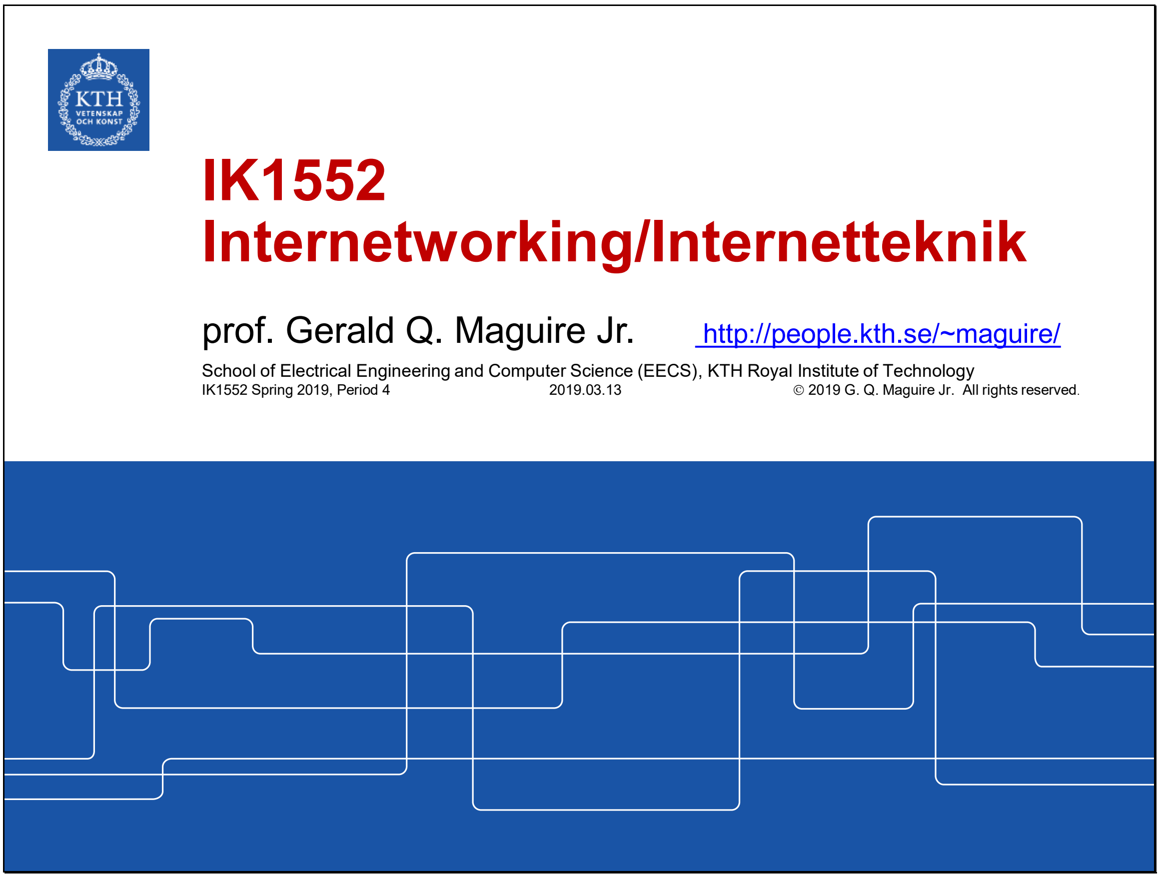
Slide 1


IK1552
Internetworking/Internetteknik
prof. Gerald Q. Maguire Jr. http://people.kth.se/~maguire/
School of Electrical Engineering and Computer Science (EECS), KTH Royal Institute of Technology
IK1552 Spring 2019, Period 4 2019.03.13 © 2019 G. Q. Maguire Jr. All rights reserved.

Slide 2

Module 6: Stream Control Transmission Protocol (STCP)

Lecture notes of G. Q. Maguire Jr.

For use in conjunction with James F. Kurose and Keith W. Ross, *Computer Networking: A Top-Down Approach*.

Slide 3

Main Transport layer protocols

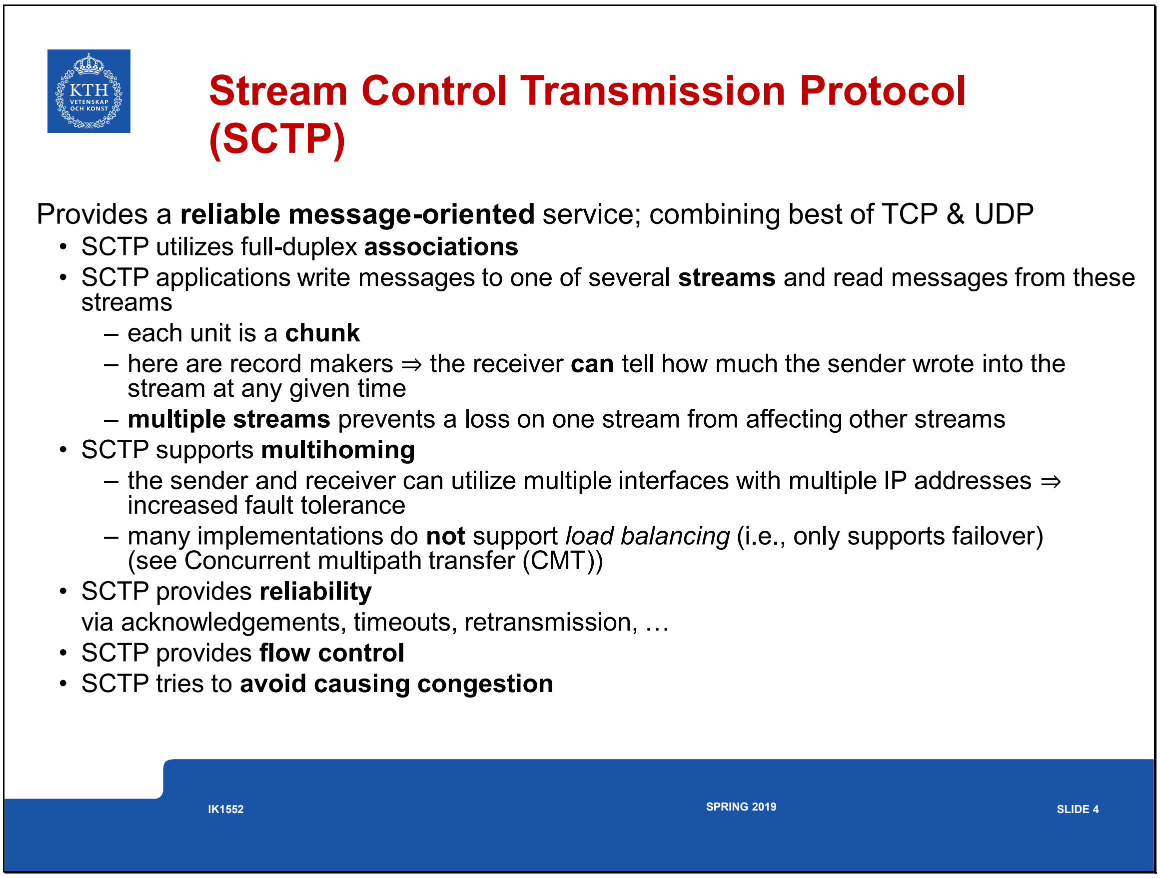
Three main transport layer protocols:

- User Datagram Protocol (UDP)
  Connectionless **unreliable** service
- Transmission Control Protocol (TCP)
  Connection-oriented **reliable** stream service
- Stream Control Transmission Protocol (STCP) <<< today's topic
  a modern transmission protocol with many facilities which the user can chose from

IK1552 SPRING 2019 SLIDE 3

Andreas Jungmaier, "A Gentle Introduction to SCTP", 19th Chaos Communications Congress, Berlin, 2002 *http://tdrwww.exp-math.uni-essen.de/inhalt/forschung/19ccc2002/html/slide_1.html*

Slide 4



R. Stewart, Q. Xie, K. Morneault, C. Sharp, H. Schwarzbauer, T. Taylor, I. Rytina, M. Kalla, L. Zhang, and V. Paxson, “Stream Control Transmission Protocol”, IETF RFC 2960, October 2000 *http://www.ietf.org/rfc/rfc2960.txt*

Randall R. Stewart and Qiaobing Xie, “Stream Control Transmission Protocol: A Reference Guide”, Addison-Wesley, 2002, ISBN 0-201-72186-4.

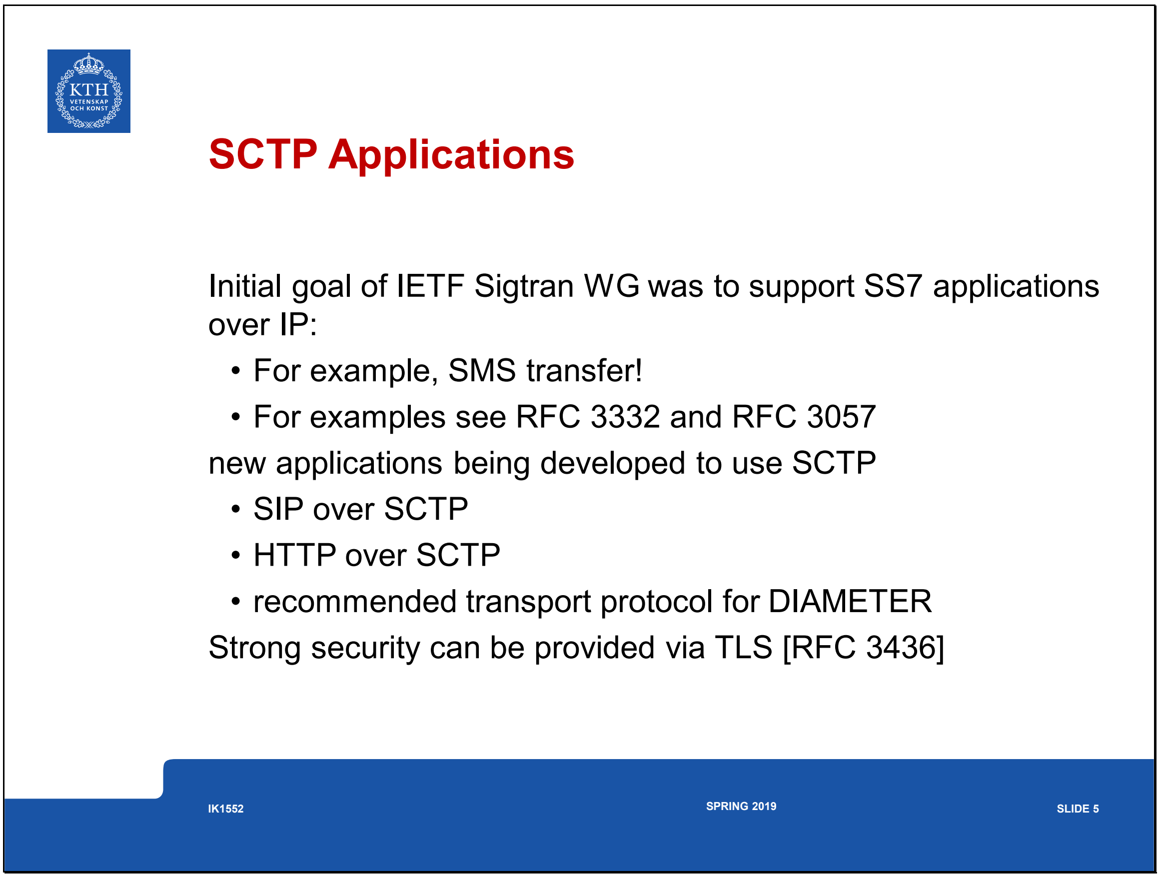
R. Stewart, ‘Stream Control Transmission Protocol’, *Internet Request for Comments*, vol. RFC 4960 (Proposed Standard), Sep. 2007 [Online]. Available: http://www.rfc-editor.org/rfc/rfc4960.txt

Guo Wei and Cheng Shiduan , Load Sharing in Stream Control Transmission Protocol, ITC19/ Performance Challenges for Efficient Next Generation Networks LIANG X.J. and XIN Z.H.(Editors), V.B. IVERSEN and KUO G.S.(Editors), Beijing University of Posts and Telecommunications Press, pp 1797-1805.
http://www.i-teletraffic.org/fileadmin/ITCBibDatabase/2005/guo05.pdf

J. R. Iyengar, P. D. Amer, and R. Stewart, ‘Concurrent Multipath Transfer Using SCTP Multihoming Over Independent End-to-End Paths’, *IEEE/ACM Transactions on Networking*, vol. 14, no. 5, pp. 951–964, Oct. 2006. DOI: 10.1109/TNET.2006.882843
http://www.eecis.udel.edu/~amer/PEL/poc/pdf/Journal-iyengar-CMToverIndependent%20Paths.pdf

T. Dreibholz, M. Becke, and H. Adhari, SCTP Socket API Extensions for Concurrent Multipath Transfer,
Internet-Draft, January 12, 2015, Expires: July 16, 2015
https://tools.ietf.org/html/draft-dreibholz-tsvwg-sctpsocket-multipath-09

Slide 5



G. Sidebottom, K. Morneault, and J. Pastor-Balbas, 'Signaling System 7 (SS7) Message Transfer Part 3 (MTP3) - User Adaptation Layer (M3UA)', *Internet Request for Comments*, vol. RFC 3332 (Proposed Standard), Sep. 2002 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3332.txt

K. Morneault and J. Pastor-Balbas, 'Signaling System 7 (SS7) Message Transfer Part 3 (MTP3) - User Adaptation Layer (M3UA)', *Internet Request for Comments*, vol. RFC 4666 (Proposed Standard), Sep. 2006 [Online]. Available: http://www.rfc-editor.org/rfc/rfc4666.txt

K. Morneault, S. Rengasami, M. Kalla, and G. Sidebottom, 'ISDN Q.921-User Adaptation Layer', *Internet Request for Comments*, vol. RFC 3057 (Proposed Standard), Feb. 2001 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3057.txt

A. Jungmaier, E. Rescorla, and M. Tuexen, 'Transport Layer Security over Stream Control Transmission Protocol', *Internet Request for Comments*, vol. RFC 3436 (Proposed Standard), Dec. 2002 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3436.txt

Slide 6

SCTP Header

0 – 15	16 – 31	
16 bit source port	16 bit source port	12 bytes
32 bit verification tag		
32 bit Checksum		
Control chunk(s) {If any}		
Data chunk(s) {If any}		

SCTP packet (see Forouzan figure 13.4 pg. 350)

IP protocol x84 = SCTP

General Header

- As with UDP & TCP, SCTP provides de/multiplexing via the 16 bit source and destination ports.
- Associations between end points are defined by a unique **verification tag**
  A separate verification tag is used in each direction
- SCTP applies a CRC-32 end-to-end checksum to its general header and all the chunks
  (Previously it used Adler-32 checksum [RFC 3309])

Chunks

- Control information is contained in Control Chunks (these always **precede** any data chunks)
- Multiple data chunks can be present - each containing data for different streams

J. Stone, R. Stewart, and D. Otis, ‘Stream Control Transmission Protocol (SCTP) Checksum Change’, *Internet Request for Comments*, vol. RFC 3309 (Proposed Standard), Sep. 2002 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3309.txt

Slide 7

SCTP Chunk

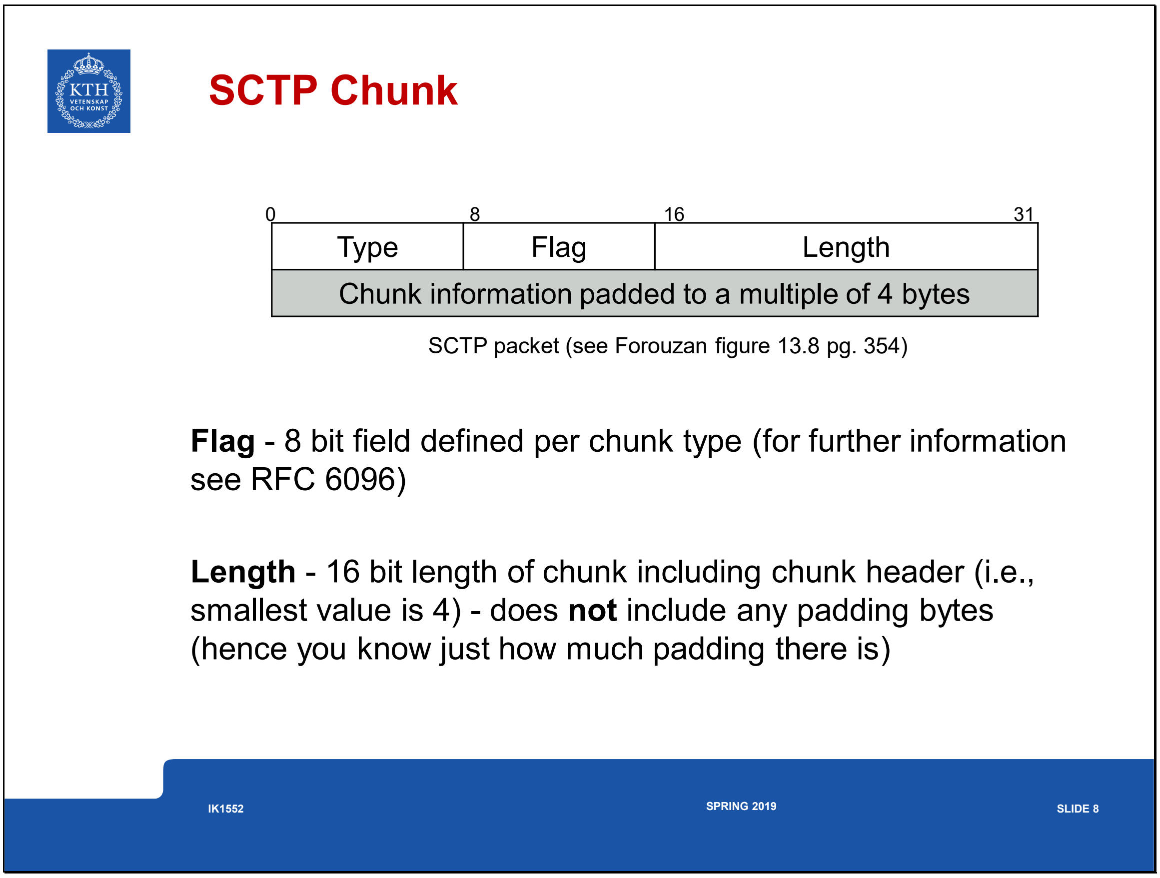
0	8	16	31
Type	Flag	Length	
Chunk information padded to a multiple of 4 bytes			

SCTP packet (see Forouzan figure 13.8 pg. 354)

Type

Type	Chunk	Description	Type	Chunk	Description
0	DATA	User data	7	SHUTDOWN	Terminate an association
1	INIT	Setup an association	8	SHUTDOWN-ACK	Acknowledge SHUTDOWN chunk
2	INIT-ACK	Acknowledge an INIT chunk	9	ERROR	Reports errors without shutting down
3	SACK	Selective Acknowledgement	10	COOKIE ECHO	Third packet in establishment of an association
4	HEARTBEAT	Probe to see if peer is alive	11	COOKIE ACK	Acknowledges COOKIE ECHO chunk
5	HEARTBEAT-ACK	Acknowledgement of a HEARTBEAT chunk	14	SHUTDOWN COMPLETE	Third packet in an association terminations
6	ABORT	Abort an association	192	FORWARD TSN	To adjust the cumulative TSN

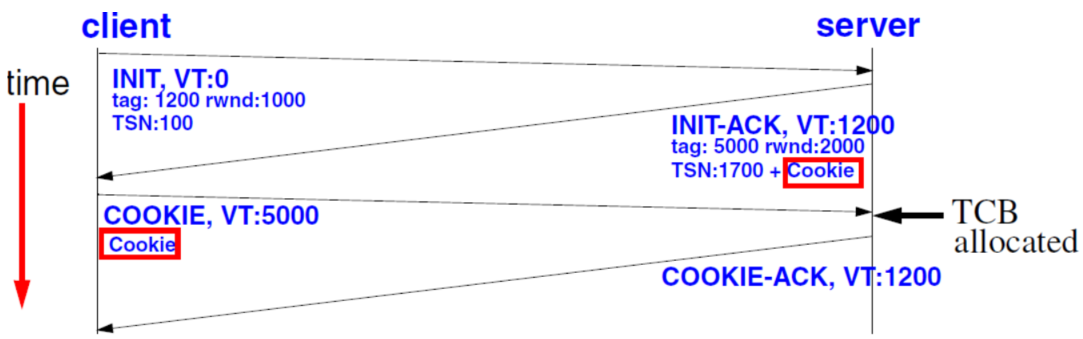
Slide 8



M. Tuexen and R. Stewart, ‘Stream Control Transmission Protocol (SCTP) Chunk Flags Registration’, *Internet Request for Comments*, vol. RFC 6096 (Proposed Standard), Jan. 2011 [Online]. Available: http://www.rfc-editor.org/rfc/rfc6096.txt

Slide 9

Association establishment - 4-way handshake



See Forouzan figure 13.19 page 363

The entity initiating the connection is (normally) called the "client" and does an **active** open, where as the server needed to previously do a **passive** open.

Slide 10

INIT Chunk

0	8	16	31
Type=1	Flag=0	Length	
Initiation tag			
Advertised receiver window credit (rwnd)			
Outbound streams		Maximum inbound streams	
Initial Transmission sequence number (TSN)			
variable-length parameters (optional)			

SCTP INIT chunk (see Forouzan figure 13.10 pg. 357)

Initiation tag

- defines the tags for this association to be used by the other party
- reduce the risk due to a blind attacker (since there is only a 1 in 2^{32} chance of guessing the right tag)
- can reject delayed packets - thus avoiding the need for TCP's TIME-WAIT timer

Advertised receiver window credit - defines rwnd (i.e., how much the receiver can send to this party)

Outbound streams - suggested upper number of streams **from** this sender (can be reduced by receiver)

Maximum inbound streams - upper limit of streams **to** this sender

Slide 11

INIT Chunk (continued)

Transmission sequence number (TSN)

Initializes the TSN in the outbound direction, initialized to a random value

Variable-length parameters

- IP address(es) of endpoint (see also RFC 5061)
  - Multiple addresses are used to support multihoming
  - The **receiver** selects the primary address for the other endpoint
- Type of addresses
- Support
- …

R. Stewart, Q. Xie, M. Tuexen, S. Maruyama, and M. Kozuka, ‘Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration’, *Internet Request for Comments*, vol. RFC 5061 (Proposed Standard), Sep. 2007 [Online]. Available: http://www.rfc-editor.org/rfc/rfc5061.txt

Slide 12

INIT ACK Chunk

0	8	16	31
Type=2	Flag=0	Length	
Initiation tag			
Advertised receiver window credit (rwnd)			
Outbound streams		Maximum inbound streams	
Initial Transmission sequence number (TSN)			
Parameter type: 7		Parameter length	
State Cookie			
variable-length parameters (optional)			

SCTP INIT ACK chunk (see Forouzan figure 13.11 pg. 358)

The same fields as in the INIT chunk (with **Initiation tag** value set to that of the INIT) - but with the addition of a **required parameter** with a state cookie.

- **Parameter type: 7 = State Cookie**
- **Parameter length = size of State Cookie + 4 (the parameter type and length fields)**

A packet carrying this INIT ACK chunk can not contain any other control or data chunks.

Slide 13

State Cookie

Use of the COOKIE prevents a SYN flood like attack - since resources are not allocated until the COOKIE ECHO chuck is received.

However, state has to be saved from the initial INIT chunk - therefore it is placed in the cookie in a way that only the server can access it (hence the cookie is sealed with an HMAC {aka digest} after being created {aka "baked"}). This requires that the server has a secret key which it uses to compute this digest.

If the sender of the INIT is an attacker located on another machine, they would not be able to receive the cookie if they faked the source address in the INIT - since the INIT ACK is sent to the address and contains the cookie!

- Without a cookie ⇒ no association is created and no resources (such as TCB) are tied up!

IK1552 SPRING 2019 SLIDE 13

Slide 14

COOKIE ECHO Chunk

0	8	16	31
Type=10	Flag=0	Length	
State Cookie			

SCTP COOKIE ECHO chunk (see Forouzan figure 13.12 pg. 359)

(chunk) **Type: 10** = COOKIE ECHO

(chunk) **length** = size of State Cookie + 4 (the parameter type and length fields)

State Cookie

- simply a copy of the COOKIE data from the INIT ACK chunk
- The COOKIE data is opaque (i.e., only the sender can read the cookie)

A packet carrying this COOKIE ECHO chunk can contain other control or data chunks -- in particular it can carry the first user (client) data!

Slide 15

COOKIE ACK Chunk

0	8	16	31
Type=11	Flag=0	Length=4	

SCTP COOKIE ACK chunk (see Forouzan figure 13.13 pg. 359)

Completes the 4 way handshake.

A packet with this chunk can also carry control and data chunks (in particular the first of the user (server) data.

Slide 16

Data Chunk

0	8				16 31
Type=0	Reserved	U	B	E	Length
Transmission sequence number (TSN)					
Stream identifier (SI)					Stream Sequence number (SSN)
Protocol Identifier					
User data					

SCTP Data Chunk (see Forouzan figure 13.9 pg. 356)

Flags:

U - Unordered - for delivery to the application right away

B - Beginning (chunk position - for use with fragmentation)

E - End chunk

Transmission sequence number (TSN) - only data chunks consume TSNs

Stream identifier (SI)

Stream Sequence number (SSN)

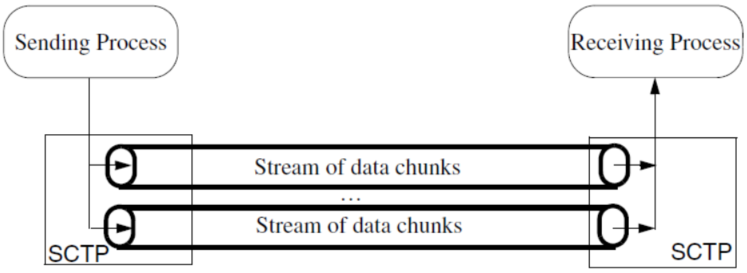
Protocol Identifier

User data

- at least 1 byte of user data; padded to 32 bit boundaries
- although a message can be spread over multiple chunks, each chunk contains data from **only** a single message (like UDP, each message results in one or more data SCTP chunks)

Slide 17

Multiple-Streams



Multiple-streams (see Forouzan figure 13.2 pg. 347)

The figure above shows a **single** association.

Each stream has a unique **stream identifier** (SI) and maintains its **own stream sequence number** (SSN).

Unordered data chunks (i.e., with U = 0) - do **not** consume a SSN and are delivered when they arrive at the destination.

Multiple streams and unordered data avoid TCP's **head of line blocking**.

IK1552 SPRING 2019 SLIDE 17

Slide 18

Selective Acknowledgement (SACK) Chunk

0	8	16	31
Type=3	Flag=0	Length	
cumulative TSN acknowledgement			
Advertised receiver window credit (rwnd)			
Number of gap ACK blocks: N		Number of duplicates: M	
Gap ACK block #1 start TSN offset		Gap ACK block #1 end TSN offset	
…		…	
Gap ACK block #N start TSN offset		Gap ACK block #N end TSN offset	
Duplicate TSN 1			
…			
Duplicate TSN M			

SCTP Data Chunk (see Forouzan figure 13.9 pg. 356)

Cumulative Transmission sequence number (TSN) acknowledgement - the last data chunk received in sequence

Gap = received sequence of chunks (indicated with start .. end TSNs)

Duplicate TSN - indicating duplicate chunks (if any)

SACK always sent to the IP address where the corresponding packet originated

Slide 19

ERROR chunk

0	8	16	31
Type=9	Flag	Length	
One or more causes (optional)			

SCTP ERROR chunk (see Forouzan figure 13.17 pg. 361)

Error code	Description
1	Invalid Stream identifier
2	Missing mandatory parameter
3	State cookie error
4	Out of resource
5	Unresolvable address
6	Unrecognized chunk type
7	Invalid mandatory parameters
8	Unrecognized parameter
9	No user data
10	Cookie received while shutting down

Slide 20

Association Termination

Two forms of termination:

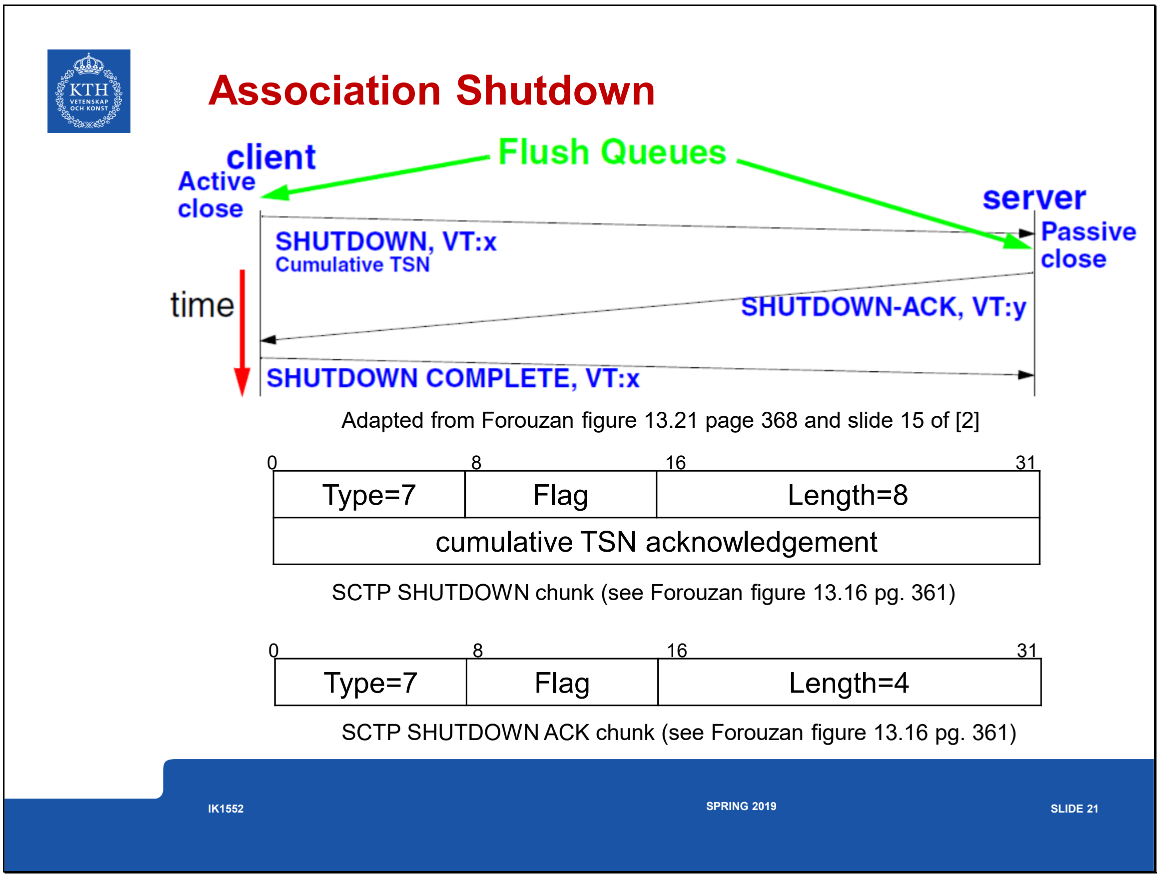
- **Association Abort**
  - Used in the event of a fatal error
  - uses same error codes as the ERROR Chunk
  - Chunk format

0	8	16	31
Type=6	Flag=6	Length	
One or more causes (optional)			

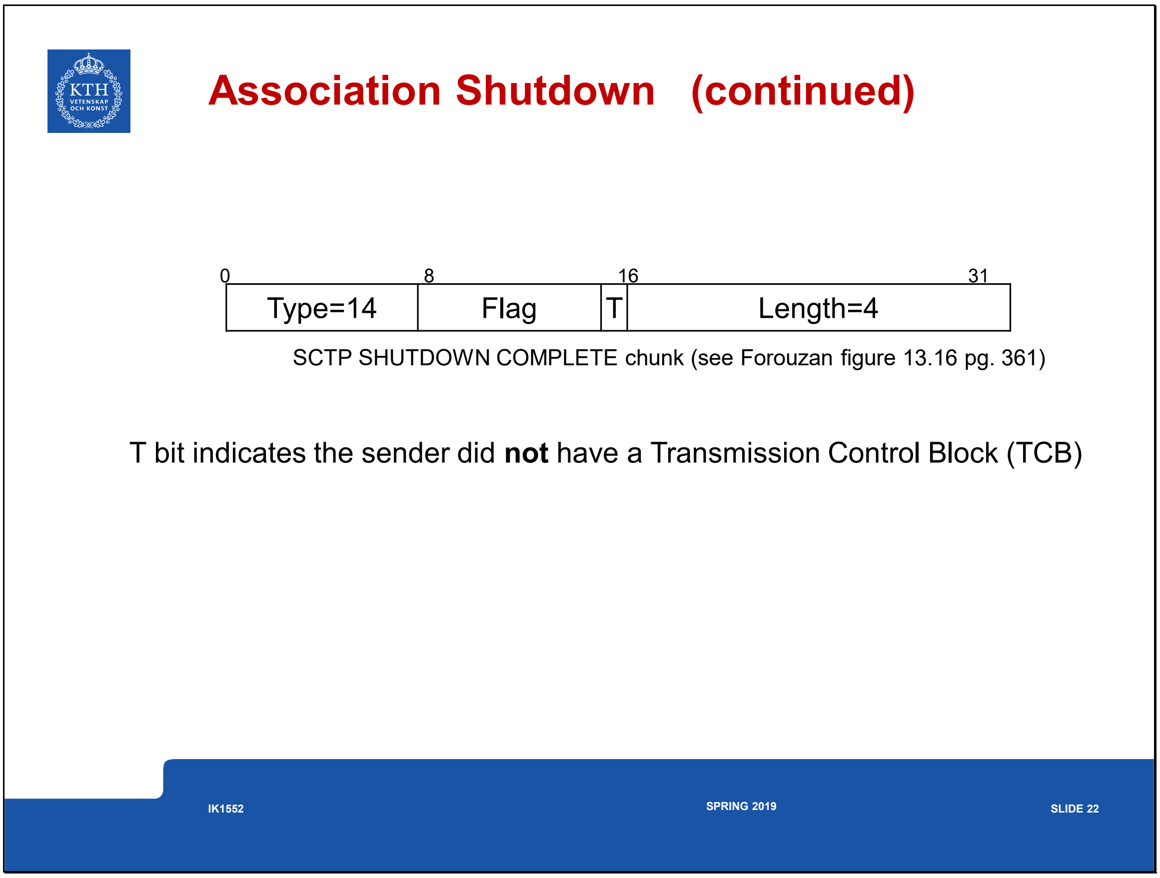
SCTP ABORT chunk (see Forouzan figure 13.18 pg. 362)

- **Association Shutdown** - graceful termination

Slide 21


Association Shutdown
client
Active close
Flush Queues
server
Passive close
SHUTDOWN, VT:x
Cumulative TSN
time
SHUTDOWN-ACK, VT:y
SHUTDOWN COMPLETE, VT:x
Adapted from Forouzan figure 13.21 page 368 and slide 15 of [2]
0 8 16 31
Type=7 Flag Length=8
cumulative TSN acknowledgement
SCTP SHUTDOWN chunk (see Forouzan figure 13.16 pg. 361)
0 8 16 31
Type=7 Flag Length=4
SCTP SHUTDOWN ACK chunk (see Forouzan figure 13.16 pg. 361)
IK1552
SPRING 2019
SLIDE 21

Slide 22


Association Shutdown (continued)
0
8
16
31
Type=14
Flag
T
Length=4
SCTP SHUTDOWN COMPLETE chunk (see Forouzan figure 13.16 pg. 361)
T bit indicates the sender did **not** have a Transmission Control Block (TCB)
IK1552
SPRING 2019
SLIDE 22

Slide 23

SCTP Example - Daytime [Jungmaier 2004]

server# ./daytime_server -s 192.168.1.2 -vv
1 : Communication up (1 paths)
1 : Network status change: path 0 is now REACHABLE
1 : Shutdown complete

client# ./terminal -vv -r 13 -d 192.168.1.2 -s 192.168.1.1
1 : Communication up (1 paths, 1 In-Streams, 1 Out-Streams)
1 : Network status change: path 0 (towards 192.168.1.2) is now REACHABLE
Wed Apr 27 11:52:04 2005
1 : Shutdown received

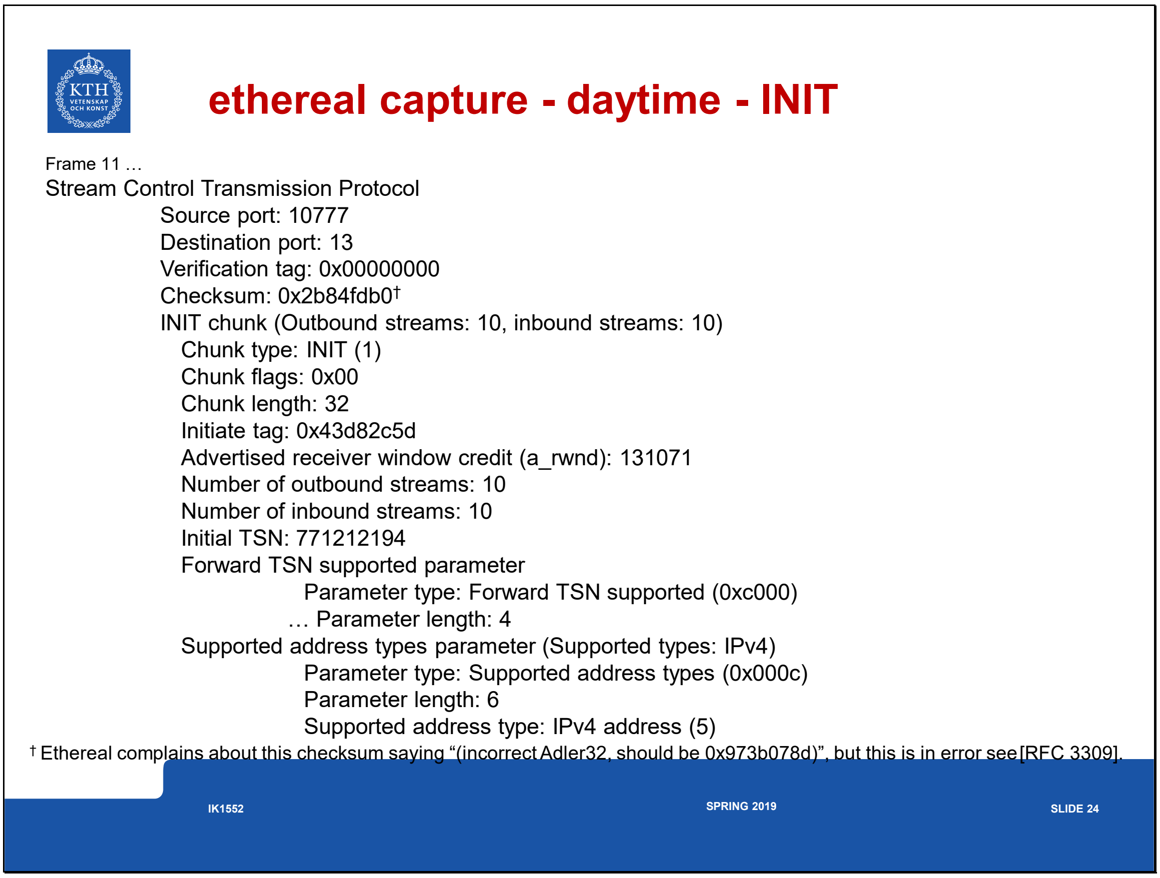
11	74.864232	192.168.1.1	192.168.1.2	SCTP	INIT
12	74.864552	192.168.1.2	192.168.1.1	SCTP	INIT_ACK
13	74.864808	192.168.1.1	192.168.1.2	SCTP	COOKIE_ECHO
14	74.865073	192.168.1.2	192.168.1.1	SCTP	COOKIE_ACK
15	74.865273	192.168.1.2	192.168.1.1	SCTP	DATA
16	74.865733	192.168.1.1	192.168.1.2	SCTP	SACK
17	74.865933	192.168.1.2	192.168.1.1	SCTP	SHUTDOWN
18	74.866132	192.168.1.1	192.168.1.2	SCTP	SHUTDOWN_ACK
19	74.866195	192.168.1.2	192.168.1.1	SCTP	SHUTDOWN_COMPLETE

SCTP Daytime example - output from Ethereal

IK1552 SPRING 2019 SLIDE 23

Andreas Jungmaier , Herbert Hölzlwimmer, Michael Tüxen , and Thomas Dreibholz, "sctplib-1.0.2", Siemens AG and the Institute of Computer Networking Technology, University of Essen, Germany, August 2004 http://www.sctp.de/sctp-download.html {Note that a later version 1.0.3 was released March 4th, 2005}

Slide 24



J. Stone, R. Stewart, and D. Otis, ‘Stream Control Transmission Protocol (SCTP) Checksum Change’, *Internet Request for Comments*, vol. RFC 3309 (Proposed Standard), Sep. 2002 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3309.txt

Slide 25

ethereal capture - daytime - INIT-ACK

Frame 12 … Stream Control Transmission Protocol
Source port: 13
Destination port: 10777
Verification tag: 0x43d82c5d
Checksum: 0x7f61f237
INIT_ACK chunk (Outbound streams: 1, inbound streams: 1)
    Chunk type: INIT_ACK (2)
    Chunk flags: 0x00
    Chunk length: 128
    Initiate tag: 0x5d581d9a
    Advertised receiver window credit (a_rwnd): 131071
    Number of outbound streams: 1
    Number of inbound streams: 1
    Initial TSN: 1514529259
    State cookie parameter (Cookie length: 100 bytes)
        Parameter type: State cookie (0x0007)
        Parameter length: 104
    State cookie: 5D581D9A0001FFFF000100015A45E1EB...
    Forward TSN supported parameter
        Parameter type: Forward TSN supported (0xc000)
        1... .... = Bit: Skip parameter and continue prosessing of the chunk
        .1.. .... = Bit: Do report
        Parameter length: 4

IK1552 SPRING 2019 SLIDE 25

J. Stone, R. Stewart, and D. Otis, 'Stream Control Transmission Protocol (SCTP) Checksum Change', *Internet Request for Comments*, vol. RFC 3309 (Proposed Standard), Sep. 2002 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3309.txt

Slide 26

ethereal capture - daytime - COOKIE-ECHO

Frame 13 …
Source port: 10777
Destination port: 13
Verification tag: 0x5d581d9a
Checksum: 0x3af3f579
COOKIE_ECHO chunk (Cookie length: 100 bytes)
    Chunk type: COOKIE_ECHO (10)
    0... .... = Bit: Stop processing of the packet
    .0.. .... = Bit: Do not report
    Chunk flags: 0x00
    Chunk length: 104
    Cookie: 5D581D9A0001FFFF000100015A45E1EB...

J. Stone, R. Stewart, and D. Otis, 'Stream Control Transmission Protocol (SCTP) Checksum Change', *Internet Request for Comments*, vol. RFC 3309 (Proposed Standard), Sep. 2002 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3309.txt

Slide 27

ethereal capture - daytime - COOKIE-ACK

Frame 14 …
Source port: 13
Destination port: 10777
Verification tag: 0x43d82c5d
Checksum: 0x762d80d7
COOKIE_ACK chunk
- Chunk type: COOKIE_ACK (11)
- Chunk flags: 0x00
- Chunk length: 4

J. Stone, R. Stewart, and D. Otis, 'Stream Control Transmission Protocol (SCTP) Checksum Change', *Internet Request for Comments*, vol. RFC 3309 (Proposed Standard), Sep. 2002 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3309.txt

Slide 28

ethereal capture - daytime - DATA

Frame 15 …
Source port: 13
Destination port: 10777
Verification tag: 0x43d82c5d
Checksum: 0xf8fb1754
DATA chunk(ordered, complete segment, TSN: 1514529259, SID: 0, SSN: 0, PPID: 0, payload length: 25 bytes)
    Chunk type: DATA (0)
    Chunk flags: 0x03
    ....1 = E-Bit: Last segment
    .... ..1. = B-Bit: First segment
    .... .0.. = U-Bit: Ordered delivery
    Chunk length: 41
    TSN: 1514529259
    Stream Identifier: 0x0000
    Stream sequence number: 0
    Payload protocol identifier: not specified (0)
    Chunk padding: 000000
Data (25 bytes)

```
0000 57 65 64 20 41 70 72 20 32 37 20 31 31 3a 34 33    Wed Apr 27 11:43
0010 3a 32 32 20 32 30 30 35 0a                         :22 2005.
```

IK1552 SPRING 2019 SLIDE 28

J. Stone, R. Stewart, and D. Otis, 'Stream Control Transmission Protocol (SCTP) Checksum Change', *Internet Request for Comments*, vol. RFC 3309 (Proposed Standard), Sep. 2002 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3309.txt

Slide 29

ethereal capture - daytime - SACK

Frame 16 …
Source port: 10777
Destination port: 13
Verification tag: 0x5d581d9a
Checksum: 0xfa994e35
SACK chunk (Cumulative TSN: 1514529259, a_rwnd: 131071, gaps: 0, duplicate TSNs: 0)
- Chunk type: SACK (3)
- Chunk flags: 0x00
- Chunk length: 16
- Cumulative TSN ACK: 1514529259
- Advertised receiver window credit (a_rwnd): 131071
- Number of gap acknowldgement blocks : 0
- Number of duplicated TSNs: 0

J. Stone, R. Stewart, and D. Otis, 'Stream Control Transmission Protocol (SCTP) Checksum Change', *Internet Request for Comments*, vol. RFC 3309 (Proposed Standard), Sep. 2002 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3309.txt

Slide 30

ethereal capture - daytime - SHUTDOWN

Frame 17 …
Source port: 13
Destination port: 10777
Verification tag: 0x43d82c5d
Checksum: 0xf447d00f
SHUTDOWN chunk (Cumulative TSN ack: 771212193)
- Chunk type: SHUTDOWN (7)
- Chunk flags: 0x00
- Chunk length: 8
- Cumulative TSN Ack: 771212193

IK1552 SPRING 2019 SLIDE 30

J. Stone, R. Stewart, and D. Otis, 'Stream Control Transmission Protocol (SCTP) Checksum Change', *Internet Request for Comments*, vol. RFC 3309 (Proposed Standard), Sep. 2002 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3309.txt

Slide 31

ethereal capture - daytime - SHUTDOWN_ACK

Frame 18

Source port: 10777

Destination port: 13

Verification tag: 0x5d581d9a

Checksum: 0x9f44d056

SHUTDOWN_ACK chunk

- Chunk type: SHUTDOWN_ACK (8)
- Chunk flags: 0x00
- Chunk length: 4

J. Stone, R. Stewart, and D. Otis, 'Stream Control Transmission Protocol (SCTP) Checksum Change', *Internet Request for Comments*, vol. RFC 3309 (Proposed Standard), Sep. 2002 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3309.txt

Slide 32

ethereal capture - daytime - SHUTDOWN_COMPLETE

Frame 19…

Source port: 13

Destination port: 10777

Verification tag: 0x43d82c5d

Checksum: 0x3db6e771

SHUTDOWN_COMPLETE chunk

    Chunk type: SHUTDOWN_COMPLETE (14)

    Chunk flags: 0x00

    ....0 = T-Bit: TCB destroyed

    Chunk length: 4

J. Stone, R. Stewart, and D. Otis, 'Stream Control Transmission Protocol (SCTP) Checksum Change', *Internet Request for Comments*, vol. RFC 3309 (Proposed Standard), Sep. 2002 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3309.txt

Slide 33

Fault Management

Endpoint Failure Detection

- Endpoint keeps a counter of the total number of consecutive retransmissions to its peer (including retransmissions to all the destination transport addresses [= port + IP address] of the peer if it is multi-homed). When this counter exceeds 'Association.Max.Retrans', the endpoint will consider the peer endpoint unreachable and shall stop transmitting any more data to it (the association enters the CLOSED state).
- Counter is reset each time:
  - a DATA chunk sent to that peer is acknowledged (by the reception of a SACK) or
  - a HEARTBEAT-ACK is received from the peer

Path Failure Detection

- Each time (1) T3-rtx timer expires on any address or (2) a HEARTBEAT sent to an idle address is **not** acknowledged within a RTO, then the error counter of that destination will be incremented. When this error counter exceeds 'Path.Max.Retrans' for that destination address, then the endpoint marks the destination transport address as inactive and notifies the upper layer.
- the endpoint clears the error counter of this destination transport address when:
  - an outstanding TSN is acknowledged or
  - a HEARTBEAT address is acknowledged
- When the primary path is marked inactive, then the sender **may** automatically transmit new packets to an alternate destination address if one exists and is active
  - If more than one alternate address is active ⇒ only **one** transport address is chosen as the new destination transport address.

Slide 34

HEARTBEAT and HEARTBEAT ACK Chunks

0	8	16	31
Type=4 or 5	Flag=0	Length	
Parameter type: 1		Parameter length	
Sender specific information			

SCTP HEARTBEAT and HEARTBEAK ACK chunks (see Forouzan figure 13.15 pg. 360)

(chunk) **Type: 4** = HEARTBEAT
(chunk) **Type: 5** = HEARTBEAT ACK
(chunk) **length** = size of sender specific information + 4 (the parameter type and length fields)

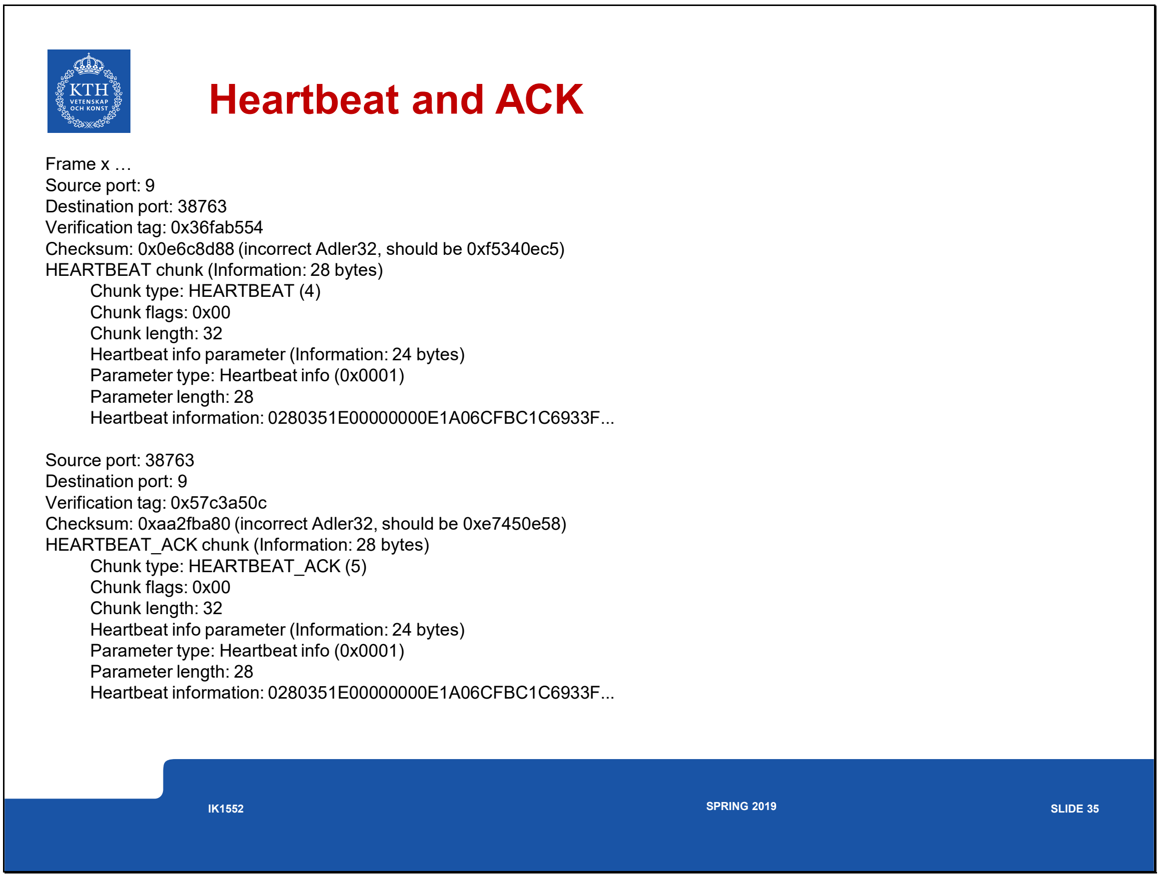
Sender specific information

- The sender puts its Local time and transport address in (note that the sctplib implementation 1.0.2 puts the time in as an unsigned 32 bit integer and puts the path index in (also as an unsigned 32 bit integer) and add a HMAC computed over these values [Jungmaier 2004]
- The acknowledgement simply contains a copy of this information

Heartbeats every ~30 seconds.

Andreas Jungmaier , Herbert Hölzlwimmer, Michael Tüxen , and Thomas Dreibholz, "sctplib-1.0.2", Siemens AG and the Institute of Computer Networking Technology, University of Essen, Germany, August 2004 *http://www.sctp.de/sctp-download.html*
{Note that a later version 1.0.3 was released March 4th, 2005}

Slide 35



J. Stone, R. Stewart, and D. Otis, 'Stream Control Transmission Protocol (SCTP) Checksum Change', *Internet Request for Comments*, vol. RFC 3309 (Proposed Standard), Sep. 2002 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3309.txt

Slide 36

Differences from TCP Congestion Control

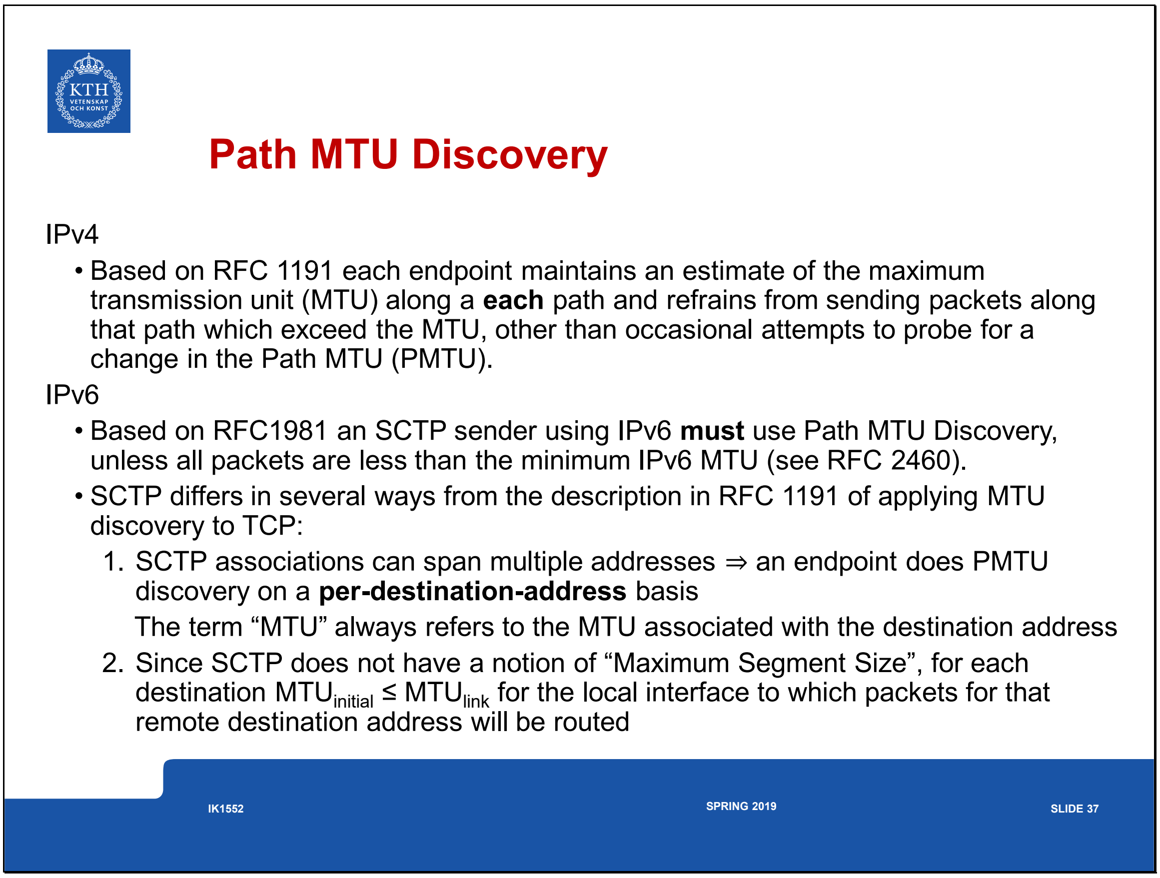
Any DATA chunk that has been acknowledged by SACK, including DATA that arrived out of order, are **only** considered fully delivered when the Cumulative TSN ACK Point passes the TSN of the DATA chunk

- ⇒ cwnd controls the amount of outstanding data, rather than (as in the case of non-SACK TCP) the upper bound between the highest acknowledged sequence number and the latest DATA chunk that can be sent within the congestion window
- ⇒ different fast-retransmit & fast-recovery than non-SACK TCP
  - Retransmission based on both retransmission timer (with an RTO per path)
  - Three SACKS (i.e., 4 consecutive duplicate SACKs indicating missing chunks) ⇒ immediate retransmission of these missing chunks

Sender

- uses the same destination address until instructed by the upper layer (however, SCTP may change to an alternate destination in the event an address is marked inactive) ⇒ retransmission can be to a different transport address than the original transmission.
- keeps separate congestion control parameters (cwnd, ssthresh, and partial_bytes_acked) for each of the destination addresses it can send to (i.e., not each source-destination pair)
  - these parameters should decay if the address is not used
  - does **slow-start** upon the **first** transmission to **each of destination addresses**

Slide 37



J. C. Mogul and S. E. Deering, ‘Path MTU discovery’, *Internet Request for Comments*, vol. RFC 1191 (Draft Standard), Nov. 1990 [Online]. Available: http://www.rfc-editor.org/rfc/rfc1191.txt
J. McCann, S. Deering, and J. Mogul, ‘Path MTU Discovery for IP version 6’, *Internet Request for Comments*, vol. RFC 1981 (Draft Standard), Aug. 1996 [Online]. Available: http://www.rfc-editor.org/rfc/rfc1981.txt
S. Deering and R. Hinden, ‘Internet Protocol, Version 6 (IPv6) Specification’, *Internet Request for Comments*, vol. RFC 2460 (Draft Standard), Dec. 1998 [Online]. Available: http://www.rfc-editor.org/rfc/rfc2460.txt

Slide 38

Path MTU Discovery IPv6 (Continued)

3. When retransmitting to a remote address for which the IP datagram appears too large for the path MTU to that address, the IP datagram **should** be retransmitted without the DF bit set, enabling it to be fragmented. While *initial* transmissions of IP datagrams **must** have DF set.
4. Sender maintains an **association PMTU** (= smallest PMTU discovered for all of the peer's destination addresses); when fragmenting messages this association PMTU is used to calculate the size of each fragment ⇒ retransmissions can sent to an alternate address without encountering IP fragmentation

Slide 39

SCTP header continued

Reliability is provided by a 32 bit SCTP **sequence numbers** (TSN)
- The initial sequence number is a random 32 bit number
- These sequence numbers are in the header of individual chunks
- This cumulative number is used to provide both flow control and error control

SCTP **resequences** data at the receiving side

SCTP **discards duplicate** data at the receiving side

The **window** size (or more exactly the receive window size (rwnd)) - indicates how many bytes the receiver is prepared to receive (this number is **relative** to the acknowledgement number).

Slide 40

Forward Cumulative TSN

Allows an endpoint to signal to its peer that it should move the cumulative acknowledgement forward [RFC 3578]. This protocol extension adds a new parameter (Forward-TSN-Supported) to INIT and INIT ACK, and a new FORWARD TSN chunk type. It provides an example of a partially reliable service.

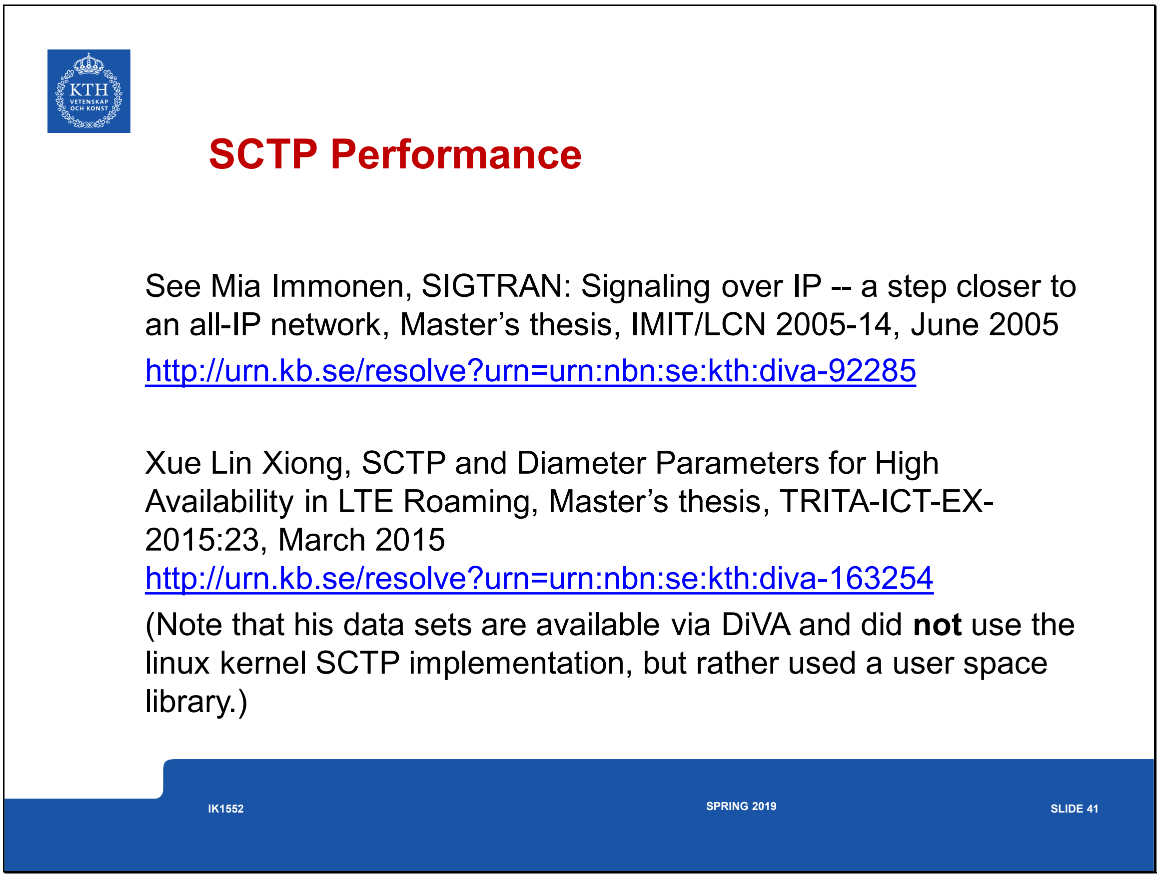
0	8	16	31
Type=192	Flag=0	Length	
New cumulative TSN			
Stream #1		Stream Sequence #1	
…		…	
Stream #N		Stream Sequence #N	

SCTP FORWARD TSN Chunk (see [RFC 3578])

Stream_i a stream number that was skipped by this FWD-TSN.
Stream Sequence_i = the largest stream sequence number in stream_i being skipped
Receiver can use the Stream_i and Stream Sequence_i fields to enable delivery of (stranded) TSN's that remain in the stream re-ordering queues.

R. Stewart, M. Ramalho, Q. Xie, M. Tuexen, and P. Conrad, 'Stream Control Transmission Protocol (SCTP) Partial Reliability Extension', *Internet Request for Comments*, vol. RFC 3758 (Proposed Standard), May 2004 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3758.txt

Slide 41



See Mia Immonen, SIGTRAN: Signaling over IP -- a step closer to an all-IP network, Master’s thesis, KTH Royal Institute of Technology, Institutionen för Mikroelektronik och Informationsteknik, IMIT/LCN 2005-14, June 2005 http://urn.kb.se/resolve?urn=urn:nbn:se:kth:diva-92285

Slide 42

Transport Protocol Functional Overview

From table 1-1 of [Stewart and Xie 2002] on page 12; also appears in ["SCTP Primer"]

Protocol Feature	SCTP	TCP	UDP
State required at each endpoint	Yes	Yes	No
Reliable data transfer	Yes	Yes	No
Congest control and avoidance	Yes	Yes	No
Message boundary conservation	Yes	No	Yes
Path MTU discovery and message fragmentation	Yes	Yes	No
Message bundling	Yes	Yes	No
Multi-homed hosts support	Yes	No	No
Multi-stream support	Yes	No	No
Unordered data delivery	Yes	No	Yes
Security cookie against SYN flood attack	Yes	No	No
Built-in heartbeat (readability check)	Yes	No	No

Randall R. Stewart and Qiaobing Xie, “Stream Control Transmission Protocol: A Reference Guide”, Addison-Wesley, 2002, ISBN 0-201-72186-4.

“SCTP Primer”, Mon, Mar 1, 2004 03:35:54 PM
http://datatag.web.cern.ch/datatag/WP3/sctp/primer.htm

Slide 43

RFC3554 - On the use of SCTP with IPSEC

From the abstract of RFC 3554:

> "… This document describes functional requirements for IPsec and IKE to facilitate their use in securing SCTP traffic. In particular, we discuss additional support in the form of a new ID type in IKE [RFC2409] and implementation choices in the IPsec processing to accommodate for the **multiplicity of source and destination addresses** associated with a single SCTP association."

[Emphasis in the text above added by Maguire]

S. Bellovin, J. Ioannidis, A. Keromytis, and R. Stewart, 'On the Use of Stream Control Transmission Protocol (SCTP) with IPsec', *Internet Request for Comments*, vol. RFC 3554 (Proposed Standard), Jul. 2003 [Online]. Available: http://www.rfc-editor.org/rfc/rfc3554.txt

D. Harkins and D. Carrel, 'The Internet Key Exchange (IKE)', *Internet Request for Comments*, vol. RFC 2409 (Proposed Standard), Nov. 1998 [Online]. Available: http://www.rfc-editor.org/rfc/rfc2409.txt

Slide 44

RFC 4895 - Authenticated Chunks for SCTP

new Parameter Types

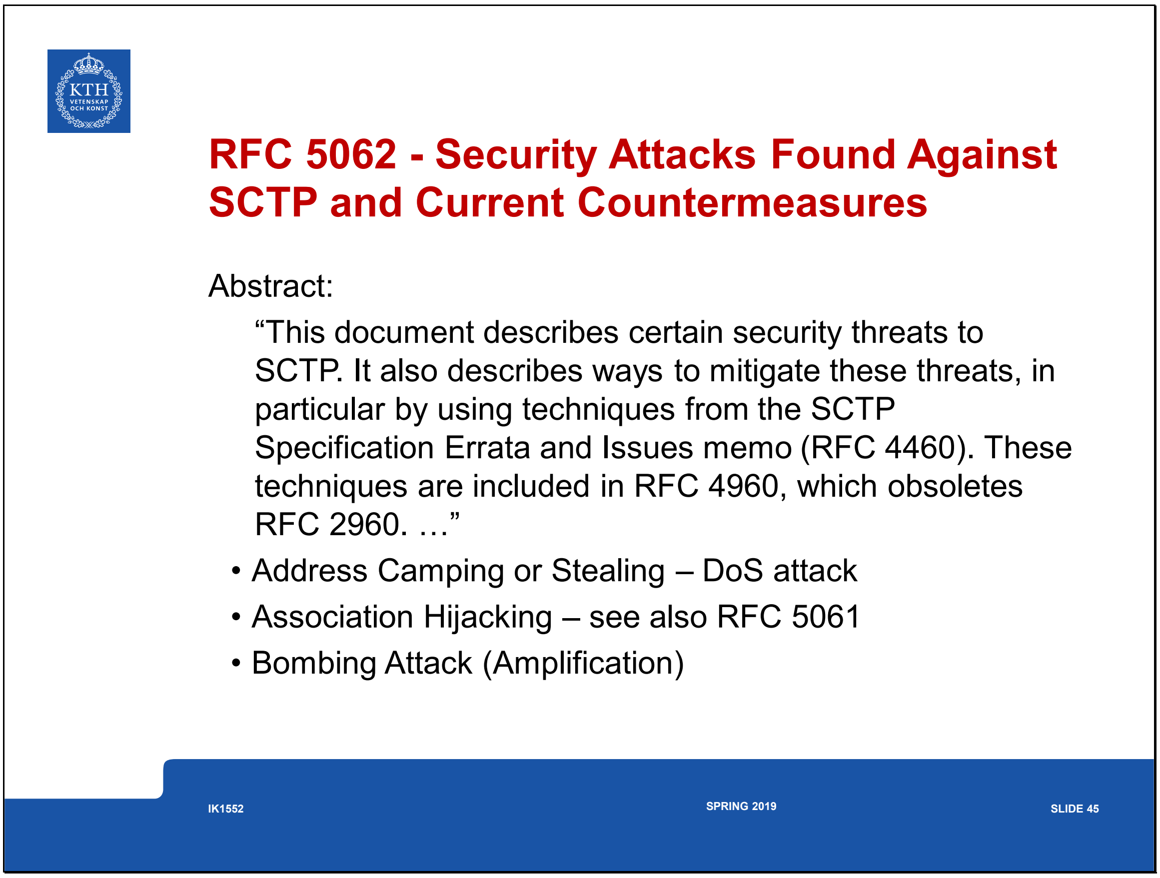
- Random Parameter (RANDOM) = 0x8002
- Chunk List Parameter (CHUNKS) = 0x8003
- Requested HMAC Algorithm Parameter (HMAC-ALGO) = 0x8004

new chunk Type: Authentication Chunk (AUTH) = 0x0F

new Error Cause: Unsupported HMAC Identifier = 0x0105

M. Tuexen, R. Stewart, P. Lei, and E. Rescorla, ‘Authenticated Chunks for the Stream Control Transmission Protocol (SCTP)’, *Internet Request for Comments*, vol. RFC 4895 (Proposed Standard), Aug. 2007 [Online]. Available: http://www.rfc-editor.org/rfc/rfc4895.txt

Slide 45



R. Stewart, M. Tuexen, and G. Camarillo, 'Security Attacks Found Against the Stream Control Transmission Protocol (SCTP) and Current Countermeasures', *Internet Request for Comments*, vol. RFC 5062 (Informational), Sep. 2007 [Online]. Available: http://www.rfc-editor.org/rfc/rfc5062.txt
R. Stewart, I. Arias-Rodriguez, K. Poon, A. Caro, and M. Tuexen, 'Stream Control Transmission Protocol (SCTP) Specification Errata and Issues', *Internet Request for Comments*, vol. RFC 4460 (Informational), Apr. 2006 [Online]. Available: http://www.rfc-editor.org/rfc/rfc4460.txt
R. Stewart, 'Stream Control Transmission Protocol', *Internet Request for Comments*, vol. RFC 4960 (Proposed Standard), Sep. 2007 [Online]. Available: http://www.rfc-editor.org/rfc/rfc4960.txt
R. Stewart, Q. Xie, M. Tuexen, S. Maruyama, and M. Kozuka, 'Stream Control Transmission Protocol (SCTP) Dynamic Address Reconfiguration', *Internet Request for Comments*, vol. RFC 5061 (Proposed Standard), Sep. 2007 [Online]. Available: http://www.rfc-editor.org/rfc/rfc5061.txt

Slide 46

Some of the SCTP RFCs

- M. Tuexen, I. Ruengeler, and R. Stewart, 'SACK-IMMEDIATELY Extension for the Stream Control Transmission Protocol', Internet Request for Comments, vol. RFC 7053 (Proposed Standard), Nov. 2013. http://www.rfc-editor.org/rfc/rfc7053.txt
- R. Stewart, M. Tuexen, and P. Lei, 'Stream Control Transmission Protocol (SCTP) Stream Reconfiguration', Internet Request for Comments, vol. RFC 6525 (Proposed Standard), Feb. 2012. http://www.rfc-editor.org/rfc/rfc6525.txt
- R. Stewart, M. Tuexen, K. Poon, P. Lei, and V. Yasevich, 'Sockets API Extensions for the Stream Control Transmission Protocol (SCTP)', Internet Request for Comments, vol. RFC 6458 (Informational), Dec. 2011. http://www.rfc-editor.org/rfc/rfc6458.txt
- M. Tuexen and R. Stewart, 'Stream Control Transmission Protocol (SCTP) Chunk Flags Registration', Internet Request for Comments, vol. RFC 6096 (Proposed Standard), Jan. 2011. http://www.rfc-editor.org/rfc/rfc6096.txt
- M. Tuexen, R. Seggelmann, and E. Rescorla, 'Datagram Transport Layer Security (DTLS) for Stream Control Transmission Protocol (SCTP)', Internet Request for Comments, vol. RFC 6083 (Proposed Standard), Jan. 2011. http://www.rfc-editor.org/rfc/rfc6083.txt
- X. Fu, C. Dickmann, and J. Crowcroft, 'General Internet Signaling Transport (GIST) over Stream Control Transmission Protocol (SCTP) and Datagram Transport Layer Security (DTLS)', Internet Request for Comments, vol. RFC 6084 (Experimental), Jan. 2011. http://www.rfc-editor.org/rfc/rfc6084.txt

Slide 47

More RFCs

- J. H. Salim and K. Ogawa, 'SCTP-Based Transport Mapping Layer (TML) for the Forwarding and Control Element Separation (ForCES) Protocol', Internet Request for Comments, vol. RFC 5811 (Proposed Standard), Mar. 2010 http://www.rfc-editor.org/rfc/rfc5811.txt
- M. Allman, K. Avrachenkov, U. Ayesta, J. Blanton, and P. Hurtig, 'Early Retransmit for TCP and Stream Control Transmission Protocol (SCTP)', Internet Request for Comments, vol. RFC 5827 (Experimental), May 2010 http://www.rfc-editor.org/rfc/rfc5827.txt
- R. Stewart, M. Tuexen, and G. Camarillo, 'Security Attacks Found Against the Stream Control Transmission Protocol (SCTP) and Current Countermeasures', Internet Request for Comments, vol. RFC 5062 (Informational), Sep. 2007 http://www.rfc-editor.org/rfc/rfc5062.txt
- C. Bestler and R. Stewart, 'Stream Control Transmission Protocol (SCTP) Direct Data Placement (DDP) Adaptation', Internet Request for Comments, vol. RFC 5043 (Proposed Standard), Oct. 2007 http://www.rfc-editor.org/rfc/rfc5043.txt
- R. Stewart, 'Stream Control Transmission Protocol', Internet Request for Comments, vol. RFC 4960 (Proposed Standard), Sep. 2007 http://www.rfc-editor.org/rfc/rfc4960.txt
- M. Tuexen, R. Stewart, P. Lei, and E. Rescorla, 'Authenticated Chunks for the Stream Control Transmission Protocol (SCTP)', Internet Request for Comments, vol. RFC 4895 (Proposed Standard), Aug. 2007 http://www.rfc-editor.org/rfc/rfc4895.txt
- M. Tuexen, R. Stewart, and P. Lei, 'Padding Chunk and Parameter for the Stream Control Transmission Protocol (SCTP)', Internet Request for Comments, vol. RFC 4820 (Proposed Standard), Mar. 2007 http://www.rfc-editor.org/rfc/rfc4820.txt
- R. Stewart, I. Arias-Rodriguez, K. Poon, A. Caro, and M. Tuexen, 'Stream Control Transmission Protocol (SCTP) Specification Errata and Issues', Internet Request for Comments, vol. RFC 4460 (Informational), Apr. 2006 http://www.rfc-editor.org/rfc/rfc4460.txt

Slide 48

More RFCs

- J. Rosenberg, H. Schulzrinne, and G. Camarillo, 'The Stream Control Transmission Protocol (SCTP) as a Transport for the Session Initiation Protocol (SIP)', Internet Request for Comments, vol. RFC 4168 (Proposed Standard), Oct. 2005 http://www.rfc-editor.org/rfc/rfc4168.txt
- L. Coene and J. Pastor-Balbas, 'Telephony Signalling Transport over Stream Control Transmission Protocol (SCTP) Applicability Statement', Internet Request for Comments, vol. RFC 4166 (Informational), Feb. 2006 http://www.rfc-editor.org/rfc/rfc4166.txt
- J. Pastor and M. Belinchon, 'Stream Control Transmission Protocol (SCTP) Management Information Base (MIB)', Internet Request for Comments, vol. RFC 3873 (Proposed Standard), Sep. 2004 http://www.rfc-editor.org/rfc/rfc3873.txt
- R. Stewart, M. Ramalho, Q. Xie, M. Tuexen, and P. Conrad, 'Stream Control Transmission Protocol (SCTP) Partial Reliability Extension', Internet Request for Comments, vol. RFC 3758 (Proposed Standard), May 2004 http://www.rfc-editor.org/rfc/rfc3758.txt
- S. Bellovin, J. Ioannidis, A. Keromytis, and R. Stewart, 'On the Use of Stream Control Transmission Protocol (SCTP) with IPsec', Internet Request for Comments, vol. RFC 3554 (Proposed Standard), Jul. 2003 http://www.rfc-editor.org/rfc/rfc3554.txt
- A. Jungmaier, E. Rescorla, and M. Tuexen, 'Transport Layer Security over Stream Control Transmission Protocol', Internet Request for Comments, vol. RFC 3436 (Proposed Standard), Dec. http://www.rfc-editor.org/rfc/rfc3436.txt
- L. Ong and J. Yoakum, 'An Introduction to the Stream Control Transmission Protocol (SCTP)', Internet Request for Comments, vol. RFC 3286 (Informational), May 2002 http://www.rfc-editor.org/rfc/rfc3286.txt
- L. Coene, 'Stream Control Transmission Protocol Applicability Statement', Internet Request for Comments, vol. RFC 3257 (Informational), Apr. 2002 http://www.rfc-editor.org/rfc/rfc3257.txt

Slide 49

Summary

This module has discussed:

- SCTP
- Message framing
- Multi-homing
- Multi-streaming
- How SCTP differs from TCP
- Measurements of an implementation (there are other implementations such as that included with [Stewart and Xie 2002]):
  - *http://www.sctp.de*
  - Linux Kernel SCTP *http://sourceforge.net/projects/lksctp*

See also: Randall Stewart's Home Page: http://people.freebsd.org/~rrs/

Randall R. Stewart and Qiaobing Xie, "Stream Control Transmission Protocol: A Reference Guide", Addison-Wesley, 2002, ISBN 0-201-72186-4.

Slide 50

¿Questions?