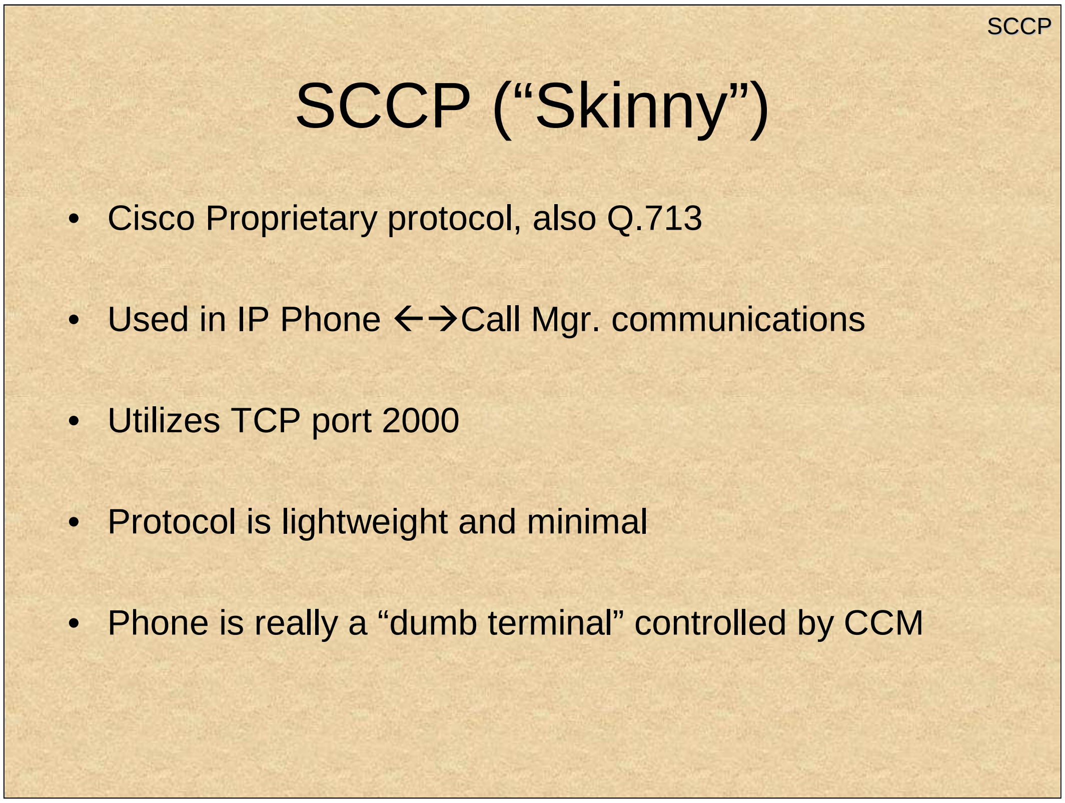
Table of Contents:

About the Author:

Jonathan Levin specializes in training and consulting services. This, and many other training materials, are created and constantly updated to reflect the ever changing environment of the IT industry.

To report errata, or for more details, feel free to email **JL@HisOwn.com**

This material is protected under copyright laws. Unauthorized reproduction, alteration, use in part or in whole is prohibited, without express permission from author. I put a LOT of effort into my work (and I hope it shows). Respect that.



SCCP

SCCP (“Skinny”)

- Cisco Proprietary protocol, also Q.713
- Used in IP Phone ←→Call Mgr. communications
- Utilizes TCP port 2000
- Protocol is lightweight and minimal
- Phone is really a “dumb terminal” controlled by CCM

The Cisco “Skinny” protocol was originally developed by the Selsius Corporation. With their acquisition by Cisco, this became a Cisco propietary protocol, that is used in the communication between the Cisco IP Phones (mostly 79xx) and the Cisco Call Manager.

The protocol is a very lightweight one (hence the nickname “Skinny”). The Call Manager does all the H.323 and SIP processing, acting as a proxy, leaving the IP Phone the task of processing the VoIP RTP datastream.

The protocol is rather scarcely documented, as full documentation is available only to Cisco affiliates. The rest of this section attempts to explain this protocol, thanks to a lot of research, packet captures, and common sense.

SCCP

SCCP ("Skinny") Messages

(in order of appearance)

Stage I – Phone/CallMgr registration

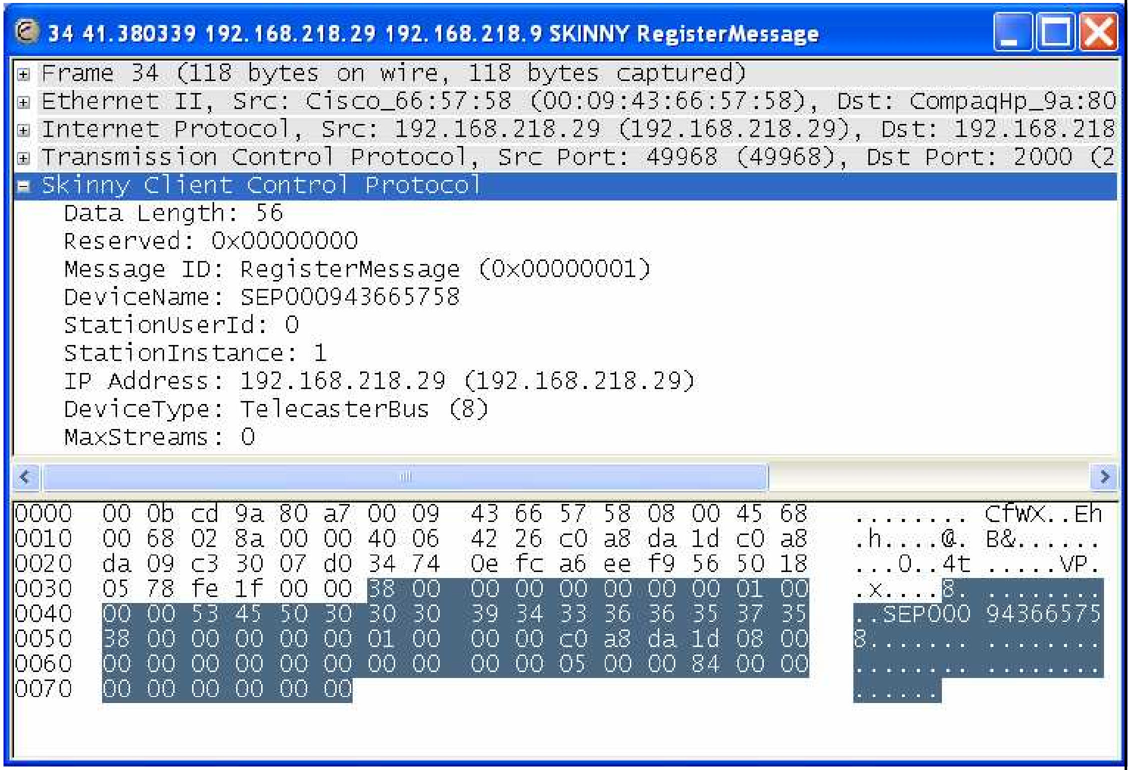
	Msg	Usage	Data
→	0001	RegisterMessage	Device Name, Station UserID & Instance, IP Address, Device Type, Max Streams
→	0002	IPPortMessage	IP and Port Terminal is listening on
←	0081	RegisterAckMessage	Keep Alive Interval, Date Template (M/D/YA), Secondary Keep Alive Interval
←	009B	CapabilitiesRequest	Call Mgr asks for Station capabilties
→	0010	CapabilitiesResponse	CapCount capabilities(PayLoad/MaxFramesPerPacket)
→	000F	VersionRequest	Station requests Call Mgr version
←	0098	VersionResponse	Call Mgr Version
→	000E	ButtonTemplateRequest	--
←	0097	ButtonTemplateMessage	Button offset/count and 40-something button defs
→	000D	TimeDateRequest	--
←	0094	DefineTimeDate	Y/M/WD/D, Hour/Min/Sec/mSec, 32-bit TimeStamp

→: Phone to Call Mgr ←: Call Mgr to phone

The table above shows the SCCP message type, as they "appear" in the lifespan of a telephone. In particular, this table shows the phone registration process with the call manager.

The phone registers its IP, as well as its type and name. The CCM asks it to provide its "capabilities" (voice/video codecs supported). It then caches the IP-Phone capabilities and translates them to H.323 capabilities.

The illustration to the right depicts a typical Registration message, as captured by Ethereal's protocol dissector.

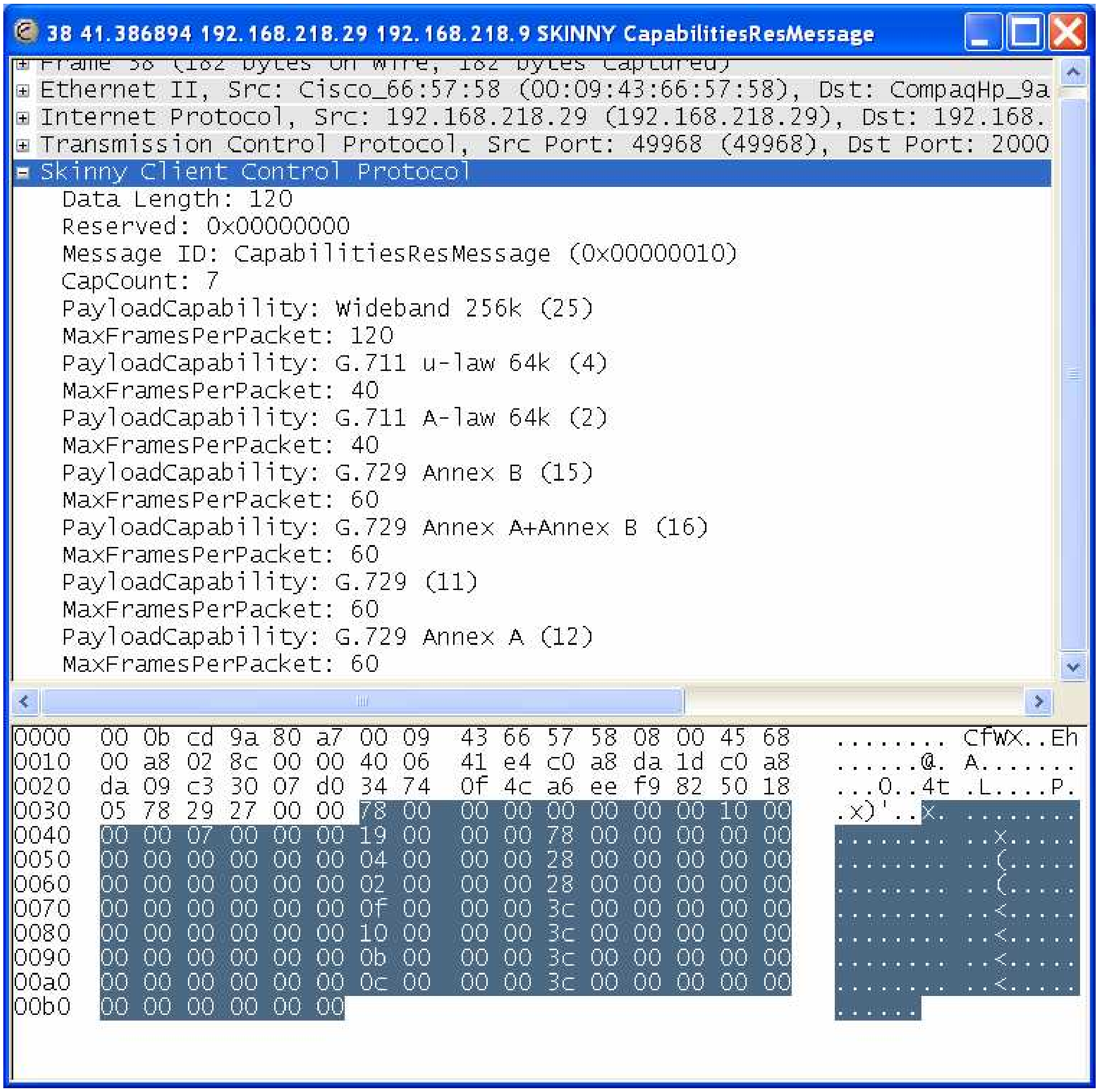


34 41.380339 192.168.218.29 192.168.218.9 SKINNY RegisterMessage

```
⊞ Frame 34 (118 bytes on wire, 118 bytes captured)
⊞ Ethernet II, Src: Cisco_66:57:58 (00:09:43:66:57:58), Dst: CompaqHp_9a:80
⊞ Internet Protocol, Src: 192.168.218.29 (192.168.218.29), Dst: 192.168.218
⊞ Transmission Control Protocol, Src Port: 49968 (49968), Dst Port: 2000 (2
⊟ Skinny Client Control Protocol
    Data Length: 56
    Reserved: 0x00000000
    Message ID: RegisterMessage (0x00000001)
    DeviceName: SEP000943665758
    StationUserId: 0
    StationInstance: 1
    IP Address: 192.168.218.29 (192.168.218.29)
    DeviceType: TelecasterBus (8)
    MaxStreams: 0
```

```
0000  00 0b cd 9a 80 a7 00 09  43 66 57 58 08 00 45 68   ........ CfWX..Eh
0010  00 68 02 8a 00 00 40 06  42 26 c0 a8 da 1d c0 a8   .h....@. B&......
0020  da 09 c3 30 07 d0 34 74  0e fc a6 ee f9 56 50 18   ...0..4t .....VP.
0030  05 78 fe 1f 00 00 38 00  00 00 00 00 00 00 01 00   .x....8. ........
0040  00 00 53 45 50 30 30 30  39 34 33 36 36 35 37 35   ..SEP000 94366575
0050  38 00 00 00 00 00 01 00  00 00 c0 a8 da 1d 08 00   8....... ........
0060  00 00 00 00 00 00 00 00  00 00 05 00 00 84 00 00   ........ ........
0070  00 00 00 00 00 00                                  ......
```

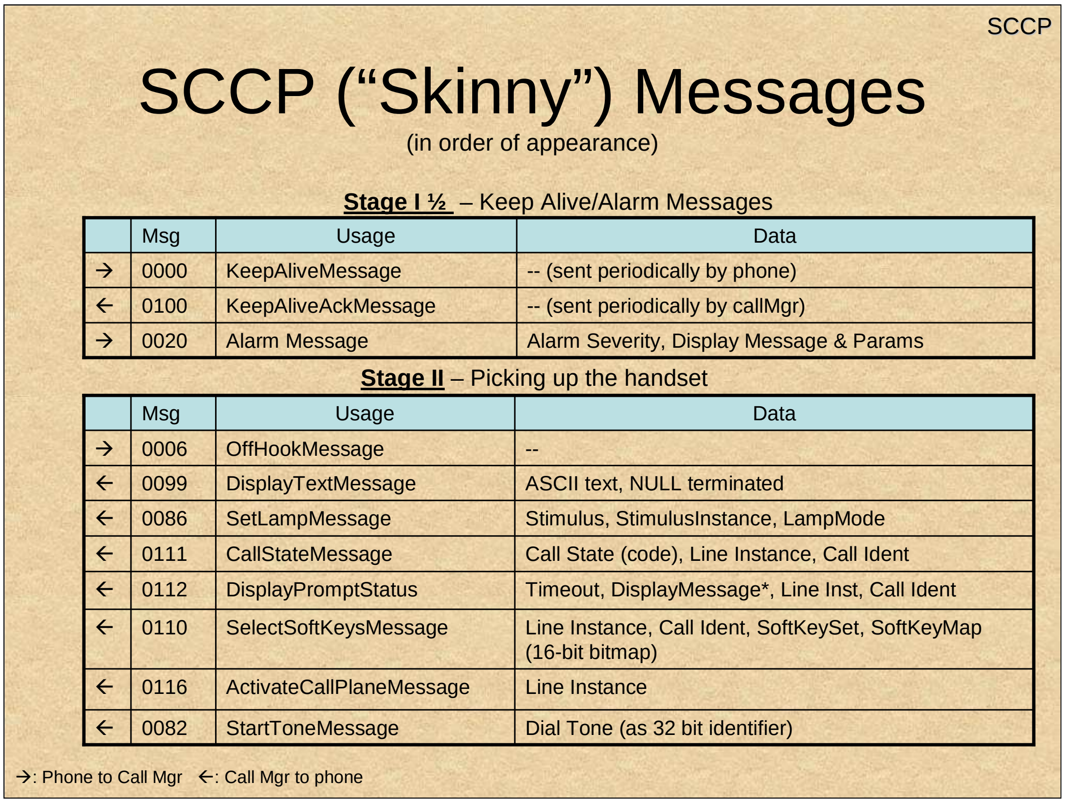
The Capabilities Response message is shown in the following illustration:



38 41.386894 192.168.218.29 192.168.218.9 SKINNY CapabilitiesResMessage

```
⊞ Frame 38 (182 bytes on wire, 182 bytes captured)
⊞ Ethernet II, Src: Cisco_66:57:58 (00:09:43:66:57:58), Dst: CompaqHp_9a
⊞ Internet Protocol, Src: 192.168.218.29 (192.168.218.29), Dst: 192.168.
⊞ Transmission Control Protocol, Src Port: 49968 (49968), Dst Port: 2000
⊟ Skinny Client Control Protocol
    Data Length: 120
    Reserved: 0x00000000
    Message ID: CapabilitiesResMessage (0x00000010)
    CapCount: 7
    PayloadCapability: Wideband 256k (25)
    MaxFramesPerPacket: 120
    PayloadCapability: G.711 u-law 64k (4)
    MaxFramesPerPacket: 40
    PayloadCapability: G.711 A-law 64k (2)
    MaxFramesPerPacket: 40
    PayloadCapability: G.729 Annex B (15)
    MaxFramesPerPacket: 60
    PayloadCapability: G.729 Annex A+Annex B (16)
    MaxFramesPerPacket: 60
    PayloadCapability: G.729 (11)
    MaxFramesPerPacket: 60
    PayloadCapability: G.729 Annex A (12)
    MaxFramesPerPacket: 60
```

```
0000  00 0b cd 9a 80 a7 00 09  43 66 57 58 08 00 45 68   ........ CfWX..Eh
0010  00 a8 02 8c 00 00 40 06  41 e4 c0 a8 da 1d c0 a8   ......@. A.......
0020  da 09 c3 30 07 d0 34 74  0f 4c a6 ee f9 82 50 18   ...0..4t .L....P.
0030  05 78 29 27 00 00 78 00  00 00 00 00 00 00 10 00   .x)'..x. ........
0040  00 00 07 00 00 00 19 00  00 00 78 00 00 00 00 00   ........ ..x.....
0050  00 00 00 00 00 00 04 00  00 00 28 00 00 00 00 00   ........ ..(.....
0060  00 00 00 00 00 00 02 00  00 00 28 00 00 00 00 00   ........ ..(.....
0070  00 00 00 00 00 00 0f 00  00 00 3c 00 00 00 00 00   ........ ..<.....
0080  00 00 00 00 00 00 10 00  00 00 3c 00 00 00 00 00   ........ ..<.....
0090  00 00 00 00 00 00 0b 00  00 00 3c 00 00 00 00 00   ........ ..<.....
00a0  00 00 00 00 00 00 0c 00  00 00 3c 00 00 00 00 00   ........ ..<.....
00b0  00 00 00 00 00 00                                  ......
```



SCCP

SCCP (“Skinny”) Messages

(in order of appearance)

Stage I ½ – Keep Alive/Alarm Messages

	Msg	Usage	Data
→	0000	KeepAliveMessage	-- (sent periodically by phone)
←	0100	KeepAliveAckMessage	-- (sent periodically by callMgr)
→	0020	Alarm Message	Alarm Severity, Display Message & Params

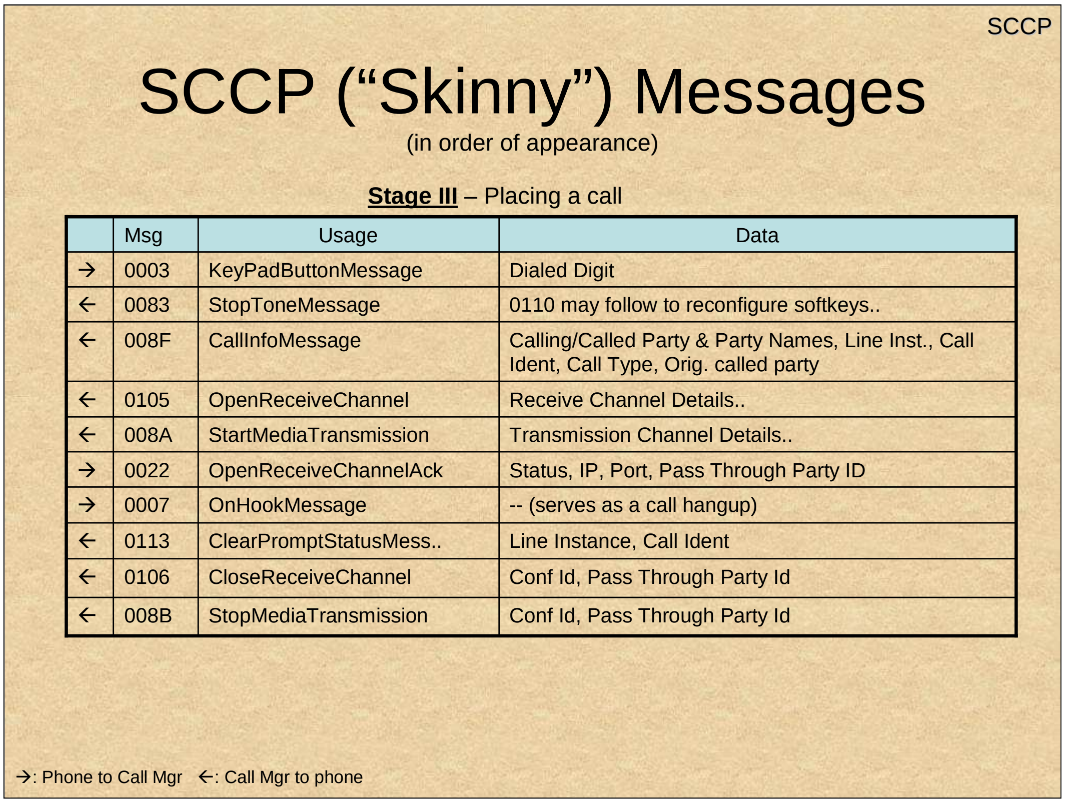
Stage II – Picking up the handset

	Msg	Usage	Data
→	0006	OffHookMessage	--
←	0099	DisplayTextMessage	ASCII text, NULL terminated
←	0086	SetLampMessage	Stimulus, StimulusInstance, LampMode
←	0111	CallStateMessage	Call State (code), Line Instance, Call Ident
←	0112	DisplayPromptStatus	Timeout, DisplayMessage*, Line Inst, Call Ident
←	0110	SelectSoftKeysMessage	Line Instance, Call Ident, SoftKeySet, SoftKeyMap (16-bit bitmap)
←	0116	ActivateCallPlaneMessage	Line Instance
←	0082	StartToneMessage	Dial Tone (as 32 bit identifier)

→: Phone to Call Mgr ←: Call Mgr to phone

The phone periodically sends “KeepAlive” messages to the CCM (as instructed by the CCM during the registration). Alarms are sent in case of errors – network errors, mostly, such as a phone’s inability to load a file from the TFTP, etc.

When a user picks up the handset, the phone sends an “OffHook” message to the CCM. The CCM, in turn, tells the phone e-x-a-c-t-l-y what to do. From the lamp on/off, through the prompt, key settings, and even the dialtone.



SCCP

SCCP ("Skinny") Messages

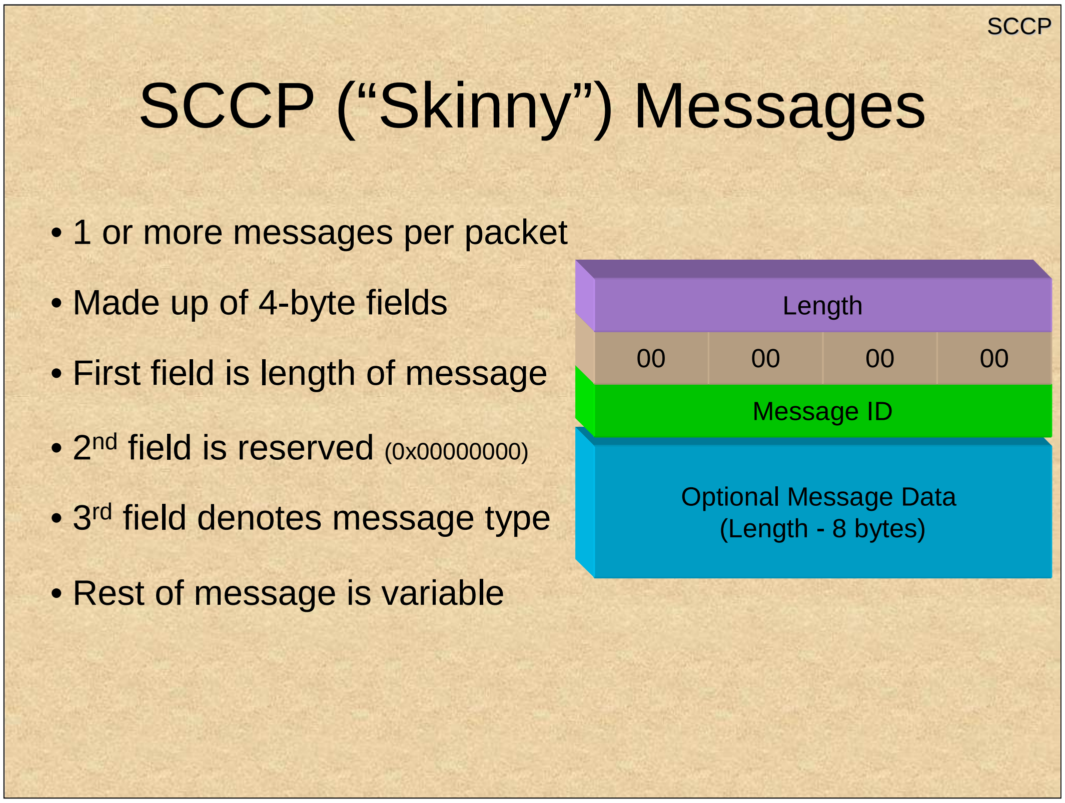
(in order of appearance)

Stage III – Placing a call

	Msg	Usage	Data
→	0003	KeyPadButtonMessage	Dialed Digit
←	0083	StopToneMessage	0110 may follow to reconfigure softkeys..
←	008F	CallInfoMessage	Calling/Called Party & Party Names, Line Inst., Call Ident, Call Type, Orig. called party
←	0105	OpenReceiveChannel	Receive Channel Details..
←	008A	StartMediaTransmission	Transmission Channel Details..
→	0022	OpenReceiveChannelAck	Status, IP, Port, Pass Through Party ID
→	0007	OnHookMessage	-- (serves as a call hangup)
←	0113	ClearPromptStatusMess..	Line Instance, Call Ident
←	0106	CloseReceiveChannel	Conf Id, Pass Through Party Id
←	008B	StopMediaTransmission	Conf Id, Pass Through Party Id

→: Phone to Call Mgr ←: Call Mgr to phone

The phone signals the end of a call by an "OnHook" message, telling the call manager the user replaced the handset (therefore hung up the call). It's then that the Call Manager tells the phone to stop transmitting, close the channels, set the call State to OnHook (= disconnected), and present the default user prompt.



SCCP

SCCP (“Skinny”) Messages

- 1 or more messages per packet
- Made up of 4-byte fields
- First field is length of message
- 2nd field is reserved (0x00000000)
- 3rd field denotes message type
- Rest of message is variable

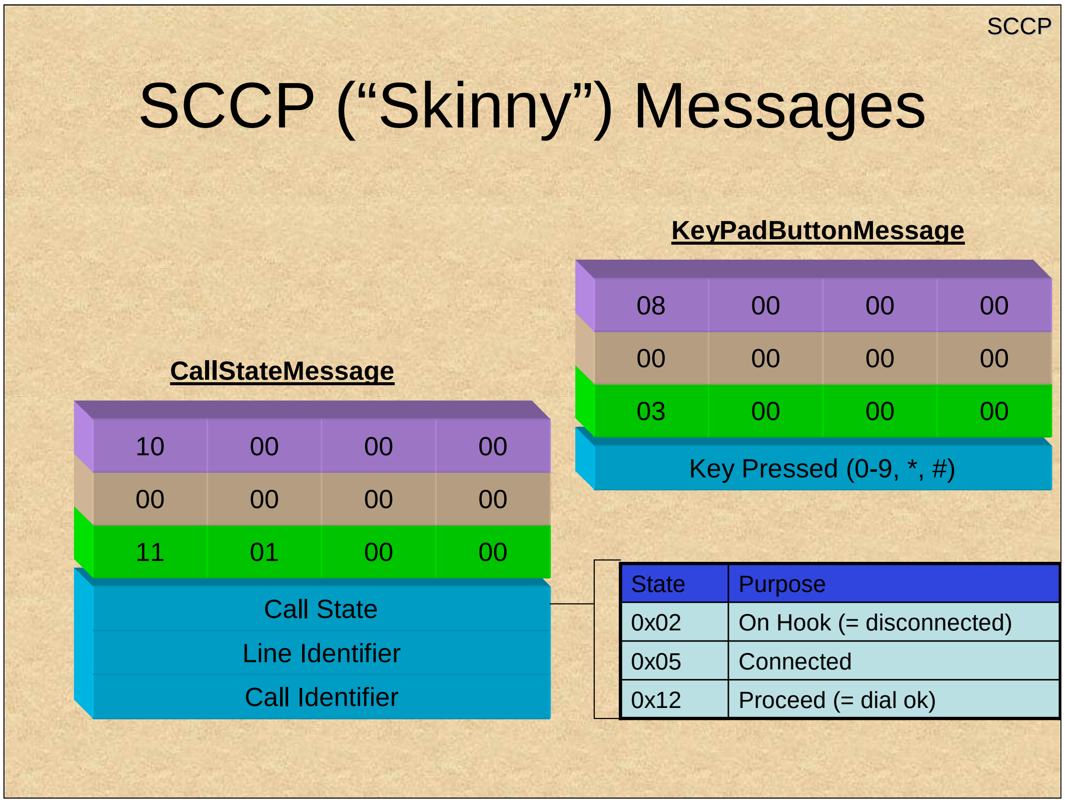
Length

00 00 00 00

Message ID

Optional Message Data
(Length - 8 bytes)

As stated, SCCP is an extremely simple (and wasteful(!)) protocol. The slide above depicts the basic format of a SCCP message. All “fields” are 4 bytes (i.e. words), for easier processing at the phone side. The first field is the length of the message (i.e. the rest of the fields, excluding the “reserved” field, next, which is always zero). Then, the message type – and, if applicable, message arguments. Most messages, however, are of fixed size, as they have a predefined number of arguments. The messages containing strings, however (usually NULL terminated), may differ.



SCCP

SCCP ("Skinny") Messages

KeyPadButtonMessage

08	00	00	00
00	00	00	00
03	00	00	00
Key Pressed (0-9, *, #)			

CallStateMessage

10	00	00	00
00	00	00	00
11	01	00	00
Call State			
Line Identifier			
Call Identifier			

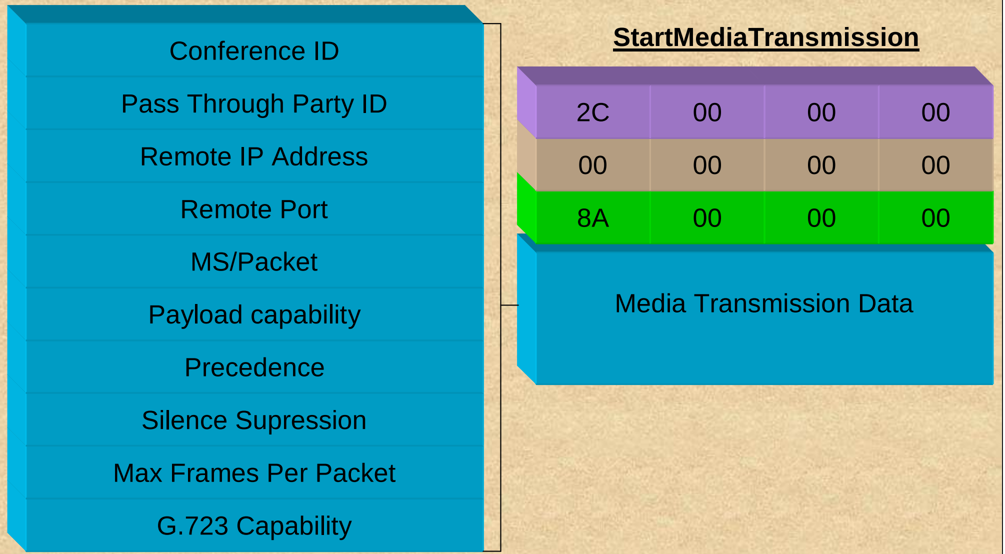
State	Purpose
0x02	On Hook (= disconnected)
0x05	Connected
0x12	Proceed (= dial ok)

The slide above shows the important "dialing" messages that SCCP supports. These are the KeyPadButton Message (for each dialed digit) and the CallState Message. The latter is sent by the Call Manager to the Station at various stages of the call lifespan, with the codes specified in the table above.

Note, again, that the protocol is VERY wasteful. Each digit is sent on its own in a KeyPadButton Message (as one byte out of the four).

SCCP

SCCP ("Skinny") Messages

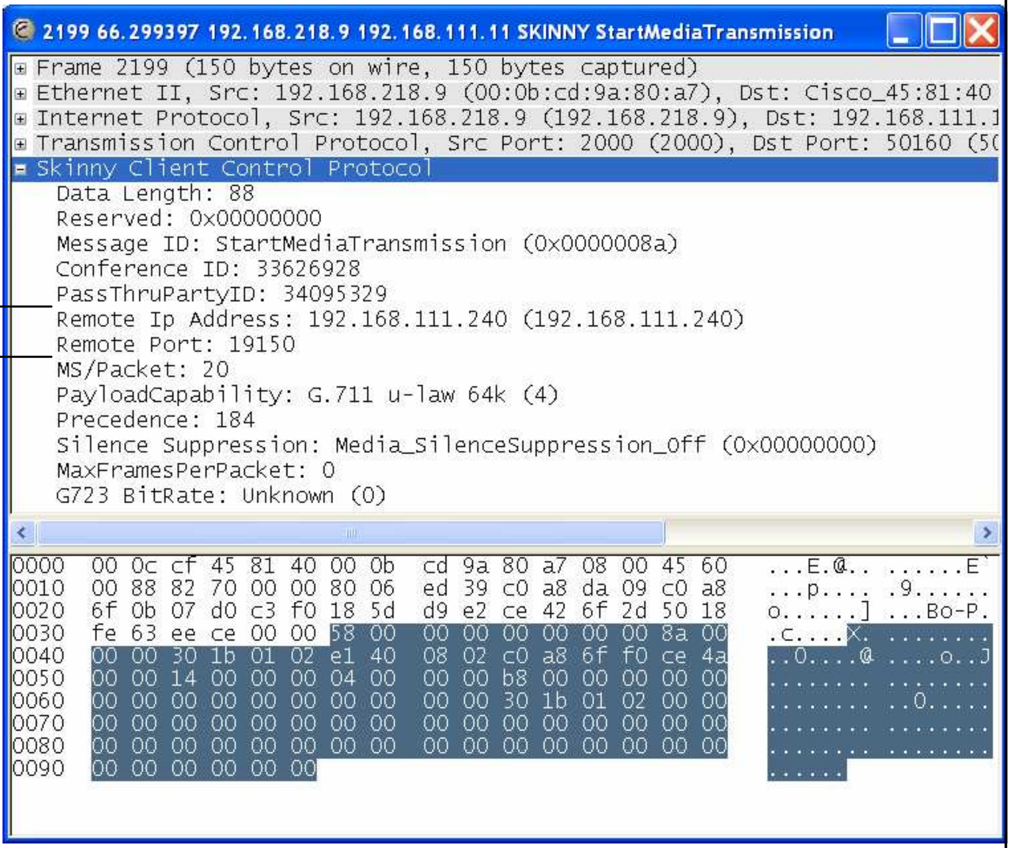


2C	00	00	00
00	00	00	00
8A	00	00	00

The "Start Media Transmission" is one of the more complex SCCP messages, due to its many fields. Its format is shown above, and in the following illustration.

The "Payload Capability" denotes the type of RTP transport (e.g. "4" for G.711, as we have seen for H.323). RTP is handled next.

CCM instructs phone to connect to this IP and port with RTP



2199 66.299397 192.168.218.9 192.168.111.11 SKINNY StartMediaTransmission

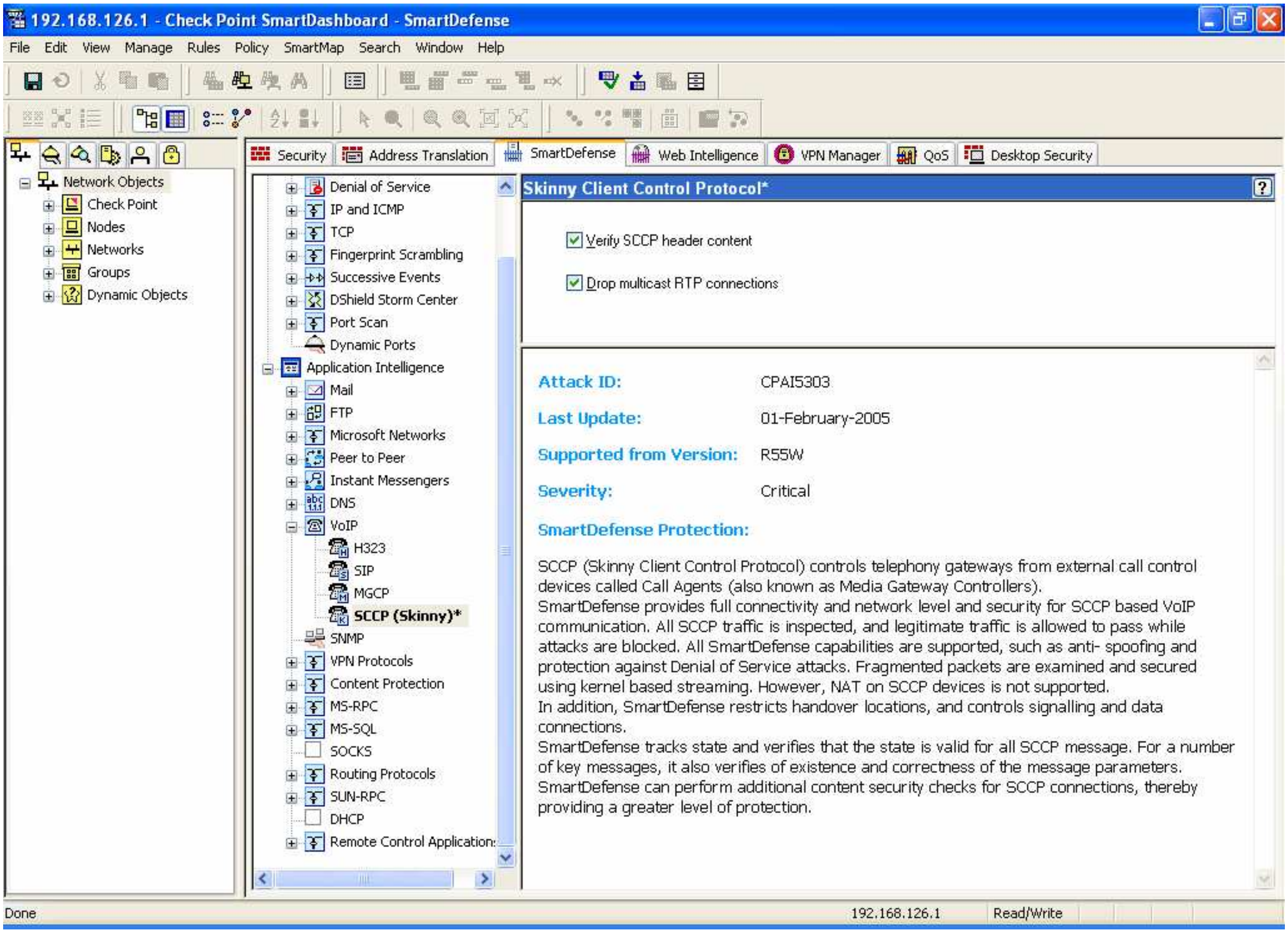
```
⊞ Frame 2199 (150 bytes on wire, 150 bytes captured)
⊞ Ethernet II, Src: 192.168.218.9 (00:0b:cd:9a:80:a7), Dst: Cisco_45:81:40
⊞ Internet Protocol, Src: 192.168.218.9 (192.168.218.9), Dst: 192.168.111.1
⊞ Transmission Control Protocol, Src Port: 2000 (2000), Dst Port: 50160 (5(
⊟ Skinny Client Control Protocol
    Data Length: 88
    Reserved: 0x00000000
    Message ID: StartMediaTransmission (0x0000008a)
    Conference ID: 33626928
    PassThruPartyID: 34095329
    Remote Ip Address: 192.168.111.240 (192.168.111.240)
    Remote Port: 19150
    MS/Packet: 20
    PayloadCapability: G.711 u-law 64k (4)
    Precedence: 184
    Silence Suppression: Media_SilenceSuppression_Off (0x00000000)
    MaxFramesPerPacket: 0
    G723 BitRate: Unknown (0)

0000  00 0c cf 45 81 40 00 0b  cd 9a 80 a7 08 00 45 60   ...E.@.. ......E`
0010  00 88 82 70 00 00 80 06  ed 39 c0 a8 da 09 c0 a8   ...p.... .9......
0020  6f 0b 07 d0 c3 f0 18 5d  d9 e2 ce 42 6f 2d 50 18   o......] ...Bo-P.
0030  fe 63 ee ce 00 00 58 00  00 00 00 00 00 00 8a 00   .c....X. ........
0040  00 00 30 1b 01 02 e1 40  08 02 c0 a8 6f f0 ce 4a   ..0....@ ....o..J
0050  00 00 14 00 00 00 04 00  00 00 b8 00 00 00 00 00   ........ ........
0060  00 00 00 00 00 00 00 00  00 00 30 1b 01 02 00 00   ........ ..0.....
0070  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........ ........
0080  00 00 00 00 00 00 00 00  00 00 00 00 00 00 00 00   ........ ........
0090  00 00 00 00 00 00                                  ......
```

SCCP

SCCP (“Skinny”) Firewall Features

- Standard VoIP defenses
- No SCCP specific options aside from basic validation
- SCCP NAT features not supported



Skinny Client Control Protocol*

☑ Verify SCCP header content

☑ Drop multicast RTP connections

Attack ID:	CPAI5303
Last Update:	01-February-2005
Supported from Version:	R55W
Severity:	Critical

SmartDefense Protection:

SCCP (Skinny Client Control Protocol) controls telephony gateways from external call control devices called Call Agents (also known as Media Gateway Controllers).
SmartDefense provides full connectivity and network level and security for SCCP based VoIP communication. All SCCP traffic is inspected, and legitimate traffic is allowed to pass while attacks are blocked. All SmartDefense capabilities are supported, such as anti- spoofing and protection against Denial of Service attacks. Fragmented packets are examined and secured using kernel based streaming. However, NAT on SCCP devices is not supported.
In addition, SmartDefense restricts handover locations, and controls signalling and data connections.
SmartDefense tracks state and verifies that the state is valid for all SCCP message. For a number of key messages, it also verifies of existence and correctness of the message parameters.
SmartDefense can perform additional content security checks for SCCP connections, thereby providing a greater level of protection.