

## Содержание

[Введение](#)

[Предварительные условия](#)

[Требования](#)

[Используемые компоненты](#)

[Проблема](#)

[Процедура](#)

[Решение](#)

## Введение

Этот документ описывает, как устранить неполадки ошибки вызова, когда сценарий Жизнеспособности клиентского голосового портала (CVP) настроен на входящем одноранговом телефонном соединении шлюза для внешнего доступа.

Внесенный Кабиром Нурудином, специалистом службы технической поддержки Cisco.

## Предварительные условия

### Требования

Компания Cisco рекомендует предварительно ознакомиться со следующими предметами:

- CVP Всесторонний поток вызовов
- IOS-шлюз и протокол PRI
- Cisco унифицированный Intelligent Contact Management (ICM), Cisco Unified Contact Center Enterprise (UCCE) развертывания

### Используемые компоненты

Сведения, содержащиеся в этом документе, касаются следующих версий программного обеспечения:

- Сервер CVP 9.0 и выше
- UCCE 9.0 и выше

- IOS-шлюз 15. X

Сведения, представленные в этом документе, были получены от устройств, работающих в специальной лабораторной среде. Все устройства, описанные в этом документе, были запущены с чистой (стандартной) конфигурацией. В рабочей сети необходимо изучить потенциальное воздействие всех команд до их использования.

## Проблема

Когда сервис добавлен к входящие POT Dialpeer для инициирования сценария Жизнеспособности, сбоя вызова с этой причиной:

**ccCallDisconnect:Cause Value=81**

**Примечание:** выполните команду отладки, **debug voip ccapi inout**, для наблюдения, код ошибки сообщил, когда отказывает вызов. Кроме того, выполните **сообщение CCSIP отладки** команды отладки, если вы хотите видеть, что сообщения об ошибках SIP сообщили, когда отказывает вызов.

Когда **сообщение CCSIP отладки** включено, сообщают о противоречивых сообщениях об ошибках.

Сообщения SIP, не давайте много индикации, кроме которой сбой вызова после того, как возвращена метка сети VRU, который может вводить в заблуждение.

Если сервис удален из от точки вызова входящих POT, все хорошо работает.

## Процедура

Шаг 1. Один Шлюзы для внешнего доступа включают отладки: **debug voip ccapi inout**, **сообщение CCSIP отладки**, **отлаживает голосовое приложение все** и **debug isdn q931**.

Шаг 2. Воссоздайте проблему. Закажите телефонный разговор со шлюзом.

Шаг 3. Соберите журналы. Используйте команду, **show logging**.

При рассмотрении журналов CCAPi (**debug voip ccapi inout**) вы просто видите, что вызов

успешно вручен от сценария жизнеспособности и затем сбоев вызова:

### [Спойлер](#)

\*13 сентября 19:29:57.507 кар://21/9E2AF1DC800C/CCAPI/cc\_process\_call\_setup\_ind: >>>> ССАPI вручил cid 21 с меткой 101 к приложению "\_ManagedAppProcess\_cvр-жизнеспособность"

\*13 сентября 19:29:58.507 кар://21/9E2AF1DC800C/CCAPI/ccCallSetupAck: Вызовите Id=21

\*13 сентября 19:29:58.551 кар://21/9E2AF1DC800C/CCAPI/cc\_api\_call\_disconnected: Причина Value=81, Interface=0x2500E10, вызовите Id=21

\*13 сентября 19:29:58.551 кар://21/9E2AF1DC800C/CCAPI/cc\_api\_call\_disconnected: Запись вызова (Responded=FALSE, причина Value=81, повторяют Count=0),

\*13 сентября 19:29:58.551 кар://22/9E2AF1DC800C/CCAPI/ccCallDisconnect: Причина Value=81, Tag=0x0, запись вызова (предыдущее разъединение Cause=0, Cause=0 разъединения)

\*13 сентября 19:29:58.551 кар://22/9E2AF1DC800C/CCAPI/ccCallDisconnect: Причина Value=81, запись вызова (Responded=FALSE, причина Value=81)

\*13 сентября 19:29:58.551 кар://21/9E2AF1DC800C/CCAPI/ccCallDisconnect: **Причина Value=81, Tag=0x0, запись вызова (предыдущее разъединение Cause=0, Cause=81 разъединения)**

\*13 сентября 19:29:58.551 кар://21/9E2AF1DC800C/CCAPI/ccCallDisconnect: Причина Value=81, запись вызова (Responded=TRUE, причина Value=81)

\*13 сентября 19:29:57.507 кар://21/9E2AF1DC800C/CCAPI/cc\_process\_call\_setup\_ind: >>>> ССАPI вручил cid 21 с меткой 101 к приложению "\_ManagedAppProcess\_cvр-жизнеспособность" \*Sep 13 19:29:58.507

кар://21/9E2AF1DC800C/CCAPI/ccCallSetupAck: Назовите Id=21\*Sep 13 19:29:58.551

кар://21/9E2AF1DC800C/CCAPI/cc\_api\_call\_disconnected: Причина Value=81, Interface=0x2500E10, Вызов Id=21\*Sep 13 19:29:58.551

кар://21/9E2AF1DC800C/CCAPI/cc\_api\_call\_disconnected: Запись вызова (Responded=FALSE, Причина Value=81, Повторная попытка Count=0) \*Sep 13 19:29:58.551

кар://22/9E2AF1DC800C/CCAPI/ccCallDisconnect: Причина Value=81, Tag=0x0, Запись вызова (Предыдущее Разъединение Cause=0, Разъединение Cause=0) \*Sep 13 19:29:58.551

кар://22/9E2AF1DC800C/CCAPI/ccCallDisconnect: Причина Value=81, Запись вызова (Responded=FALSE, Причина Value=81) \*Sep 13 19:29:58.551

кар://21/9E2AF1DC800C/CCAPI/ccCallDisconnect: Причина Value=81, Tag=0x0, Запись вызова (Предыдущее Разъединение Cause=0, Разъединение Cause=81) \*Sep 13

19:29:58.551 кар://21/9E2AF1DC800C/CCAPI/ccCallDisconnect: Причина Value=81, Запись вызова (Responded=TRUE, Причина Value=81)

Если вы посмотрели на отладки для tcl (например, **отлаживаете голосовое приложение все**), вы видите абонента, разъединенного, когда tcl пытается получить состояние текущего вызова.

### [Спойлер](#)

\*13 сентября 20:29:54.211 кар://81//TCL:/tcl\_InfotagObjCmd: infotag получают evt\_state\_current

\*13 сентября 20:29:54.211 кап://81//TCL:/tcl\_InfotagGetObjCmd: infotag получают evt\_state\_current

\*13 сентября 20:29:54.211 кап://81//AFW\_:/vtr\_ev\_state\_current: argc 2

\*13 сентября 20:29:54.211 кап://81//AFW\_:/vtr\_ev\_state\_current: Событие [CALL\_INIT]

\*13 сентября 20:29:54.211 кап://81//TCL:/tcl\_FSMObjCmd: fsm setstate CALLER\_DISCONNECTED

\*13 сентября 20:29:54.211 кап://81//TCL:/tcl\_FSMSetStateObjCmd: setstate setstate CALLER\_DISCONNECTED

\*13 сентября 20:29:54.211 кап://81//TCL:/tcl\_InfotagObjCmd: infotag получают con\_ofleg 81

\*13 сентября 20:29:54.211 кап://81//TCL:/tcl\_InfotagGetObjCmd: infotag получают con\_ofleg 81

\*13 сентября 20:29:54.211 кап://81//AFW\_:/vtr\_co\_ofleg: argc 3 argindex 2

\*13 сентября 20:29:54.211 кап://81//Tcl:/tcl\_parseCallID\_vartagObj: Участок Трансляции VARTAG Count=1

\*13 сентября 20:29:54.211 кап://81//AFW\_:/vtr\_co\_ofleg: EV\_CONNECTIONS []

\*13 сентября 20:29:54.211 кап://81//TCL:/tcl\_LegObjCmd: разъединение участка 81

\*13 сентября 20:29:54.211 кап://81//TCL:/tcl\_LegDisconnectObjCmd: разъединение 81

\*13 сентября 20:29:54.211 кап://81//Tcl:/tcl\_parseCallID\_vartagObj: Участок Трансляции VARTAG Count=1

\*13 сентября 20:29:54.211 кап://81//TCL:/tcl\_InfotagObjCmd: infotag получают evt\_state\_current

\*13 сентября 20:29:54.211 кап://81//TCL:/tcl\_InfotagGetObjCmd: infotag получают evt\_state\_current \*13 сентября 20:29:54.211 кап://81//AFW\_:/vtr\_ev\_state\_current: argc 2\*Sep 13 20:29:54.211 кап://81//AFW\_:/vtr\_ev\_state\_current: Событие [CALL\_INIT] \*Sep 13 20:29:54.211 кап://81//TCL:/tcl\_FSMObjCmd: fsm setstate CALLER\_DISCONNECTED \*13 сентября 20:29:54.211 кап://81//TCL:/tcl\_FSMSetStateObjCmd: setstate setstate CALLER\_DISCONNECTED \*13 сентября 20:29:54.211 кап://81//TCL:/tcl\_InfotagObjCmd: infotag получают con\_ofleg 81 \*13 сентября 20:29:54.211 кап://81//TCL:/tcl\_InfotagGetObjCmd: infotag получают con\_ofleg 81 \*13 сентября 20:29:54.211 кап://81//AFW\_:/vtr\_co\_ofleg: 20:29:54 argc 3 argindex 2\*Sep 13.211 кап://81//Tcl:/tcl\_parseCallID\_vartagObj: Участок Трансляции VARTAG Count=1\*Sep 13 20:29:54.211 кап://81//AFW\_:/vtr\_co\_ofleg: EV\_CONNECTIONS [] \*Sep 13 20:29:54.211 кап://81//TCL:/tcl\_LegObjCmd: разъединение участка 81 \*13 сентября 20:29:54.211 кап://81//TCL:/tcl\_LegDisconnectObjCmd: разъедините 81 \*13 сентября 20:29:54.211 кап://81//Tcl:/tcl\_parseCallID\_vartagObj: Участок Трансляции VARTAG Count=1

Теперь, если вы погружаетесь вниз к сообщениям ISDN (**debug isdn q931**), вы видите, что сообщения не находятся в синхронизации с сообщениями PSTN. Существует Сообщение FACILITY, которое входит сразу после настройки и перед обработкой вызова. Сценарий жизнеспособности не в состоянии обработать эту ситуацию.

### [Спойлер](#)

\*13 сентября 19:53:31.763 кап: ISDN Se0/0/0:23 Q931: RX <-УСТАНОВЛИВАЕТ фунт = 8 callref = 0x1114

\*13 сентября 19:53:31.767 кап: ISDN Se0/0/0:23 Q931: RX <-фунт СРЕДСТВА = 8 callref = 0x1114

\*13 сентября 19:53:31.767 кап: ISDN Se0/0/0:23 \*\* ОШИБКА \*\*: L3\_BadPeerMsg: событие 0x62 сг 0x9114 callid 0x0

\*13 сентября 19:53:31.767 кап: ISDN Se0/0/0:23 Q931: TX-> фунт RELEASE\_COMP = 8 callref = 0x9114

\*13 сентября 19:53:32.767 кап: ISDN Se0/0/0:23 Q931: TX-> фунт CALL\_PROC = 8 callref = 0x9114

\*13 сентября 19:53:32.767 кап: ISDN Se0/0/0:23 Q931: TX-> ПОДКЛЮЧАЕТ фунт = 8 callref =

0x9114

\*13 сентября 19:53:32.803 кар: ISDN Se0/0/0:23 Q931: RX <-фунт RELEASE\_COMP = 8 callref = 0x1114

\*13 сентября 19:53:32.807 кар: ISDN Se0/0/0:23 Q931: RX <-фунт RELEASE\_COMP = 8 callref = 0x1114

\*13 сентября 19:53:32.807 кар: ISDN Se0/0/0:23 \*\* ОШИБКА \*\*: L3\_BadPeerMsg: событие 0x5A cr 0x9114 callid 0x0

\*13 сентября 19:53:31.763 кар: ISDN Se0/0/0:23 Q931: RX <-УСТАНОВЛИВАЕТ фунт = 8 callref = 0x1114 \*13 сентября 19:53:31.767 кар: ISDN Se0/0/0:23 Q931: RX <-фунт СРЕДСТВА = 8 callref = 0x1114 \*13 сентября 19:53:31.767 кар: ISDN Se0/0/0:23 \*\* ОШИБКА \*\*: L3\_BadPeerMsg: 19:53:31 cr 0x9114 callid 0x0\*Sep 13 события 0x62.767 кар: ISDN Se0/0/0:23 Q931: TX-> фунт RELEASE\_COMP = 8 callref = 0x9114 \*13 сентября 19:53:32.767 кар: ISDN Se0/0/0:23 Q931: TX-> фунт CALL\_PROC = 8 callref = 0x9114 \*13 сентября 19:53:32.767 кар: ISDN Se0/0/0:23 Q931: TX-> ПОДКЛЮЧАЕТ фунт = 8 callref = 19:53:32 0x9114\*Sep 13.803 кар: ISDN Se0/0/0:23 Q931: RX <-фунт RELEASE\_COMP = 8 callref = 0x1114 \*13 сентября 19:53:32.807 кар: ISDN Se0/0/0:23 Q931: RX <-фунт RELEASE\_COMP = 8 callref = 0x1114 \*13 сентября 19:53:32.807 кар: ISDN Se0/0/0:23 \*\* ОШИБКА \*\*: L3\_BadPeerMsg: событие 0x5A cr 0x9114 callid 0x0

## Решение

L3\_BadPeerMsg является ключом здесь для решения этой проблемы. Когда существует несоответствие на типе коммутатора между шлюзом и PSTN, об этой ошибке сообщают.

Рекомендация состоит в том, чтобы выполнить команду для установки типа коммутатора ISDN, который согласовывают PSTN и шлюз. В этом сценарии тип коммутатора на PSTN является **основным-ni**. Для клиентов US switch-type, обычно используемый, является **основным-ni**.

**Команда isdn switch-type, основная-ni** настроенный на шлюзе для внешнего доступа, решила проблему.