

# Программный коммутатор PGW 2200 – TCAP выпуска 9.3 или более поздней версии

## Содержание

[Введение](#)

[Предварительные условия](#)

[Требования](#)

[Используемые компоненты](#)

[Условные обозначения](#)

[Общие сведения](#)

[Разрешение TCAP](#)

[Анализатор линия Ethernet](#)

[Трассировка TCAP Platform.log](#)

[Программное средство трассировки MDL](#)

[Приложение A: Метки MDL](#)

[Приложение Б: Точечные коды SS7 выхода](#)

[Приложение В: Типы сообщения SCCP](#)

[Unitdata \(UDT\)](#)

[Сервис unitdata \(UDTS\)](#)

[UDTS возвращает причины](#)

[Приложение D: Интерфейс MDL для сообщения TCAP](#)

[Приложение E: Внутренний интерфейс MDL](#)

[Дополнительные сведения](#)

## **Введение**

Часть приложений возможностей транзакции (TCAP) оказывает поддержку для интерактивных приложений в распределенной среде. TCAP определяет сквозной протокол между своими пользователями. Это может быть расположено в сети SS7 или другой сети, которая поддерживает TCAP (IP).

## **Предварительные условия**

### **Требования**

Читатели данной документации должны знать:

- [Выпуск 9 Cisco Media Gateway Controller](#)

### **Используемые компоненты**

Сведения в этом документе основываются на программном коммутаторе Cisco PGW 2200.

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.

## Условные обозначения

[Дополнительные сведения об условных обозначениях см. в документе Технические рекомендации Cisco. Условные обозначения.](#)

## Общие сведения

Протокол TCAP состоит из двух подуровней:

- Подуровень компонента
- Подуровень транзакции

Подуровень компонента взаимодействует с модулем преобразования. Модуль преобразования является эквивалентом пользователя службы или номера подсистемы (SSN). Подуровень компонента поддерживает эти сервисы:

- Сопоставление действий и ответы.
- Аварийная обработка ситуации.

Подуровень транзакции взаимодействует с Контрольной частью сигнального соединения (SCCP). TCAP только поддерживает сетевой сервис без предварительного соединения. Подуровень транзакции связывается с SCCP через интерфейс без установления соединения.

Программное обеспечение TCAP использует сервисы программного обеспечения SCCP для маршрутизации сообщений пользователю TCAP в узле - адресате. Интерфейс между TCAP и программным обеспечением SCCP сильно связан. Каждый запрос TCAP от механизма содержит глобальное название и целевой номер подсистемы. TCAP предоставляет номер подсистемы SCCP для поиска кода Пунктов передачи сигнала (STP). Если адреса SS7 и маршруты настроены правильно и полностью в рабочем состоянии, устраняют неполадки SCCP, и информация TCAP передала и получила между Cisco PGW 2200 и удаленным SCCP или узлом TCAP.

Cisco PGW 2200 использует SCCP для инкапсуляции запросов TCAP для транспорта Сторона передачи сообщений (MTP). Эта связь по протоколу SCCP между узлами передается без соединения по MTP. Cisco PGW 2200 использует SCCP Unidata (UDT) для передачи данных к удаленному узлу SCCP для передачи без установки соединения. Когда сообщение SCCP UDT отправлено успешно, PGW 2200 получает верный ответ. Это, как правило, находится в форме сообщения UDT. Обмен этими сообщениями UDT упрощает передачу без установки соединения между PGW 2200 и удаленным узлом SCCP (таким как Узел управления услугами [SCP] для поисков ядра данных TCAP). PGW 2200 определяет необязательное поле в UDT, который сообщает, что узел SCCP должен "возвратить на ошибку" содержание любого сообщения, которое это передает к удаленному узлу, если сообщение UDT является недоставленным. Сервис Unidata (UDTS) сообщение используется для упрощения этого ошибочного ответа. Сообщение UDTS указывает к PGW 2200, что сообщение UDT, полученное в удаленном узле (таком как STP или SCP), не может

быть отправлено назначению.

## Настройка понятия Cisco PGW 2200

### Разрешение TCAP

Обмен сообщениями SCCP (UDT/UDTS), обсужденный [в фоновом режиме, Раздел сведений](#) важен при устранении проблем сервисов TCAP и функциональности. Решите любые проблемы на уровне SCCP перед устранением проблем данных TCAP, передаваемых или полученных. Формат UDT и сообщения UDTS показывают в [Приложении С](#).

Используйте эти PGW Cisco 2200 программных средств для отладки вызовов, которые требуют TCAP (TCAP / SCCP) сервисы:

- [Анализатор Линия Ethernet](#) с программными средствами такой как Эфирный, ищайка UNIX и Инспектор.
- [TCAP Platform.log отслеживает](#) на PGW 2200.
- [Программное средство Трассировки MDL](#) для обработки вызовов в уровне протокола.

### Анализатор линия Ethernet

Cisco PGW 2200 использует Надежный UDP (RUDP) для передачи MTP3 и сообщений SS7 верхнего уровня между локальным MTP1 и устройствами MTP2 (такими как Терминал звена сигнализации [SLT]). Эта связь, как правило, делается по порту 7000 в интерфейсе локальной сети Ethernet Cisco PGW 2200. Это конфигурируемо. См. [руководство по конфигурации](#) для подробных данных о настройке портов PGW "stPort" в XECfgParm.dat.

Можно использовать любой анализатор Ethernet для просмотра пакетов, переданных между Cisco PGW 2200 и его локальным устройством управления MTP2. Однако не все они поддерживают MTP, и протокол SCCP использовал отображать декодируемое сообщение. Если анализатор Ethernet не доступен клиенту, используйте команду **ищайки** UNIX для устранения проблем. Выходные данные команды **ищайки** не являются дружественными к пользователю, но являются полезными в наихудшем случае.

Предпочтен анализатор Ethernet, который поддерживает стек протоколов SS7. Это позволяет вам декодировать пакеты, замеченные на интерфейсе Ethernet Cisco PGW 2200. Анализатор с открытым исходным кодом такой как [Эфирный](#) может также использоваться и доступен онлайн.

Если никакая коммерческая служебная программа анализа трафика не доступна, выполните команду **ищайки** на целевом Cisco PGW 2200 для наблюдения выходных данных данных в шестнадцатеричном виде сообщений, передаваемых и от Cisco PGW 2200. С корневыми разрешениями на Cisco PGW 2200 выполните эту команду для наблюдения данных в шестнадцатеричном виде, передаваемых из настроенного 'stPort'. Для дополнительных сведений о команде **ищайки** обратитесь к 'оперативным страницам руководства ищайки' или SUN Административные руководства.

```
#snoop -d <ethernet device name> -x 42 port <stPort>
```

Выполните эту команду для отслеживания, пакеты отослали Устройство ethernet, hmeX, на порту 7000.

```
#snoop -d hmeX -x 42 port 7000
```

Это - пример выходных данных перехваченных пакетов SS7 с командой **ищейки**.

Инспектор Cisco может также использоваться (при наличии) для показа шестнадцатеричного дампа сообщения SCCP. Заголовок сообщения SCCP декодируется, но показ выходных данных зависит от версии выбранного Инспектора. Важный момент - то, что тип сообщения видим и дает индикацию как, туда, где начать устранять неполадки потока вызовов. Шестнадцатеричный дамп показывает, что тип сообщения, 09 является сообщением UDT и типом сообщения 0a, является служебным сообщением UDTS, которое указывает на ошибку. Направление потока сообщений также полезно, так как показывают PC SS7. Если остаток шестнадцатеричного дампа показывают (зависит от версии приложения отслеживания), это может использоваться для дальнейшего декодирования SCCP и частей TCAP сообщения. Это основывается на промышленных стандартах для SCCP и TCAP.

Это - выходные данные Snoopер сообщения UDT SCCP с данными TCAP (к PSTN).

Если существует недоставленное сообщение SCCP UDT, передаваемое от Cisco PGW 2200 и / или SCCP (на удаленном узле) имеет проблемы с сообщением, Cisco PGW 2200 получает ответное сообщение UDTS. Это сообщение указывает, 'возвращают причину', которая очень полезна в устранении проблем. UDTS является типом сообщения 10 (или hex 0a).

Это - пример сообщения UDTS SCCP с данными TCAP (от PSTN).

**Примечание:** Это сообщение является примером только и может не отразить фактическое сочетание запросов и ответов / последовательность. Формат и отображенное количество данных варьируются в зависимости от Версии приложения отслеживания.

Эти выходные данные Snoopер отображают меня AM, UDT, UDTS и последовательность РЭЛА.

**Примечание:** Это сообщение является примером только и может не отразить фактическое сочетание запросов и ответов / последовательность. Формат и отображенное количество данных варьируются в зависимости от Версии приложения отслеживания.

Это - отслеживание средств прослушивания SS7, которое включает SCCP SS7 и информацию TCAP.

```
-----  
SCP(IN)- 19/03/04 18:01:54:223      SCCP      SCP(IN)  UDT      SCP(IN)  BGN  INVK  IDP  
-----  
Octet001  ITU-T SS7                      Time=19/03/02 18:01:54:223  
-----  
11010011  BIB/BSN                               1/83  
10010110  FIB/FSN                               1/22  
..111111  SU type/length                        MSU63  
00.....  Spare                                 0  
-----  
Octet004  Service information octet  
-----  
....0011  Service indicator                     SCCP Signalling Connection Control Part  
..00....  Message priority                       0  
10.....  Network indicator                     N National network  
-----  
Octet005  Routing label
```

```

-----
..... DPC                10337 SCP(IN)
..... OPC                10321
0001.... SLS              1
-----
Octet009 Message type
-----
00001001 Message type          UDT   Unitdata
-----
Octet010 SCCP Protocol Class parameter
-----
....0001 Protocol class        Class 1
0000.... Message handling      No special options
00000011 Ptr -> Called number   3
00000111 Ptr -> Calling #      7
00001011 Pointer -> Data       11
-----
Octet014 SCCP Called Party Address parameter
-----
00000100 Parameter length      4
.....1 Sgnl pt code bit       SPC present
.....1. Subsystem # bit       SSN present
..0000.. Global title ind     No global title included
.1..... Routing bit           DPC and SSN based routing
0..... Reserved natl use      0
..... Point code              10337 SCP(IN)
00..... Spare                 0
11111100 Subsystem number     INAP      IN-CS1+
-----
Octet019 SCCP Calling Party Address parameter
-----
00000100 Parameter length      4
.....1 Sgnl pt code bit       SPC present
.....1. Subsystem # bit       SSN present
..0000.. Global title ind     No global title included
.1..... Routing bit           DPC and SSN based routing
0..... Reserved natl use      0
..... Point code              10321
00..... Spare                 0
11111100 Subsystem number     INAP      IN-CS1+
-----
Octet024 SCCP Data parameter
-----
01100001 Parameter length     97
01100010 Tag                  BGN Begin, constructor, application-wide
01011111 Length               95
-----
Octet027 Originating Transaction ID
-----
...01000 Tag                  Originating Transaction ID
010..... Class and form       Application-wide, primitive
00000011 Length               3
..... Originating ID         F30051
-----
Octet032 TCAP Dialogue Portion
-----
...01011 Tag                  TCAP Dialogue Portion
011..... Class and form       Application-wide, constructor
00100011 Length               35
-----
Octet034 TCAP External
-----
...01000 Tag                  TCAP External
001..... Class and form       Universal, constructor

```

00100001	Length	33
-----		
Octet036	Object identifier	
-----		
...00110	Tag	Object identifier
000.....	Class and form	Universal, primitive
00000111	Length	7
00000000	Organization	itu-t recommendation
00010001	q	Q
.....	773 (X'305)	773
00000001	as(1)	1
00000001	Protocol data unit	dialogue PDU(1)
00000001	version(1)	1
10100000	Single-ASN.1-typeTag	Parameter
00010110	Length	22
-----		
Octet047	Dialogue request	
-----		
...00000	Tag	Dialogue request
011.....	Class and form	Application-wide, constructor
00010100	Length	20
-----		
Octet049	Protocol-version	
-----		
...00000	Tag	Protocol-version
100.....	Class and form	Context-specific, primitive
00000010	Length	2
00000111	Unused Bit	07
.0000000	Unused Bit	00
1.....	Protocol Version	Version 1
-----		
Octet053	Application-context-name	
-----		
...00001	Tag	Application-context-name
101.....	Class and form	Context-specific, constructor
00001110	Length	14
-----		
Octet055	Object Identifier	
-----		
...00110	Tag	Object identifier
000.....	Class and form	Universal, primitive
00001100	Length	12
00101010	Protocol	ccitt identified-organization
10000110	SubProtocol	etsi
00111010	Domain	inDomain
00000000	Network	in-Network
10001001	AC Name	ac (application context)
01100001	Service	csl-ssp-to-scp(0)
00110011	Version	Reserved
.....	Contents	01 00 01 00 01
-----		
Octet069	TCAP Component Portion	
-----		
...01100	Tag	TCAP Component Portion
011.....	Class and form	Application-wide, constructor
10000000	Length	128
-----		
Octet071	Invoke component	
-----		
...00001	Tag	Invoke component
101.....	Class and form	Context-specific, constructor
00101111	Length	47
-----		
Octet073	Invoke ID	

```

-----
...00010 Tag Invoke ID
000..... Class and form Universal, primitive
00000001 Length 1
00000001 Invoke ID 01
-----
Octet076 Operation Code
-----
...00010 Tag Local
000..... Class and form Universal, primitive
00000001 Length 1
..... Operation Code IDP InitialDP
-----
Octet079 Parameter Sequence
-----
...10000 Tag Parameter Sequence
001..... Class and form Universal, constructor
00100111 Length 39
-----
Octet081 ServiceKey
-----
...00000 Tag ServiceKey
100..... Class and form Context-specific, primitive
00000001 Length 1
..... Service key 94
-----
Octet084 CalledPartyNumber
-----
...00010 Tag CalledPartyNumber
100..... Class and form Context-specific, primitive
00000111 Length 7
.0000011 Nature of address National (significant) number( national use )
1..... Odd/even Odd number of address signals
...0000 Spare 00
.001.... Numbering plan ISDN (Telephony) numbering plan (Rec. E.164)
1..... Internal network # Routing to internal network number not allowed
..... Address signals 999956738
0000.... Filler 0
-----
Octet093 CallingPartyNumber
-----
...00011 Tag CallingPartyNumber
100..... Class and form Context-specific, primitive
00000111 Length 7
.0000011 Nature of address National (significant) number( national use )
1..... Odd/even Odd number of address signals
.....01 Screening Indicator User provided, verified and passed
....00.. Presentation? Presentation allowed
.001.... Numbering plan ISDN (Telephony) numbering plan (Rec. E.164)
0..... Number Incomplete? Complete
..... Address signals 2199997137
0000.... Filler 0
-----
Octet102 CallingPartysCategory
-----
...00101 Tag CallingPartysCategory
100..... Class and form Context-specific, primitive
00000001 Length 1
00001010 CallngPartyCategory Ordinary calling subscriber
-----
Octet105 ForwardCallIndicators
-----
...11010 Tag ForwardCallIndicators
100..... Class and form Context-specific, primitive

```

```

00000010 Length 2
.....0 Nat'l/International Call to be treated as a national call
.....00. End-to-end method No end-to-end method available
....1... Interworking Interworking encountered
...0.... End-to-end info No end-to-end information available
..1..... ISUP indicator ISDN user part used all the way
01..... ISUP preference ISDN user part not required all the way
.....1 Orig ISDN access Originating access ISDN
.....00. SCCP method No indication
....0... Spare 0
0000.... ReservedForNat'lUse 0
-----
Octet109 BearerCapability
-----
...11011 Tag BearerCapability
101..... Class and form Context-specific, constructor
00000101 Length 5
-----
Octet111 Bearer Cap
-----
...00000 Tag Bearer Cap
100..... Class and form Context-specific, primitive
-----
Octet112 User service information parameter
-----
00000011 Parameter length 3
-----
Octet113 User service info octet 3
-----
...00000 Transfer capability Speech
.00..... Coding standard CCITT standardized coding
1..... Extension bit 1
-----
Octet114 User service info octet 4
-----
...10000 Transfer rate 64 kbit/s
.00..... Transfer mode circuit mode
1..... Extension bit 1
-----
Octet115 User service info octet 5
-----
...00011 Layer 1 protocol Recommendation G.711 A-law
.01..... Layer 1 Identifier User information layer 1 protocol
1..... Extension bit 1
-----
Octet116 CalledPartyNumber
-----
...00010 Tag CalledPartyNumber
110..... Class and form Private use, primitive
00000010 Length 2
.0000000 Nature of address Spare
0..... Odd/even Even Number of Address signals
...1010 Spare 0A
.000.... Numbering plan Spare (no interpretation)
0..... Internal network # Routing to internal network number allowed
-----
Octet120 End-of-contents
-----
00000000 Tag 00
00000000 Length 00
-----
Checksum CRC16..... 0001011001110111 hex=1677
-----

```



-----  
SCP(IN)- 19/03/04 18:01:54:269 SCCP SCP(IN) UDT SCP(IN) CON INVK CUE

-----  
Octet001 ITU-T SS7 Time=19/03/02 18:01:54:269  
-----  
10000001 BIB/BSN 1/1  
10110010 FIB/FSN 1/50  
..111111 SU type/length MSU63  
00..... Spare 0  
-----  
Octet004 Service information octet  
-----  
....0011 Service indicator SCCP Signalling Connection Control Part  
..00.... Message priority 0  
10..... Network indicator N National network  
-----  
Octet005 Routing label  
-----  
..... DPC 10321  
..... OPC 10337 SCP(IN)  
1010.... SLS 10  
-----  
Octet009 Message type  
-----  
00001001 Message type UDT Unitdata  
-----  
Octet010 SCCP Protocol Class parameter  
-----  
....0001 Protocol class Class 1  
0000.... Message handling No special options  
00000011 Ptr -> Called number 3  
00000111 Ptr -> Calling # 7  
00001011 Pointer -> Data 11  
-----  
Octet014 SCCP Called Party Address parameter  
-----  
00000100 Parameter length 4  
.....1 Sgnl pt code bit SPC present  
.....1. Subsystem # bit SSN present  
..0000.. Global title ind No global title included  
.1..... Routing bit DPC and SSN based routing  
0..... Reserved natl use 0  
..... Point code 10321 Matinha  
00..... Spare 0  
11111100 Subsystem number INAP IN-CS1+  
-----  
Octet019 SCCP Calling Party Address parameter  
-----  
00000100 Parameter length 4  
.....1 Sgnl pt code bit SPC present  
.....1. Subsystem # bit SSN present  
..0000.. Global title ind No global title included  
.1..... Routing bit DPC and SSN based routing  
0..... Reserved natl use 0  
..... Point code 10337 SCP(IN)  
00..... Spare 0  
11111100 Subsystem number INAP IN-CS1+  
-----  
Octet024 SCCP Data parameter  
-----  
01001001 Parameter length 73

01100101	Tag	CON Continue, constructor, application-wide
01000111	Length	71
-----		
Octet027	Originating Transaction ID	
-----		
...01000	Tag	Originating Transaction ID
010.....	Class and form	Application-wide, primitive
00000011	Length	3
.....	Originating ID	7A01B4
-----		
Octet032	Destination Transaction ID	
-----		
...01001	Tag	Destination Transaction ID
010.....	Class and form	Application-wide, primitive
00000011	Length	3
.....	Destination ID	F30051
-----		
Octet037	TCAP Dialogue Portion	
-----		
...01011	Tag	TCAP Dialogue Portion
011.....	Class and form	Application-wide, constructor
00101111	Length	47
-----		
Octet039	TCAP External	
-----		
...01000	Tag	TCAP External
001.....	Class and form	Universal, constructor
00101101	Length	45
-----		
Octet041	Object identifier	
-----		
...00110	Tag	Object identifier
000.....	Class and form	Universal, primitive
00000111	Length	7
00000000	Organization	itu-t recommendation
00010001	q	Q
.....	773 (X'305)	773
00000001	as(1)	1
00000001	Protocol data unit	dialogue PDU(1)
00000001	version(1)	1
10100000	Single-ASN.1-typeTag	Parameter
00100010	Length	34
-----		
Octet052	Dialogue response	
-----		
...00001	Tag	Dialogue response
011.....	Class and form	Application-wide, constructor
00100000	Length	32
-----		
Octet054	Protocol-version	
-----		
...00000	Tag	Protocol-version
100.....	Class and form	Context-specific, primitive
00000010	Length	2
00000111	Unused Bit	07
.0000000	Unused Bit	00
1.....	Protocol Version	Version 1
-----		
Octet058	Application-context-name	
-----		
...00001	Tag	Application-context-name
101.....	Class and form	Context-specific, constructor
00001110	Length	14
-----		

```

Octet060 Object Identifier
-----
...00110 Tag Object identifier
000..... Class and form Universal, primitive
00001100 Length 12
00101010 Protocol ccitt identified-organization
10000110 SubProtocol etsi
00111010 Domain inDomain
00000000 Network in-Network
10001001 AC Name ac (application context)
01100001 Service cs1-ssp-to-scp(0)
00110011 Version Reserved
..... Contents 01 00 01 00 01
-----

Octet074 Result
-----
...00010 Tag Result
101..... Class and form Context-specific, constructor
00000011 Length 3
-----

Octet076 Integer
-----
...00010 Tag Integer
000..... Class and form Universal, primitive
00000001 Length 1
..... Value accepted
-----

Octet079 Result-source-diagnostic
-----
...00011 Tag Result-source-diagnostic
101..... Class and form Context-specific, constructor
00000101 Length 5
-----

Octet081 Dialogue service user
-----
...00001 Tag Dialogue service user
101..... Class and form Context-specific, constructor
00000011 Length 3
-----

Octet083 Integer
-----
...00010 Tag Integer
000..... Class and form Universal, primitive
00000001 Length 1
..... Value Null
-----

Octet086 TCAP Component Portion
-----
...01100 Tag TCAP Component Portion
011..... Class and form Application-wide, constructor
10000000 Length 128
-----

Octet088 Invoke component
-----
...00001 Tag Invoke component
101..... Class and form Context-specific, constructor
00000110 Length 6
-----

Octet090 Invoke ID
-----
...00010 Tag Invoke ID
000..... Class and form Universal, primitive
00000001 Length 1
00000001 Invoke ID 01

```

```

-----
Octet093  Operation Code
-----
...00010  Tag                Local
000..... Class and form  Universal, primitive
00000001  Length             1
.....    Operation Code  CUE  Continue
-----
Octet096  End-of-contents
-----
00000000  Tag                00
00000000  Length             00
-----
Checksum  CRC16.....         0011010011100010 hex=34E2
-----
-----
-----

```

### Совет устранения неполадок: UDTs возвращает причину

Для сообщения UDTs, 'возвращаются, причиной' является первый байт после типа сообщения 0a. Это значение помогает определять, почему STP / SCP передает ошибочный ответ UDTs. Если эта информация не видима в анализаторе, продолжите к разделу [Трассировки TCAP Platform.log](#) для разрешения трассировок TCAP в журнале Cisco PGW 2200.

### [Трассировка TCAP Platform.log](#)

MML позволяет пользователю запускать трассировку TCAP, которая формирует дампы сообщений <Trace> для канального контроллера TCAP в /opt/CiscoMGC/var/log/platform.log. Трассировка TCAP позволяет пользователю видеть TCAP / сообщения SCCP, передаваемые канальному контроллеру SS7 для маршрутизации к MTP3 переключения SS7. См. [Приложение E](#) для потока сообщений запроса TCAP через программное обеспечение PGW 2200.

Отслеживание TCAP запущено через mml с командой **sta-tcap-trc**. Для получения связанных сведений включите ведение журнала отладки для канального контроллера SS7 и TCAP.

Это - пример того, как разрешить трассировку TCAP:

```

mml> set-log:TCAP-01:debug,confirm MGC-01 - Media Gateway Controller 2004-03-26 11:17:31.503 EST
M COMPLD "TCAP-01" ; mml> set-log:ss7-i-1:debug,confirm MGC-01 - Media Gateway Controller 2004-
03-26 11:17:40.715 EST M COMPLD "ss7-i-1" ; mml> sta-tcap-trc MGC-01 - Media Gateway Controller
2004-03-26 11:05:27.040 EST M RTRV SROF "TCAP-01" ;

```

**Примечание:** Ведение журнала отладки может иметь эффект на производительность системы и не должно использоваться в производственной среде под большим количеством вызовов. Запланируйте свой период технического обслуживания соответственно.

### Сообщения TCAP, передаваемые Cisco PGW 2200

Как только IN\_TRIGGER передается механизму, существа механизма для передачи сообщения из PGW 2200. Информация, переданная от уровня протокола, передана к канальному контроллеру TCAP. Часть TCAP передается вниз канальному контроллеру SCCP. Кроме того, журнал создан в platform.log, чтобы указать, что было 'передано' сообщение TCAP. От предыдущего UDT - сообщения (показанный в части анализатора этого документа) вы видите, как PGW 2200 регистрирует информацию, отнесенную к этому тому же сообщению в platform.log. Этот журнал платформы совпадает с содержимым данных, показанным в

[Типовом Отказе Сообщения SCCP: Unitdata /](#) таблица [Сервиса Unitdata](#) в [Приложении С](#). От этой таблицы первое значение является значением длины данных (52 hex = 82 десятичных числа). Фактическая часть данных TCAP придерживается длины сообщения. Если анализатор или инспектор не доступны, этот platform.log может использоваться для просмотра / TCAP отладки и SCCP-транзакции.

**Совет устранения неполадок:** Если сообщение TCAP не передается вниз SCCP, существует проблема на уровне Механизма или MDL. Устраните неполадки трассировки MDL и посмотрите Ltrigger и сигнал LTriggerRelease.

Эти выходные данные показывают журнал PGW 2200, который передача TCAP вниз складывает к SCCP/МТР.

После того, как TCAP передает сообщение к SCCP, канальный контроллер SS7 играет RECEIVED MSG FROM SCCP и регистрирует шестнадцатеричное представление сообщения для указания на получение сообщения. Этот шестнадцатеричный дамп включает SCCP и части TCAP как показано в этих выходных данных.

### Советы устранения неполадок:

- Используйте формат сообщения SCCP, который, как показывают в [Приложении С](#), декодировал тип сообщения, информация заголовка SCCP (показанный в [выходных данных](#) в желтом цвете) и начало данных TCAP (показанный в [выходных данных](#) синего цвета). 1e0002 в [выходных данных](#) представляет код точки получателя от dpc.dat, и дамп сообщения SCCP сразу начинается после типа "1" (начинающийся с типа сообщения SCCP).
- PGW 2200 регистрирует счетчик и Сигналы тревоги для SCCP, событий TCAP и SS7. Если измерения включены, проверьте счетчики для сообщения TCAP. Также проверьте SCCP, UDT и UDTS, полученный и переданный. См. эти документы для рабочих процедур MGC. [Управление системными измерениями](#) [Измерения Cisco MGC](#) [Получение транзакций TCAP](#)
- Если канальный контроллер SS7 не получает сообщение, передаваемое из PGW 2200, проверяет, что TCAP передал сообщение вниз к SCCP. Если уровень TCAP передает сообщение вниз, это может быть, потому что SCCP не имеет достаточной информации для построения соответствующего сообщения SCCP. Это может также быть индикацией, что подсистема SS7 не настроена должным образом или не доступна. Проверьте этот список для проверки: [Конфигурация Точечного кода SS7](#) и [статус Конфигурация подсистемы SS7](#) [Настройка маршрутизации Подсистемы SS7](#) [Локальное и Удаленное состояние SSNB](#) [обслуживании конфигурация](#)

```
(trigger.dat)Проверка системыmm1>rtrv-spc:all MGC-01 - Media Gateway Controller 2004-03-26
13:22:05.492 EST M RTRV "ss7svc1:DPC=001.022.001,DNW=2:OPC=001.001.001:IS"
"ss7svc2:DPC=001.022.002,DNW=2:OPC=001.001.001:IS"
"itusn1:DPC=001.004.001,DNW=2:OPC=001.001.001:IS"
"itusn2:DPC=001.003.001,DNW=2:OPC=001.001.001:IS"
"itusn3:DPC=001.004.001,DNW=2:OPC=001.001.001:IS" ; mm1> prov-rtrv:ss7subsys:NAME="itusn1"
MGC-01 - Media Gateway Controller 2004-03-26 11:48:26.321 EST M RTRV
"session=fix551tgp:ss7subsys" ; mm1> rtrv-lssn:all MGC-01 - Media Gateway Controller 2004-
03-26 11:49:01.985 EST M RTRV "TCAP-01:SSN=12,PST=IS" "TCAP-01:SSN=101,PST=IS" "TCAP-
01:SSN=102,PST=IS" ; mm1> rtrv-rssn:all MGC-01 - Media Gateway Controller 2004-03-26
11:49:04.695 EST M RTRV "scp1:PC=001.004.001,SSN=12,PST=IS"
"scp1:PC=001.004.001,SSN=48,PST=IS" ; mm1> prov-rtrv:inservice:name="finap-initdp" MGC-01 -
Media Gateway Controller 2004-03-29 14:45:25.738 EST M RTRV "session=fix551tgp:inservice" ;
mm1> prov-rtrv:SS7ROUTE:NAME="route4" MGC-01 - Media Gateway Controller 2004-03-30
```

```
11:53:08.493 EST M RTRV "session=fix551tgp:SS7ROUTE" ;
```

- Если вся эта информация, кажется, корректна (как показано в выходных данных, отображенных выше), проверяют тэговые значения, передаваемые вниз от уровня протокола TCAP, такие как SSN, адрес SCCPCalledParty и / или адрес SCCPCallingParty.

## Сообщения TCAP, которые вводят Cisco PGW 2200

Обратная логика может использоваться для отслеживания сообщения SS7, которое входит в Cisco PGW 2200, который предназначен к TCAP / уровень пользователя SCCP стека SS7. Журналы PGW 2200 показывают сообщение SS7, которое входит в каналный контроллер SS7 (от линии SS7) и передается TCAP для обработки. Сообщение сломано на каждом уровне стека SS7. Кроме того, обратите внимание на OPC/DPC, Служебный индикатор (SIO) и выбор звена сигнализации (SLS). OPC и DPC представлены в формате ITU (только в данном примере).

**Совет устранения неполадок:** Проверьте тип сообщения, полученный от линии SS7. Если сообщение UDTS, получают чек, 'возвращают причину'.

Эти выходные данные показывают журнал PGW 2200, когда это получает сообщения SCCP от линии SS7:

**Совет устранения неполадок:** Используйте формат сообщения SCCP, который, как показывают в [Приложении С](#), декодировал тип сообщения, информация заголовка SCCP (показанный в [выходных данных](#) в желтом цвете) и запуск данных TCAP. 1e0002 в выходных данных выше представляет вызывающий адрес (OPC) для сообщения, полученного в PGW, как представлено в dpc.dat. Дамп сообщения SCCP сразу начинается после "0" (начало с типа сообщения SCCP).

Эти выходные данные от журнала PGW 2200, когда это получает TCAP UDTS по SCCP/MTP:

```
Thu Mar 25 18:35:35:385 2004 EST | ss7-i-1 (PID 27288) <Debug>
```

```
CP Received PDU from ssetId 3, chan 0
```

```
Thu Mar 25 18:35:35:385 2004 EST | ss7-i-1 (PID 27288) <Trace>
```

```
PROT_TRACE_MTP3_PDU: Hex dump of MTP3 and UP messages 1d0005 0
```

```
CP DATA IND len: 68 data: 83 09 48 08 a2 0a Thu Mar 25 18:35:35:385 2004 EST | ss7-i-1 (PID 27288) <Debug> >>>> from: 821 to opc 809 (bytes 63) sio 83 sls a: Thu Mar 25 18:35:35:385 2004 EST | ss7-i-1 (PID 27288) <Trace> PROT_TRACE_MTP3_PDU: Hex dump of MTP3 and UP messages 1e0002 0
0a 01 03 0d 11 04 ffffffff8c 09 08 65 0a ffffffff8b 21 08 30 00 18 38 33 44 44 29 62 27 48 01 02 6c
22 ffffffff8a 20 02 01 01 02 01 00 30 18 ffffffff80 04 00 00 00 01 ffffffff82 07 01 10 18 38 33 44 44
fffffff83 07 01 11 07 13 11 00 10 Thu Mar 25 18:35:35:385 2004 EST | TCAP (PID 27283) <Debug> Got
91 bytes from fifo /tmp/sccp_input (fd=16) Thu Mar 25 18:35:35:385 2004 EST | ss7-i-1 (PID
27288) <Debug> RECEIVED SCCP STACK MSG !--- Indicates message is from MTP(SS7 stack). !--- Lines
omitted. Thu Mar 25 18:35:35:385 2004 EST | TCAP (PID 27283) <Debug> 00 01 00 01 1E 00 15 00 00
00 1A 00 00 02 00 00 00 00 00 08 21 00 00 08 09 FFF0A 0A 01 03 0D 11 04 FFF09 08 65 0A FFF21
08 30 00 18 38 33 44 44 29 62 27 48 01 02 6C 22 FFF20 02 01 01 02 01 00 30 18 FFF04 00 00 00 01
FFF07 01 10 18 38 33 44 44 FFF07 01 11 07 13 11 00 10 Thu Mar 25 18:35:35:386 2004 EST | TCAP
(PID 27283) <Debug> ioTcSuIntfc::handleNotInd: Cause =1 Thu Mar 25 18:35:35:386 2004 EST | TCAP
(PID 27283) <Debug> Calling StUiStuDatReq(), spId = 1 Thu Mar 25 18:35:35:386 2004 EST | TCAP
(PID 27283) <Debug> Deleted spDlgEntry 2-69 Thu Mar 25 18:35:35:386 2004 EST | TCAP (PID 27283)
<Debug> Sending msgType 15 to Engine !--- TCAP sends response to Engine which is translated into
L.
```

Эти выходные данные от журнала PGW 2200, когда это получает недопустимое TCAP - сообщение по SCCP / MTP:

## Программное средство трассировки MDL

Cisco PGW 2200 использует триггеры для инициирования транзакции TCAP. Транзакции протокола TCAP используют метод `IN_TRIGGER`, чтобы передать и получить сообщения к и от уровня контроля за TCAP. Когда аналитический результат соответствий вызова вводит 22, протокол TCAP `IN_TRIGGER` инициализируется. Информацией TCAP / сообщения обмениваются между уровнем протокола TCAP (например, триггеры, записанные на языке MDL) и процессом механизма Cisco PGW 2200 с помощью метки, длины, и значения или синтаксиса TLV. Механизм тогда передает информацию к каналному контроллеру TCAP для дальнейшей обработки.

Используйте трассировку Cisco PGW 2200 MDL для наблюдения данных, которые передаются и от уровня протокола TCAP до контроллера TCAP (через механизм). Канальный контроллер TCAP делает необходимую обработку на полученных сообщениях MDL и вперед их к соответствующему IOCC (или TALI-IOCC, IP-IOCC или SS7-IOCC). Механизм также преобразовывает информацию о сообщении TCAP, полученную от канального контроллера TCAP (через SCCP / MTP3) в формат TLV, который можно передать к уровню протокола TCAP, также известному как `IN_TRIGGER`. Для отслеживания TCAP заходят в уровень протокола, выполняют эти шаги:

1. Запустите трассировку MDL.  
`mdl> sta-sc-trc:ss7svc1:log="udts",confirm`
2. Позвоните, который иницирует сервис TCAP (совершает нападки, результат анализа вводят `IN_TRIGGER`).
3. Остановите трассировку MDL.  
`mdl> stp-sc-trc:all MGC-01 - Media Gateway Controller 2004-03-24 17:41:04.702 EST M COMPLD "ALL:Trace stopped for the following files:  
../var/trace/udts_ss7svc2_20040324174103.btr`
4. Выполните `get_trc` для просмотра перехваченной трассировки MDL.  
`get_trc.sh udts_ss7svc2_20040324174103.btr`
5. Выполните **опцию S** для наблюдения 'печати sim' вызова, который показывает поток сообщений между внутренними процессами PGW 2200.
6. Выполните **опцию D** для наблюдения фактической трассировки вызова через код PGW 2200. **Примечание:** Содержание, показанное **опциями D и S** в `get_trc.sh`, может не быть очевидным для понимания, поскольку данные показывают с типами внутренних данных и именами переменной. Однако, описание что искать для отладки транзакций TCAP, показан в **Анализе слежения MDL для раздела TCAP**.

### Анализ слежения MDL для TCAP

Используйте 'печать sim' (опция S `get_trc.sh`) для просмотра полного потока вызовов в уровне протокола Cisco PGW 2200. Печать sim напоминает один показанный в [Приложении D](#). Если это не делает, попытайтесь обратить внимание, куда полученный поток вызовов отличается, и начните устранять неполадки с тем событием. Для устранения проблем TCAP сосредоточьте свое внимание на одном из этих событий.

- LTrigger
- LTriggerInformation
- LTriggerNext
- LTriggerRelease

Это внутренние события, которые ведут механизм состояний `IN_TRIGGER`.

Используйте трассировку Cisco PGW 2200 MDL, чтобы видеть, что фактический код течет для каждого из этих событий. Результаты LTrigger в ВЫХОДНОМ `IN_TRIGGER` и другие три

передаются полученные IN\_TRIGGER ВХОДНЫМ сообщением IN\_TRIGGER от механизма.

## Исходящие сообщения TCAP

Для определения сообщений, которые входят и из MDL для TCAP ищите IN\_TRIGGER в трассировке MDL. [Типовой Синтаксис IN\\_TRIGGER от Трассировки MDL](#) графические показы отосланное сообщение и один полученный в MDL к и от механизма. OUTPUT указывает, что IN\_TRIGGER отправил запрос для Механизма для передачи сообщения TCAP.

## Советы устранения неполадок

- Используйте трассировку MDL, чтобы проверить, что сообщение TRIGGER передавалось механизму, если не передавались IN\_TRIGGER или OUTPUT.
- Проверьте dialplan для конфигурации результата IN\_TRIGGER.
- Проверьте штатное и / или конфигурация trigger.dat.
- Проверьте, что сообщение передавалось из канального контроллера SS7. Если сообщение никогда не делало его из канального контроллера SS7, это - результат канального контроллера SCCP, не имеющего достаточно информации, чтобы направить вызов или создать допустимое сообщение.
- Проверьте конфигурация SS7\_SUBSYSTEM и конфигурация SCCP.
- Проверьте состояние SSN.
- Проверьте статус ПК.

Если выходные данные IN\_TRIGGER успешны, трассировка Cisco PGW 2200 MDL отображает ответ на то сообщение как INPUT в IN\_TRIGGER.

## Произведите выборку синтаксиса IN\_TRIGGER от трассировки MDL

Сообщение INPUT является ответом от механизма в отношении запроса (или сообщение OUTPUT) передаваемый от протокола TCAP. Механизм может ответить от своего собственного имени или от имени уровня TCAP.

Сообщение IN\_TRIGGER указывает, что MDL передает TCAP / данные SCCP вниз к механизму и канальным контроллерам, которые будут использоваться для построения сообщения UDT, которое отослано на ЛИНИИ в SCP. Информация, передаваемая вниз механизму, получена из файла trigger.dat, и это показывает непосредственно выше выходных данных этого сообщения. Для наблюдения содержания этого сообщения как, MDL создал его, перейдите из текста IN\_TRIGGER. Запуск процедуры создания сообщения обозначен SendMessage()..., как показано здесь.

## Советы устранения неполадок

- Если запрос TCAP передается из Cisco PGW 2200 с неверными данными, трассировка MDL может использоваться для наблюдения точно, где Cisco PGW 2200 получил свою информацию. Большая часть информации прибывает из файла trigger.dat. Видеть, где Cisco PGW 2200 получил свою информацию для исходящего сообщения, поиск (от IN\_TRIGGER) для рассматриваемого элемента TCAP. Например, если тип TCAP неправильно закодирован, ищите строку tcapType в трассировке MDL (вокруг writing field tcapType).
- Для наблюдения где trigger.dat чтений Cisco PGW 2200 для кодирования содержания TCAP ищите строки, показанные в этой таблице. Эти строки представляют вызовы



процедуры, используемые для получения информации о trigger.dat. Эти вызовы процедуры должны произойти между событием INPUT LTrigger и рассматриваемым сообщением OUTPUT IN\_TRIGGER.

Name	Описание	Строка поиска MDL
TT	Триггерная запись таблицы	GetTT
MA	Запись действия сообщения	GetMA
MS	Сообщение, передающее запись	GetMS
OC	Передача операции	GetOS
PS	Запись передачи параметра	GetPS
RR	Запись полученного ответа	GetRR
MR	Сообщение, получающее запись	GetMR
Или	Получение операции	GetOR
PR	Запись получения параметра	GetPR
PA	Запись действия ответа	GetRA
AD	Данные действий	GetAD

## Входящие сообщения TCAP

Сообщение INPUT является ответом от механизма в отношении запроса. Механизм может ответить от своего собственного имени или от имени уровня TCAP. Входящее сообщение определено строкой сообщения INPUT IN\_TRIGGER в трассировке Cisco PGW 2200 MDL как показано в выходных данных данного примера. Данный пример также показывает сообщение, которое декодируется. Это полезно, если необходимо определить какие-либо проблемы, которые могут существовать с ответом TCAP.

Для декодирования Сообщения ядра, полученного Cisco PGW 2200 MDL, используйте тот же формат TLV, описанный ранее в этом документе. Они обмениваются сообщениями, сразу декодируются после текста, INPUT IN\_TRIGGER.

```
INPUT "IN_TRIGGER": 00 00 00 02 00 00 00 69 00 02 0d 00 12 00 04
00 00 08 21 00 11 00 04 00 00 00 02 00 10 00 12 00 00 00 08 21 0c 01 67 02
04 50 00 00 00 00 00
```

```
08 09 00 13 00 0d 03 00 2a 81 76 82 15 01 01 01 01 00 01 00 05 00 01 01
00 06 00 03 01 00 17 00 07 00 01 04 00 09 00 0f a0 0d 30 0b 80 01 0a 81
01 00 a2 03 80 0
```

```
1 01 00 05 00 01 01 00 06 00 03 01 00 23 00 07 00 01 05 00 09 00 1a 80
10 30 0e a0 0c a0 0a a1 05 a0 03 81 01 06 82 01 0a 81 01 01 a2 03 80 01
01 00 0a 00 00
```

reading element header: TcapMessageStyle

```
reading field callRef
!--- Identifies call reference for MDL / engine Xaction. '0000 0000 0000 0000 0000 0000 0000
0010'B ok reading field processed !--- Identifies process ID for MDL/engine Xaction. '0000 0000
0000 0000 0000 0000 0110 1001'B ok reading field msgType !--- Identifies message type for
MDL/engine Xaction. '0000 0000 0000 0010'B !--- Message type 2 = ITU CONTINUE. ok reading field
tagCount !--- Identifies the number of tags included in this message. '0000 1101'B 13 0x0d ok ok
reading element _Continue !--- TCAP message type. reading field RAW 1136 bits read ok reading
field DATA reading element header: TcapElementStyle !--- Tag element #1. reading field ieId !---
Tag element #1 TAG ID. '0000 0000 0001 0010'B ok reading field ieLength !--- Tag element #1 Tag
Length. '0000 0000 0000 0100'B !--- 4 bytes. ok ok reading element TcapDatabaseIdElem reading
field RAW 32 bits read ok reading field DATA !--- Tag element #1 data portion begins. '0000
0000'B 0 0x00 !--- Byte 1. '0000 0000'B 0 0x00 !--- Byte 1. '0000 1000'B 8 0x08 !--- Byte 1.
'0010 0001'B 33 0x21 "!" !--- Byte 1. 'B ok ok reading element header: TcapElementStyle !---
Tag element #2. reading field ieId
```

Это - пример выходных данных входящего отклика к сообщению UDTs:

```
INPUT "IN_TRIGGER": 00 00 00 02 00 00 00 69 00 0f 02 00 0b
00 01 01 00 0a 00 00
```

reading element header: TcapMessageStyle

```
reading field callRef
'0000 0000 0000 0000 0000 0000 0000 0010'B
ok
reading field processId
'0000 0000 0000 0000 0000 0000 0110 1001'B
ok
reading field msgType
!--- Message type - Information message. '0000 0000 0000 1111'B ok reading field tagCount '0000
0010'B 2 0x02 ok ok reading element _Information reading field RAW 72 bits read ok reading field
DATA reading element header: TcapElementStyle reading field ieId '0000 0000 0000 1011'B ok
reading field ieLength '0000 0000 0000 0001'B ok ok reading element TcapErrorElem !--- TCAP
error element. reading field RAW 8 bits read ok reading field DATA reading field octet1 reading
field error '0000 0001'B 1 0x01 !--- TCAP error element = 01 -> TCAP_ERROR_SSN_OOS. ok ok ok ok
ok ok Continuing State Machine: IN_TRIGGER (105) STATE * INPUT Information AS <messageData>
CC.db.nonEssentialData.TCAPTransactionUnixEndTimeElem.DATA :=
MGetTime(CC.db.nonEssentialData.TCAPTransactionMsecEndTimeElem.DATA) -> 1080257735
```

Другая ценная часть информации, которую можно получить из трассировки Cisco PGW 2200 MDL (для вызовов TCAP) является оценкой причины LTriggerRelease. INErrorElem, закодированный в LTriggerRelease также, предоставляет понимание, почему вызов или транзакция TCAP не работают как ожидалось. Посмотрите этот рисунок MDL Cisco PGW 2200, который показывает LTriggerRelease, который отослан в ответ на начальное событие LTrigger, полученное IN\_TRIGGER. См. [Приложение E](#) для подробных данных о событиях IN\_TRIGGER и значениях INErrorElem.

## [Приложение A: Метки MDL](#)

Метки Cisco PGW 2200 MDL обмениваются между Cisco PGW 2200 MDL и механизмом. Это Приложение описывает заказ, содержание и формат всех меток, используемых в

транзакциях TCAP. Информация, используемая для начальной загрузки этих, которые оценивает метка, получена из контекста вызова и оценивает заполненный в файле trigger.dat. Триггерный файл также используется для указания на то, что должно быть передано / от механизма для здания сообщения TCAP и что должно быть получено от механизма для обработки сообщения TCAP, когда получен ответ.

Эти метки используются для обработки вызовов TCAP:

- **ID 1 МЕТКИ – тип TCAP**  
**Описание:** Индикация относительно типа MDL TCAP  
**Длина данных:** исправленный (1)  
**Формат данных:** 1 = ETSI 300 374-1

2 = Bell Core GR-1298-CORE  
TR-NWT-001284  
TR-NWT-001285

3 = Bell Core Pre AIN  
GR-1428-CORE

- **ID 2 МЕТКИ – назначение системы**  
**Описание:** Внутренний получатель события  
**Длина данных:** исправленный (1)  
**Формат данных:** октет  
**Содержание:** 0 = внутренний SCP, 1 = TCAP Trillium
- **ID 3 МЕТКИ – адрес вызова SCCP**  
**Описание:** Данные SCCP требуются триллиумом  
**Длина данных:** Переменная  
**Формат данных:** Octet 1 Routing Indicators

Bit A 0 - Route by GT, 1 - Route by SSN  
Bit B DPC is present (Octets 2 to 4 have valid data)  
Bit C SSN is present (Octet 5 has valid data)

Octet 2 DPC Network  
Octet 3 DPC Cluster  
Octet 4 DPC Member  
Octet 5 Called SSN  
Octet 6 GTFormat

0 - No global Title Included  
1 - Global Title includes nature of address indicator only (ITU)  
- Global title includes translation type,  
numbering plan and encoding scheme.(ANSI)  
2 - Global Title Includes translation type only.(ITU/ANSI)  
3 - Global title includes translation type,  
numbering plan and encoding scheme.  
(ITU). - not used in ANSI.  
4 - Global Title includes translation type, numbering plan,  
encoding scheme and nature of address digits.  
(ITU). - Not used in ANSI.

Octet 7 Translation Type Value  
Octet 8 Numbering Plan  
0 - Unknown  
1 - ISDN Telephony  
2 - Telephony  
3 - Data  
4 - Telex  
5 - Maritime Mobile  
6 - Land Mobile  
7 - ISDN Mobile

Octet 9 Nature Of Number  
1 - Subscriber Number

- 2 - National Number
- 3 - International Number

Octet 10 Number Of Digits in octets 11 to 43

Octet 11 to 43  
Digits in IA5 format

- **ID 4 МЕТКИ – вызывающий адрес SCCP** Описание: Данные SCCP требуются триллиумом  
Длина данных: Переменная  
Формат данных: Octet 1 Routing Indicators

Bit A 0 - Route by GT, 1 - Route by SSN  
Bit B DPC is present (Octets 2 to 4 have valid data)  
Bit C SSN is present (Octet 5 has valid data)

Octet 2 DPC Network

Octet 3 DPC Cluster

Octet 4 DPC Member

Octet 5 Calling SSN

- **ID 5 МЕТКИ – тип компонента TCAP** Описание: Тип компонента TCAP  
Длина данных: исправленный (1)  
Формат данных: Octet

0 = Unknown  
1 = Invoke  
2 = Return Result Last  
3 = Return Error  
4 = Reject  
5 = Return Result Not Last  
6 = Invoke Last  
7 = Invoke Not Last

- **ID 6 МЕТКИ – код операции TCAP** Описание: Код операции сообщения TCAP  
Длина данных: Переменная (Всегда 4 для ANSI)  
Формат данных: Octet 1 Flag

0 = None  
1 = Local  
2 = Global  
3 = National  
4 = Private

Octet 2 Operation Class

Octet 3 Op Code Highest byte (ITU) Family (ANSI)

Octet 4 Op Code Next byte (ITU) Specifier (ANSI)

Octet n Op Code Least byte (ITU)

- **ID 7 МЕТКИ – TCAP вызывает ID** Описание: ID компонента  
Длина данных: исправленный (1)  
Формат данных: октет
- **ID 8 МЕТКИ – идентификатор корреляции TCAP** Описание: ID компонента, к которому коррелирует этот компонент  
Длина данных: исправленный (1)  
Формат данных: октет
- **ID 9 МЕТКИ – ANSI диалогового компонента TCAP** Описание: Тело сообщения TCAP от первого параметра и далее  
Длина данных: Переменная  
Формат данных: октет
- **ID 10 МЕТКИ – маркер конца диалога TCAP** Описание: Тело сообщения TCAP от первого параметра и далее (ПОСЛЕДОВАТЕЛЬНОСТЬ)  
Длина данных: исправленный (0)  
Формат данных: Нет
- **ID 11 МЕТКИ – ошибка** Описание: Данные об ошибках  
Длина данных: исправленный (1)  
Формат данных: октет  
Содержание: 1 = TCAP\_ERROR\_SSN\_OOS  
2 = TCAP\_ERROR\_PC\_UNAVAILABLE

3 = TCAP\_ERROR\_SERVICE\_NOT\_RESPONDING  
 4 = TCAP\_TRIGGER\_TIMEOUT

- **ID 12 МЕТКИ – индекс группы STP-SCP**  
**Описание:** Индекс группы STP-SCP, данные прошли от анализа.  
**Длина данных:** исправленный (1)  
**Формат данных:** октет  
**Содержание:** Значение индекса группы STP-SCP.
- **ID 13 МЕТКИ – транспортный протокол TCAP**  
**Описание:** Тип транспортного протокола  
**Длина данных:** исправленный (1)  
**Формат данных:** октет  
**Содержание:** 1 = TCAP\_TRANSPORT\_SCCP  
 2 = TCAP\_TRANSPORT\_TCP\_IP
- **ID 14 МЕТКИ – внешняя ошибка TCAP / проблема**  
**Описание:** Ошибка или проблемное значение, полученное или передаваемое по ошибке и компоненты Результата  
**Длина данных:** Переменная  
**Формат данных:** октет
- **ID 15 МЕТКИ – тип корпуса TCAP**  
**Описание:** Тип тела компонента  
**Длина данных:** исправленный (1)  
**Формат данных:** октет  
**Содержание:** 1 = TCAP\_BODY\_SEQUENCE  
 2 = TCAP\_BODY\_SET
- **ID 16 МЕТКИ – Информация по диалогу TCAP**  
**Описание:** TCAP Trillium включает эту МЕТКУ во все сообщения, передаваемые MDL. MDL Должен хранить эту информацию и передать ее к TCAP Trillium во всех последующих сообщениях для диалогового окна или однонаправленных сообщениях, отнесенных к вызову.  
**Длина данных:** Переменная  
**Формат данных:** октет
- **ID 17 МЕТКИ – идентификатор транзакции TCAP**  
**Описание:** TCAP Trillium включает эту МЕТКУ во все сообщения, передаваемые MDL. MDL Должен хранить эту информацию для передачи к CDB.  
**Длина данных:** Переменная  
**Формат данных:** октет
- **ID 18 МЕТКИ – идентификатор базы данных TCAP**  
**Описание:** TCAP Trillium будет включать эту МЕТКУ во все сообщения, передаваемые MDL. MDL Должен хранить эту информацию для передачи к CDB.  
**Длина данных:** Переменная  
**Формат данных:** октет

## Приложение Б: Точечные коды SS7 выхода

ETSI PC 1-1-1 (padded to 16 bits) = 00001000 00001001 = 08 09 = 809 (shown in log)  
 ETSI PC 1-4-1 (padded to 16 bits) = 00001000 00100001 = 08 21 = 821 (shown in log)  
 ETSI PC 3-3-3 (padded to 16 bits) = 00011000 00011011 = 18 1B = 181b (another ex.)

	Кластер	Сеть	Member	Точечный код
ETSI (14 битов)	3 бита	8 битов	3 бита	14 битов
ANSI (24 бита)	8 битов	8 битов	8 битов	24 бита
ПК 1-1-1 (никакое заполнение, только 14 битов)	001	000 00001	001	001000 = 8 00000001 = 01
ПК 1-4-1 (никакое заполнение, только 14 битов)	001	000001 00	001	001000 = 8 00100001 = 21

14 битов)				
ПК 3-3-3	011	000000 11	011	011000 = 18 00011011 = 1В

## Приложение В: Типы сообщения SCCR

Тип сообщения	Код типа сообщения
Запрос подключения CR	0000 0001
Подтверждение соединения CC	0000 0010
Подключение CREF отказалось	0000 0011
Освобожденный RLSD	0000 0100
Завершенный Выпуск RLC	0000 0101
Форма данных DT1 1	0000 0110
Форма данных DT2 2	0000 0111
Подтверждение приема данных АК	0000 1000
Unitdata UDT	0000 1001
Сервис Unitdata UDTS	0000 1010
Срочные данные ED	0000 1011
Подтверждение Срочных данных EA	0000 1100
Запрос Сброса RSR	0000 1101
Подтверждение сброса RSC	0000 1110
Ошибка Протокольной информационной единицы ERR	0000 1111
Тест Бездействия IT	0001 0000
XUDT Расширенный unitdata	0001 0001
XUDTS Расширенный сервис unitdata	0001 0010
LUDT Длинный unitdata	0001 0011
LUDTS Длинный сервис unitdata	0001 0100

### Unitdata (UDT)

Сообщение UDT содержит:

- Три указателя
- Параметры обозначены в этой таблице.

Параметр	Ссылка Вопрос. . 713	Введите (F V O)	Длина (октеты)
Тип сообщения	2.1	_____ F	1
Класс протокола	3.6	_____ F	1

Адрес вызываемой стороны	3.4	V	3 минимума
Адрес вызывающей стороны	3.5	V	3 минимума
Данные	3.16	V	2-Х (Примечание 1)

**Примечание:** Из-за продолжающихся исследований адреса вызываемых и вызывающих сторон SCCP, максимальной длине этого параметра нужно дальнейшее исследование. Также обращено внимание, что передача до 255 октетов пользовательских данных позволена, когда адрес вызываемых и вызывающих сторон SCCP не включает глобальное название.

### Сервис unitdata (UDTS)

Сообщение UDTS содержит:

- Три указателя.
- Параметры обозначены в этой таблице.

Параметр	Ссылка Вопрос. 713	Введите (F V O)	Длина (октеты)
Тип сообщения	2.1	_____ F	1
Возвратите причину	3.12	_____ F	1
Адрес вызываемой стороны	3.4	V	3 минимума
Адрес вызывающей стороны	3.5	V	3 минимума
Данные	3.16	V	2-Х (Примечание)

**Примечание:** Из-за продолжающихся исследований адреса вызываемых и вызывающих сторон SCCP, максимальной длине этого параметра нужно дальнейшее исследование. Также обращено внимание, что передача до 255 октетов пользовательских данных позволена, когда адрес вызываемых и вызывающих сторон SCCP не включает глобальное название.

Эта таблица показывает типовой отказ сообщения SCCP для Unitdata / сервис Unitdata:

Параметр	Введите (F V O)	Длина (октеты)	Исходящее сообщение	Входящее сообщение

			корреляц ии	корреляц ии
Тип сообщения	____ _ F	1	09	0a
Класс протокола	____ _ F	1	80	01
Указатель адреса вызываемо й стороны	____ _ F	1	03	03
Указатель адреса вызывающ ей стороны	____ _ F	1	07	0d
Указатель на данные	____ _ F	1	0b	11
Адрес вызываемо й стороны	V	3 минимума	04 c3 21 08 0c	04 c3 ... 30 00
Адрес вызывающ ей стороны	V	3 минимума	04 c3 09 08 67	18 38 33 44 44
Данные (ДАННЫЕ ТСАР)	V	04 Данных (данные ТСАР) c3 09 08 67 18 38 33 44 44 V	52 62 ... 20 00	29 62 ... 00 10

**Примечание:** Эти сообщения являются примерами только и могут не отразить фактическое сочетание запросов и ответов / последовательность.

## UDTS возвращает причины

В сервисе Unitdata возвращается Расширенный сервис Unitdata или Длинное служебное сообщение Unitdata, ", причиной" поле parameter является одно поле октета, которое содержит причину для сообщения, возвращаются. Биты 1 - 8 закодированы как показано здесь:

```
Value Bits
0 0 0 0 0 0 0 0 0 no translation for an address of such nature
1 0 0 0 0 0 0 0 1 no translation for this specific address
2 0 0 0 0 0 0 1 0 subsystem congestion
3 0 0 0 0 0 0 1 1 subsystem failure
4 0 0 0 0 0 1 0 0 unequipped user
5 0 0 0 0 0 1 0 1 MTP failure
6 0 0 0 0 0 1 1 0 network congestion
7 0 0 0 0 0 1 1 1 unqualified
8 0 0 0 0 1 0 0 0 error in message transport (Note)
9 0 0 0 0 1 0 0 1 error in local processing (Note)
```



```
10 0 0 0 0 1 0 1 0 destination cannot perform reassembly (Note)
11 0 0 0 0 1 0 1 1 SCCP failure
12 0 0 0 0 1 1 0 0 hop counter violation
13 0 0 0 0 1 1 0 1 segmentation not supported
14 0 0 0 0 1 1 1 0 segmentation failure
15 0 0 0 0 1 1 1 1
```

to

```
228 1 1 1 0 0 1 0 0 Reserved for International Use
229 1 1 1 0 0 1 0 1
```

to

```
254 1 1 1 1 1 1 1 0 Reserved for National Networks
255 1 1 1 1 1 1 1 1 Reserved
```

## Приложение D: Интерфейс MDL для сообщения TCAP

Все сообщения придерживаются общего формата TLV:

- **Экземпляр вызова и ProcessId** - 8 байтов длиной и должны быть получены Механизмом и возвращены в ответном сообщении из неизменного Механизма.
- **Идентификатор сообщения** - Определяет сообщение, которое передано или получено уровнем протокола TCAP (значения, показанные в этой [таблице](#)).
- **Тэговый Идентификатор** - Количество меток и данных метки (ID метки, длина данных и данные) диктует то, что отослано в сообщении TCAP удаленному назначению. Все размеры поля исправлены за исключением поля данных элемента метки, длина которого является переменной и определена (в октетах) по условию длина. Каждое из полей Total Length, Экземпляра вызова и Идентификатора Процесса, Идентификатора сообщения, Идентификатора Метки и Длины данных передано наиболее значимым байтом сначала.

## Приложение E: Внутренний интерфейс MDL

Внутренне, связь с Объектами государственного аппарата TCAP (SMOs) через сигналы с данными. Любой тип данных MDL может быть передан с сигналом. Названия и значения сигналов и данных перечислены здесь.

- **LTriggerОписание:** Это - первый сигнал, что LCM передает к TCAP для начала диалога. В Стремительности INTriggerElem также содержит stpScpGroupIndex.  
MESSAGE\_ACTION\_COPY\_STP\_SCP\_INDEX\_FROM\_SIGNAL\_DATA должен собираться в таблице MA для этого использоваться. **Компоненты:** INTriggerElem, BNumberElem, BNumberDataElem
- **LTriggerInformationОписание:** Когда диалог продолжается, этот сигнал передается от TCAP до LCM в ответ на LTrigger. **Компоненты:** INTriggerElem, BNumberElem, BNumberDataElem
- **LTriggerNextОписание:** Этот сигнал передается от LCM до TCAP как последующий триггерный запрос в существующем диалоге. **Компоненты:** INTriggerElem, BNumberElem, BNumberDataElem
- **LTriggerReleaseОписание:** Этот сигнал является последним, чтобы быть переданным или от LCM или от TCAP и может быть передан от TCAP в ответ на LTrigger после того, как ответ был получен от SCP. **Компоненты:** INErrorElem, BNumberElem,

**BNumberDataElem**INErrorElem имеет эти значения:1 TRIG\_ERROR\_NONE,

- 2 TRIG\_EXIT\_UNABLE\_TO\_COMPLETE\_MA\_IS\_LNP\_M\_BIT\_CLEAR,
- 3 TRIG\_ERROR\_NULL\_TRIGGER,
- 4 TRIG\_ERROR\_TRIGGER\_TABLE\_NOT\_FOUND,
- 5 TRIG\_ERROR\_UNKNOWN\_MESSAGE\_ACTION,
- 6 TRIG\_ERROR\_UNKNOWN\_RESPONSE\_ACTION,
- 7 TRIG\_ERROR\_UNKNOWN\_PARAMETER\_ACTION,
- 8 TRIG\_ERROR\_MESSAGE\_ACTION\_FAILED,
- 9 TRIG\_ERROR\_UNABLE\_TO\_LOAD\_DIALOGUE\_COMPONENT,
- 10 TRIG\_ERROR\_UNABLE\_TO\_LOAD\_TAG,
- 11 TRIG\_ERROR\_READING\_TT,
- 12 TRIG\_ERROR\_READING\_MA,
- 13 TRIG\_ERROR\_READING\_PS,
- 14 TRIG\_ERROR\_READING\_RR,
- 15 TRIG\_ERROR\_READING\_PR,
- 16 TRIG\_ERROR\_READING\_RA,
- 17 TRIG\_ERROR\_ACTION\_NOT\_COMPATIBLE\_IN\_PR,
- 18 TRIG\_ERROR\_NO\_ACTION\_DATA\_FOR\_ACTION\_RE\_TRIGGER,
- 19 TRIG\_ERROR\_NO\_ACTION\_DATA\_FOR\_ACTION\_SEND\_ACTION\_TO\_LCM,
- 20 TRIG\_ERROR\_UNKNOWN\_MESSAGE\_IN\_MS,
- 21 TRIG\_ERROR\_UNKNOWN\_PR\_ACTION,
- 22 TRIG\_ERROR\_UNABLE\_TO\_COMPLETE\_MA\_COPY\_SCCP\_GT\_FROM\_BNUMBER,
- 23 TRIG\_ERROR\_UNABLE\_TO\_COMPLETE\_MA\_COPY\_STP\_SCP\_INDEX\_FROM\_SIGNAL\_DATA,
- 24 TRIG\_ERROR\_UNKNOWN\_DIALOGUE\_COMPONENT,
- 25 TRIG\_ERROR\_SIGNAL\_IN\_WRONG\_STATE,
- 26 TRIG\_ERROR\_SCCP\_TIMEOUT,
- 27 TRIG\_ERROR\_IN\_RESPONSE\_OPERATION\_CODE\_MISSING,
- 28 TRIG\_ERROR\_IN\_RESPONSE\_INVOKE\_ID\_IN\_USE,
- 29 TRIG\_ERROR\_IN\_RESPONSE\_INVOKE\_ID\_NOT\_FOUND,
- 30 TRIG\_ERROR\_IN\_RESPONSE\_CORROLATION\_ID\_NOT\_FOUND,
- 31 TRIG\_ERROR\_IN\_RESPONSE\_UNEXPECTED\_CORROLATION\_ID,
- 32 TRIG\_ERROR\_IN\_RESPONSE\_NO\_COMPONENT\_CONTENTS,

33 TRIG\_ERROR\_IN\_RESPONSE\_INVALID\_COMPONENT\_CONTENTS ,  
34 TRIG\_ERROR\_IN\_RESPONSE\_UNEXPECTED\_INVOKE\_ID ,  
35 TRIG\_ERROR\_IN\_RESPONSE\_EXTERNAL\_ERROR\_NOT\_FOUND ,  
36 TRIG\_ERROR\_ABORT ,  
37 TRIG\_ERROR\_USER\_ABORT ,  
38 TRIG\_ERROR\_PROTOCOL\_ABORT ,  
39 TRIG\_ERROR\_UNKNOWN

## Дополнительные сведения

- [Технические примечания программного коммутатора Cisco PGW 2200](#)
- [Поддержка голосовых технологий](#)
- [Поддержка продуктов Голосовой и Унифицированной связи](#)
- [Устранение неполадок в системах IP-телефонии Cisco](#)
- [Cisco Systems – техническая поддержка и документация](#)