Fallas de llamada del Troubleshooting cuando se implementa el servicio de la supervivencia

Contenidos

Introducción

Este documento describe cómo resolver problemas una falla de llamada cuando el script porta de la supervivencia de la voz del cliente (CVP) se configura en el dial-peer entrante de un gateway de ingreso.

Contribuido por Kabeer Noorudeen, ingeniero de Cisco TAC.

Prerrequisitos

Requisitos

Cisco recomienda que tenga conocimiento sobre estos temas:

- Flujo de llamada completo del CVP
- Gateway del IOS y protocolo PRI
- Cisco unificó el Intelligent Contact Management (ICM), las implementaciones del Cisco Unified Contact Center Enterprise (UCCE)

Componentes Utilizados

La información que contiene este documento se basa en estas versiones de software:

- Servidor 9.0 del CVP y arriba
- UCCE 9.0 y arriba
- Gateway del IOS 15.X

La información que contiene este documento se creó a partir de los dispositivos en un ambiente de laboratorio específico. Todos los dispositivos que se utilizan en este documento se pusieron en funcionamiento con una configuración verificada (predeterminada). Si la red está funcionando, asegúrese de haber comprendido el impacto que puede tener cualquier comando.

Problema

Cuando un servicio se agrega al los POTS entrantes Dialpeer para accionar el script de la supervivencia, la llamada falla con esta causa:

ccCallDisconnect: Causa Value=81

Nota: funcione con el comando debug, **inout del ccapi del voip del debug**, para ver el código de error señalado cuando la llamada falla. Además, funcione con el **ccsip message del debug del** comando debug si usted quiere ver los mensajes de error del SORBO señalados cuando la llamada falla.

Los mensajes de error contrarios están señalados cuando se habilita el **ccsip message del debug**. SORBA los mensajes, no dé mucha indicación con excepción de ése que la llamada falla después de que se vuelva la escritura de la etiqueta de la red VRU, que puede ser engañosa.

Si el servicio se quita del del dial-peer de los POTS entrantes, todo trabaja muy bien.

Procedimiento

Paso 1. Uno los gatewayes de ingreso habilita los debugs: **haga el debug del inout del ccapi del voip**, **haga el debug del ccsip message, haga el debug de la aplicación de voz toda,** y **hagala el debug de ISDN q931**.

Paso 2. Reconstruya el problema. Ponga una llamada al gateway.

Paso 3. Recoja los registros. Utilice el comando, **muestre el registro.**

Si usted mira los registros del CCAPI (**inout del ccapi del voip del debug**), usted apenas ve que la llamada consigue dada con éxito del script de la supervivencia y entonces la llamada falla:

Alerón

* 13 de septiembre 19:29:57.507 CT: //21/9E2AF1DC800C/CCAPI/cc_process_call_setup_ind:
>>>>CCAPI dio el cid 21 con la etiqueta 101 al app "_ManagedAppProcess_cvp-survivability"
* 13 de septiembre 19:29:58.507 CT: //21/9E2AF1DC800C/CCAPI/ccCallSetupAck:
Llamada Id=21

* 13 de septiembre 19:29:58.551 CT: //21/9E2AF1DC800C/CCAPI/cc_api_call_disconnected:
Causa Value=81, Interface=0x2500E10, llamada Id=21
* 13 de septiembre 19:29:58.551 CT: //21/9E2AF1DC800C/CCAPI/cc_api_call_disconnected:
Entrada de llamada (Responsed=FALSE, causa Value=81, recomprobación Count=0)
* 13 de septiembre 19:29:58.551 CT: //22/9E2AF1DC800C/CCAPI/ccCallDisconnect:
Causa Value=81, Tag=0x0, entrada de llamada (desconexión anterior Cause=0, desconexión Cause=0)
* 13 de septiembre 19:29:58.551 CT: //22/9E2AF1DC800C/CCAPI/ccCallDisconnect:
Causa Value=81, entrada de llamada (Responsed=FALSE, causa Value=81)
* 13 de septiembre 19:29:58.551 CT: //21/9E2AF1DC800C/CCAPI/ccCallDisconnect:
Causa Value=81, Tag=0x0, entrada de llamada (desconexión anterior Cause=0, desconexión Cause=81)
* 13 de septiembre 19:29:58.551 CT: //21/9E2AF1DC800C/CCAPI/ccCallDisconnect:
Causa Value=81, entrada de llamada (Responsed=TRUE, causa Value=81)

* 13 de septiembre 19:29:57.507 CT: //21/9E2AF1DC800C/CCAPI/cc_process_call_setup_ind: >>>>CCAPI dio el cid 21 con la etiqueta 101 app "_ManagedAppProcess_cvp-survivability " * al 13 de septiembre 19:29:58.507 CT: //21/9E2AF1DC800C/CCAPI/ccCallSetupAck: Llamada Id=21*Sep 13 19:29:58.551 CT: //21/9E2AF1DC800C/CCAPI/cc_api_call_disconnected: Causa Value=81, Interface=0x2500E10, llamada Id=21*Sep 13 19:29:58.551 CT: //21/9E2AF1DC800C/CCAPI/cc_api_call_disconnected: Entrada de llamada (Responsed=FALSE, causa Value=81, recomprobación Count=0)*Sep 13 19:29:58.551 CT: //22/9E2AF1DC800C/CCAPI/ccCallDisconnect: Causa Value=81, Tag=0x0, entrada de llamada (desconexión anterior Cause=0, desconexión Cause=0)*Sep 13 19:29:58.551 CT: //22/9E2AF1DC800C/CCAPI/ccCallDisconnect: Causa Value=81, entrada de llamada (Responsed=FALSE, causa Value=81)*Sep 13 19:29:58.551 CT: //21/9E2AF1DC800C/CCAPI/ccCallDisconnect: Causa Value=81, Tag=0x0, entrada de llamada (desconexión anterior Cause=0, desconexión Cause=81)*Sep 13 19:29:58.551 CT: //21/9E2AF1DC800C/CCAPI/ccCallDisconnect: Causa Value=81, entrada de llamada (Responsed=TRUE, causa Value=81)

Si usted mira los debugs para tcl (e.g **aplicación toda del** debugvoice**),** usted ve al llamador desconectado cuando el tcl intenta conseguir al estado de la llamada actual.

Alerón

* 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_InfotagObjCmd: el infotag consigue evt_state_current
* 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_InfotagGetObjCmd: el infotag consigue

evt_state_current
* 13 de septiembre 20:29:54.211 CT: //81//AFW_:/vtr_ev_state_current: argc 2
* 13 de septiembre 20:29:54.211 CT: //81//AFW_:/vtr_ev_state_current: [CALL_INIT] del evento
*** 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_FSMObjCmd: setstate CALLER_DISCONNECTED FSM**
*** 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_FSMSetStateObjCmd: setstate CALLER_DISCONNECTED del setstate**
* 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_InfotagObjCmd: el infotag consigue el con_ofleg 81
* 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_InfotagGetObjCmd: el infotag consigue el con_ofleg 81
* 13 de septiembre 20:29:54.211 CT: //81//AFW_:/vtr_co_ofleg: argindex 2 del argc 3
* 13 de septiembre 20:29:54.211 CT: //81//Tcl: /tcl_parseCallID_vartagObj: Pierna Count=1 de la traducción VARTAG
* 13 de septiembre 20:29:54.211 CT: //81//AFW_:/vtr_co_ofleg: []EV_CONNECTIONS
* 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_LegObjCmd: desconexión 81 de la pierna
* 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_LegDisconnectObjCmd: desconexión 81
* 13 de septiembre 20:29:54.211 CT: //81//Tcl: /tcl_parseCallID_vartagObj: Pierna Count=1 de la traducción VARTAG
* 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_InfotagObjCmd: el infotag consigue evt_state_current * 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_InfotagGetObjCmd: el infotag consigue evt_state_current * 13 de septiembre 20:29:54.211 CT: //81//AFW_:/vtr_ev_state_current: argc 2*Sep 13 20:29:54.211 CT: //81//AFW_:/vtr_ev_state_current: [CALL_INIT] del evento * 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_FSMObjCmd: setstate CALLER_DISCONNECTED FSM * 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_FSMSetStateObjCmd: setstate CALLER_DISCONNECTED del setstate * 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_InfotagObjCmd: el infotag consigue el con_ofleg 81 * 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_InfotagGetObjCmd: el infotag consigue el con_ofleg 81 * 13 de septiembre 20:29:54.211 CT: //81//AFW_:/vtr_co_ofleg: argindex 2*Sep 13 20:29:54.211 CT del argc 3: //81//Tcl: /tcl_parseCallID_vartagObj: Pierna Count=1*Sep 13 20:29:54.211 CT de la traducción VARTAG: //81//AFW_:/vtr_co_ofleg: []EV_CONNECTIONS * 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_LegObjCmd: desconexión 81 de la pierna * 13 de septiembre 20:29:54.211 CT: //81//TCL: /tcl_LegDisconnectObjCmd: desconexión 81 * 13 de septiembre 20:29:54.211 CT: //81//Tcl: /tcl_parseCallID_vartagObj: Pierna Count=1 de la traducción VARTAG

Ahora, si usted se zambulle abajo a los mensajes isdn (**debug ISDN q931**) usted ve que los mensajes no están en sincronización con los mensajes PSTN. Hay un mensaje FACILITY que está viniendo adentro inmediatamente después de la configuración y antes del procedimiento de la llamada. El script de la supervivencia no puede manejar esta situación.

Alerón
* 13 de septiembre 19:53:31.763 CT: ISDN Se0/0/0:23 Q931: RX < - CONFIGURACIÓN paladio = 8 callref = 0x1114
*** 13 de septiembre 19:53:31.767 CT: ISDN Se0/0/0:23 Q931: RX < - RECURSO paladio = 8 callref = 0x1114**
*** 13 de septiembre 19:53:31.767 CT: ISDN Se0/0/0:23 ** ERROR **: L3_BadPeerMsg: callid 0x0 del cr 0x9114 del evento 0x62**
* 13 de septiembre 19:53:31.767 CT: ISDN Se0/0/0:23 Q931: TX- > RELEASE_COMP paladio = 8 callref = 0x9114

* 13 de septiembre 19:53:32.767 CT: ISDN Se0/0/0:23 Q931: TX- > CALL_PROC paladio = 8 callref = 0x9114
* 13 de septiembre 19:53:32.767 CT: ISDN Se0/0/0:23 Q931: El TX- > CONECTA paladio = 8 callref = 0x9114
* 13 de septiembre 19:53:32.803 CT: ISDN Se0/0/0:23 Q931: RX < - RELEASE_COMP paladio = 8 callref = 0x1114
* 13 de septiembre 19:53:32.807 CT: ISDN Se0/0/0:23 Q931: RX < - RELEASE_COMP paladio = 8 callref = 0x1114
*** 13 de septiembre 19:53:32.807 CT: ISDN Se0/0/0:23 ** ERROR **: L3_BadPeerMsg: callid 0x0 del cr 0x9114 del evento 0x5A**

* 13 de septiembre 19:53:31.763 CT: ISDN Se0/0/0:23 Q931: RX < - CONFIGURACIÓN paladio = 8 callref = 0x1114 * 13 de septiembre 19:53:31.767 CT: ISDN Se0/0/0:23 Q931: RX < - RECURSO paladio = 8 callref = 0x1114 * 13 de septiembre 19:53:31.767 CT: ISDN Se0/0/0:23 ** ERROR **: L3_BadPeerMsg: callid 0x0*Sep 13 19:53:31.767 CT del cr 0x9114 del evento 0x62: ISDN Se0/0/0:23 Q931: TX- > RELEASE_COMP paladio = 8 callref = 0x9114 * 13 de septiembre 19:53:32.767 CT: ISDN Se0/0/0:23 Q931: TX- > CALL_PROC paladio = 8 callref = 0x9114 * 13 de septiembre 19:53:32.767 CT: ISDN Se0/0/0:23 Q931: El TX- > CONECTA paladio = 8 callref = 0x9114*Sep 13 19:53:32.803 CT: ISDN Se0/0/0:23 Q931: RX < - RELEASE_COMP paladio = 8 callref = 0x1114 * 13 de septiembre 19:53:32.807 CT: ISDN Se0/0/0:23 Q931: RX < - RELEASE_COMP paladio = 8 callref = 0x1114 * 13 de septiembre 19:53:32.807 CT: ISDN Se0/0/0:23 ** ERROR **: L3_BadPeerMsg: callid 0x0 del cr 0x9114 del evento 0x5A

Solución

El L3_BadPeerMsg es la clave aquí para solucionar este problema. Este error está señalado cuando hay una discordancía en el tipo de switch entre el gateway y el PSTN.

La recomendación es funcionar con el comando de fijar el tipo del switch de ISDN que el PSTN y el gateway están de acuerdo. En este escenario el tipo de switch en el PSTN es primario-**ni.** Para los clientes E.E.U.U., el tipo de switch de uso general es primario-**ni.**

El primario-ni del comando isdn switch-type configurado en el gateway de ingreso solucionó el problema.