 11 17:19:01.471: vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_BEGIN: digit=9, rtp_timestamp=0xED31C493 Jan 11 17:19:01.471: vtsp:[4/1:1:32995, S_DIGIT_COLLECT, E_DSP_DTMF_DIGIT_BEGIN] Jan 11 17:19:01.471: act_report_digit_begin Jan 11 17:19:01.471: cc_api_call_digit_begin (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF F, srcCallId=0x80E3, digit=9, digit_begin_flags=0x1, rtp_timestamp=0xED31C493 rtp_expiration=0x0, dest_mask=0x1) Jan 11 17:19:01.471: sess_appl: ev(10=CC_EV_CALL_DIGIT_BEGIN), cid(32995), disp(0) Jan 11 17:19:01.471: cid(32995)st(SSA_CS_MAPPING)ev(SSA_EV_DIGIT_BEGIN) oldst(SSA_CS_MAPPING)cfid(-1)csz(0)in(1)fDest(0) Jan 11 17:19:01.471: ssaIgnore cid(32995), st(SSA_CS_MAPPING),oldst(0), ev(10) Jan 11 17:19:01.503: vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_OFF: digit=9, duration=65 Jan 11 17:19:01.503: vtsp:[4/1:1:32995, S_DIGIT_COLLECT, E_DSP_DTMF_DIGIT] Jan 11

17:19:01.503: act_report_digit_end Jan 11 17:19:01.503: vtsp_timer_stop: 31305308 Jan 11
17:19:01.503: dsp_cp_tone_off: [4/1:1:32995] packet_len=8 channel_id=3 pa cket_id=71 Jan 11
17:19:01.503: cc_api_call_digit_end (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF, srcCallId=0x80E3,
digit=9,duration=65,xruleCallingTag=0,xruleCalledTag=0, dest_mask=0x1), digi t_tone_mode=0 Jan
11 17:19:01.503: htsp_digit_ready: digit = 39 Jan 11 17:19:01.503: vtsp_timer: 31305308 Jan 11
17:19:01.503: htsp_process_event: [4/1:1(10), EM_OFFHOOK, E_VTSP_DIGIT]em_offhook_digit_collect
Jan 11 17:19:01.503: sess_appl: ev(9=CC_EV_CALL_DIGIT_END), cid(32995), disp(0) Jan 11
17:19:01.503: cid(32995)st(SSA_CS_MAPPING)ev(SSA_EV_CALL_DIGIT) oldst(SSA_CS_MAPPING)cfid(-
1)csz(0)in(1)fDest(0) Jan 11 17:19:01.503: ssaDigit Jan 11 17:19:01.503: ssaDigit, 0. sct-
>digit , sct->digit len 0, usrDigit 9, digit_tone_mode=0 Jan 11 17:19:01.503: ssaDigit,1.
callinfo.called , digit 9, callinfo.calling , x rulecallingtag 0, xrulecalledtag 0 Jan 11
17:19:01.503: ssaDigit, 7. callinfo.calling , sct->digit 9, result 1 Jan 11 17:19:01.603:
vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_BEGIN: digit=1, rtp_timestamp=0xED31C493 Jan 11
17:19:01.603: vtsp:[4/1:1:32995, S_DIGIT_COLLECT, E_DSP_DTMF_DIGIT_BEGIN] Jan 11 17:19:01.603:
act_report_digit_begin Jan 11 17:19:01.603: cc_api_call_digit_begin (dstVdbPtr=0x0,
dstCallId=0xFFFFFFFF F, srcCallId=0x80E3, digit=1, digit_begin_flags=0x1,
rtp_timestamp=0xED31C493 rtp_expiration=0x0, dest_mask=0x1) Jan 11 17:19:01.603: sess_appl:
ev(10=CC_EV_CALL_DIGIT_BEGIN), cid(32995), disp(0) Jan 11 17:19:01.603:
cid(32995)st(SSA_CS_MAPPING)ev(SSA_EV_DIGIT_BEGIN) oldst(SSA_CS_MAPPING)cfid(-
1)csz(0)in(1)fDest(0) Jan 11 17:19:01.603: ssaIgnore cid(32995), st(SSA_CS_MAPPING),oldst(0),
ev(10) Jan 11 17:19:01.643: vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_OFF: digit=1,
duration=75 Jan 11 17:19:01.643: vtsp:[4/1:1:32995, S_DIGIT_COLLECT, E_DSP_DTMF_DIGIT] Jan 11
17:19:01.643: act_report_digit_end Jan 11 17:19:01.643: vtsp_timer_stop: 31305322 Jan 11
17:19:01.643: cc_api_call_digit_end (dstVdbPtr=0x0, dstCallId=0xFFFFFFFF, srcCallId=0x80E3,
digit=1,duration=75,xruleCallingTag=0,xruleCalledTag=0, dest_mask=0x1), digit_tone_mode=0 Jan 11
17:19:01.643: htsp_digit_ready: digit = 31 Jan 11 17:19:01.643: vtsp_timer: 31305322 Jan 11
17:19:01.643: htsp_process_event: [4/1:1(10), EM_OFFHOOK, E_VTSP_DIGIT]em_offhook_digit_collect
Jan 11 17:19:01.643: sess_appl: ev(9=CC_EV_CALL_DIGIT_END), cid(32995), disp(0) Jan 11
17:19:01.643: cid(32995)st(SSA_CS_MAPPING)ev(SSA_EV_CALL_DIGIT) oldst(SSA_CS_MAPPING)cfid(-
1)csz(0)in(1)fDest(0) Jan 11 17:19:01.643: ssaDigit Jan 11 17:19:01.643: ssaDigit, 0. sct-
>digit 9, sct->digit len 1, usrDigit 1, digit_tone_mode=0 Jan 11 17:19:01.643: ssaDigit,1.
callinfo.called , digit 91, callinfo.calling , xrulecallingtag 0, xrulecalledtag 0 Jan 11
17:19:01.643: ssaDigit, 7. callinfo.calling , sct->digit 91, result 1 Jan 11 17:19:01.743:
vtsp_process_dsp_message: MSG_TX_DTMF_DIGIT_BEGIN: digit=8, rtp_timestamp=0xED31C493 Jan 11
17:19:01.743: vtsp:[4/1:1:32995, S_DIGIT_COLLECT, E_DSP_DTMF_DIGIT_BEGIN] Jan 11 17:19:01.743:
act_report_digit_begin Jan 11 17:19:01.743: cc_api_call_digit_begin (dstVdbPtr=0x0,
dstCallId=0xFFFFFFFF F, srcCallId=0x80E3, digit=8, digit_begin_flags=0x1,
rtp_timestamp=0xED31C493 rtp_expiration=0x0, dest_mask=0x1) Jan 11 17:19:01.743: sess_appl:
ev(10=CC_EV_CALL_DIGIT_BEGIN), cid(32995), disp(0) Jan 11 17:19:01.743:
cid(32995)st(SSA_CS_MAPPING)ev(SSA_EV_DIGIT_BEGIN) oldst(SSA_CS_MAPPING)cfid(-
1)csz(0)in(1)fDest(0) Jan 11 17:19:01.743: ssaIgnore cid(32995), st(SSA_CS_MAPPING),oldst(0),
ev(10) radius_decrypt: null length Jan 11 17:19:01.843: vtsp_process_dsp_message:
MSG_TX_DTMF_DIGIT_OFF: digit=8, duration=75 Jan 11 17:19:01.843: vtsp:[4/1:1:32995,
S_DIGIT_COLLECT, E_DSP_DTMF_DIGIT] Jan 11 17:19:01.843: act_report_digit_end Jan 11
17:19:01.843: vtsp_timer_stop: 31305342 Jan 11 17:19:01.843: cc_api_call_digit_end
(dstVdbPtr=0x0, dstCallId=0xFFFFFFFF, srcCallId=0x80E3,
digit=8,duration=75,xruleCallingTag=0,xruleCalledTag=0, dest_mask=0x1), digi t_tone_mode=0 Jan
11 17:19:01.843: htsp_digit_ready: digit = 38 Jan 11 17:19:01.843: vtsp_timer: 31305342 Jan 11
17:19:01.843: htsp_process_event: [4/1:1(10), EM_OFFHOOK, E_VTSP_DIGIT]em_offhook_digit_collect
Jan 11 17:19:01.843: sess_appl: ev(9=CC_EV_CALL_DIGIT_END), cid(32995), disp(0) Jan 11
17:19:01.843: cid(32995)st(SSA_CS_MAPPING)ev(SSA_EV_CALL_DIGIT) oldst(SSA_CS_MAPPING)cfid(-
1)csz(0)in(1)fDest(0) Jan 11 17:19:01.843: ssaDigit Jan 11 17:19:01.843: ssaDigit, 0. sct-
>digit 91, sct->digit len 2, usrDigit 8, d igit_tone_mode=0 Jan 11 17:19:01.843: ssaDigit,1.
callinfo.called , digit 918, callinfo.calling , xrulecallingtag 0, xrulecalledtag 0 Jan 11
17:19:01.843: ssaDigit, 7. callinfo.calling , sct->digit 918, result -1 Jan 11 17:19:01.843:
ccCallDisconnect (callID=0x80E3, cause=0x1C tag=0x0) Jan 11 17:19:01.843: vtsp:[4/1:1:32995,
S_DIGIT_COLLECT, E_CC_DISCONNECT] Jan 11 17:19:01.843: act_pre_con_disconnect Jan 11
17:19:01.843: vtsp_ring_noan_timer_stop: 31305342 Jan 11 17:19:01.843: dsp_cp_tone_off:
[4/1:1:32995] packet_len=8 channel_id=3 pa cket_id=71 Jan 11 17:19:01.843: dsp_voice_mode:
[4/1:1:32995] cdb 62DCEA70, cdb->codec_para ms.modem 2, inband_detect flags 0x21 Jan 11
17:19:01.843: map_dtmf_relay_type--digit relay mode: 2 Jan 11 17:19:01.843: dsp_voice_mode:
[4/1:1:32995] packet_len=24 channel_id=3 pa cket_id=73 coding_type=1 voice_field_size=160
VAD_flag=0 echo_length=256 comfort_noise=1 inband_detect=33 digit_relay_mode=2 AGC_flag=0 Jan
11 17:19:01.843: **dsp_cp_tone_on: [4/1:1:32995] packet_len=38 channel_id=3 pa**

```

cket_id=72 tone_id=3 n_freq=2 freq_of_first=480 freq_of_second=620amp_of_first=
5206 amp_of_second=2928 direction=1 on_time_first=250 off_time_first=250
on_time_second=0 off_time_second=0
Jan 11 17:19:01.843: vtsp_timer: 31305342
Jan 11 17:19:01.843: htsp_pre_connect_disconnect, cdb = 62DCEA70 cause = 1C
!--- Since the call is disconnected because the number received is "unassigned" !--- or
"invalid" the router starts to play the reorder !--- tone and a timer, which is the wait-release
!--- timeout timer, starts with default 30 seconds. !--- This call is disconnected !--- prior to
the connect state. Jan 11 17:19:01.843: htsp_process_event: [4/1:1(10), EM_OFFHOOK,
E_HTSP_PRE_CONN_DISC] Jan 11 17:19:31.844: vtsp_main: timer: 31308342
!--- The wait-release timer expires after 30 seconds. Jan 11 17:19:31.844: vtsp:[4/1:1:32995,
S_WAIT_RELEASE_NC, E_TIMER]
!--- The VTSP module is in a wait release state for that call. It also receives !--- event
timer, which means that the timer expires so that it !--- goes into another state. Jan 11
17:19:31.844: act_pre_con_disc_rel htsp_release_req: cause 28, no_onhook 0 Jan 11 17:19:31.844:
htsp_process_event: [4/1:1(10), EM_OFFHOOK,
E_HTSP_RELEASE_REQ]em_offhook_release
Jan 11 17:19:31.844: htsp_timer_stop2 em_onhook (0)[recEive and transMit4/1:1(10
)] set signal state = 0x0
Jan 11 17:19:31.844: htsp_timer_stop
Jan 11 17:19:31.844: em_start_timer: 400 ms
Jan 11 17:19:31.844: htsp_timer - 400 msec
!--- HTSP receives an event that requests the release of !--- the time slot and it goes into EM
wait !--- onhook state. But, it cannot do anything since it says I am onhook already. !--- Also,
the router starts a timer of 400 msec. Jan 11 17:19:32.296: htsp_process_event: [4/1:1(10),
EM_WAIT_ONHOOK,
E_HTSP_EVENT_TIMER]em_wait_timeout
Jan 11 17:19:32.296: em_stop_timers
Jan 11 17:19:32.296: htsp_timer_stop
Jan 11 17:19:32.296: em_start_timer: 400 ms
Jan 11 17:19:32.296: htsp_timer - 400 msec
!--- When the 400 msec timer expires, HTSP gets into EM clear pending state. !--- It also starts
another timer of 400 msec. Jan 11 17:19:32.696: htsp_process_event: [4/1:1(10), EM_CLR_PENDING,
E_HTSP_EVENT_TIMER]em_clr_timeout Jan 11 17:19:32.696: em_stop_timers Jan 11 17:19:32.696:
htsp_timer_stop Jan 11 17:19:32.696: em_start_timer: 10000 ms Jan 11 17:19:32.696: htsp_timer -
10000 msec Jan 11 17:19:32.700: htsp_dsp_message: SEND/RESP_SIG_STATUS: state=0xC timestamp=1533
systime=31308428 Jan 11 17:19:32.700: htsp_process_event: [4/1:1(10), EM_PARK,
E_DSP_SIG_1100]em_park_offhook !--- When the 400 msec timer expires, the router puts the time
slot into !--- the EM_PARK state, and it starts another timer of 10 seconds. !--- The router
still sees the ABCD=1100 from the switch. Jan 11 17:19:42.760: htsp_process_event: [4/1:1(10),
EM_PARK, E_HTSP_EVENT_TIMER]em_park_timerhtsp_report_onhook_sig
Jan 11 17:19:42.760: em_offhook (0)[recEive and transMit4/1:1(10)] set signal st
ate = 0x8em_onhook (1000)[recEive and transMit4/1:1(10)] set signal state = 0x0
Jan 11 17:19:42.760: htsp_timer2 - 300000 msec
Jan 11 17:19:42.760: htsp_process_event: [4/1:1(10),
EM_PARK, E_HTSP_EVENT_TIMER]em_park_timerhtsp_report_onhook_sig
Jan 11 17:19:42.760: em_offhook (0)[recEive and transMit4/1:1(10)]
set signal state = 0x8em_onhook (1000)[recEive and
transMit4/1:1(10)] set signal state = 0x0
Jan 11 17:19:42.760: htsp_timer2 - 300000 msec
!--- As seen from the timestamps, when the timer expires in ten seconds, !--- the router goes
offhook for one second (1000 msec) and then onhook. !--- It also starts another timer of 300000
msec (5 minutes).

```

관련 정보

- [음성 기술 지원](#)
- [음성 및 통합 커뮤니케이션 제품 지원](#)
- [Cisco IP 텔레포니 문제 해결](#)
- [기술 지원 및 문서 - Cisco Systems](#)