Transmission Control Protocol (TCP)

1

TCP: Услуги

- **RFCs 793, 1122, 1323, 2018, 2581, 2873, 2988, 3168, 4614**
- ***Надеждна комуникация между два приложни процеса (приложения)***
  - Гарантира, че данните ще бъдат доставени без загуба, дублирания или грешки.
  - През множество надеждни и/или ненадеждни мрежи
  - С контролиране на потока, грешките и задръстванията
- ***С използване на съединение***
  - Приложението изисква съединение за трансфер на данните
  - **Установяване на надеждно съединение**
    - 3-кратно ръкостискане гарантира надежден и синхронизиран старт на комуникацията между двете крайни точки
    - Съединението се установява между *двойка сокети*
      - **Сокет** = IP адрес + номер на порт
      - 1 сокет може да се използва за няколко съединения едновременно
  - **Разпадане на съединението**
    - Гарантиране доставката на всички данни преди разпадане на съединението
    - 4(3)-кратно ръкостискане
- ***От точка до точка*** *(E2E)*
  - Съединението има 2 крайни точки, идентифицирани чрез *сокети*.
  - Няма поддръжка на *multicasting* и *broadcasting*!

3

TCP: Услуги (прод.)

- ***Пълен дуплекс***
  - Крайните точки обменят данни в двете посоки едновременно
  - Всяка точка разполага с буфер за изпращане и получаване на данни
- ***Байтово-ориентирани***
  - Приложният процес предава своето съобщение към TCP
  - TCP го разглежда като непрекъснат поток от байтове
  - TCP може да **не** запази границите на съобщението
    - Може да изпрати съобщението веднага, или
    - Може да го буферира с цел акумулиране на по-голям обем от данни за изпращане наведнъж

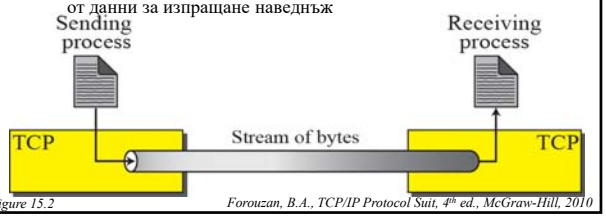
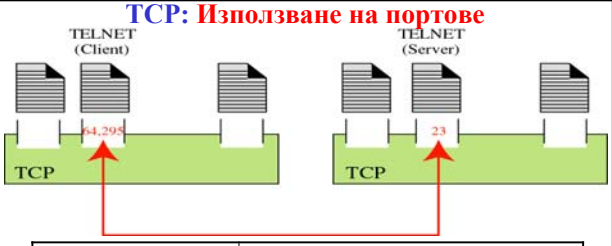


Figure 15.2 *Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010*

4

TCP: Използване на портове



Some well-known TCP ports	Application-layer Protocols
20	FTP, Data
21	FTP, Control
23	TELNET
25	SMTP
53	DNS
80	HTTP
110	POP3

Figure 12-3 *Forouzan, B.A., TCP/IP Protocol Suit, 2nd ed., McGraw-Hill, 2003*

5

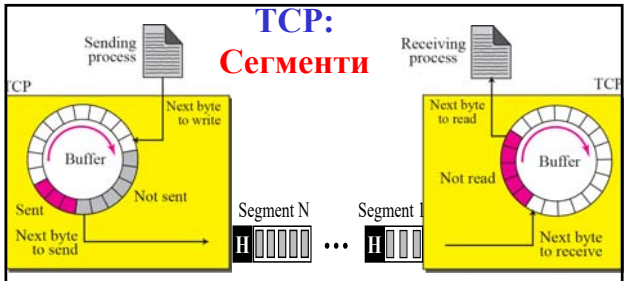
TCP: Трансфер на данни

- Данните от горния слой се разглеждат като логически поток от байтове
- Байтовете се номерират по ***mod*** 2^{32}
  - Започвайки от *номер, произволно избран и анонсиран* по време на установяване на съединението.
- Данните се буферират от TCP подателя и TCP получателя
  - Кръгов буфер
- Контрол на потока чрез плъзгащ се прозорец и кредитна схема за размера на прозореца
- Използване на *PUSH* флаг за принудително изпращане на всички данни, натрупани до момента в буфера (*end-of-block function*)
- Приложението може да укаже спешно предаване на данни (чрез *URGENT* флаг)

Figure 15.3 *Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010*

6

TCP: Сегменти



- Данните се предават разделени/обединени на/в сегменти
- Комуникиращите TCP обекти могат да договорят MAX размер на сегмента (***MSS***)
  - ***MSS*** ≤ 65515 B
- TCP подателят може да реши да:
  - Акумулира данните от няколко записа в 1 сегмент, или
  - Раздели данните от 1 запис на няколко сегмента

Figure 15.4 *Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010*

7

TCP сегмент: Формат

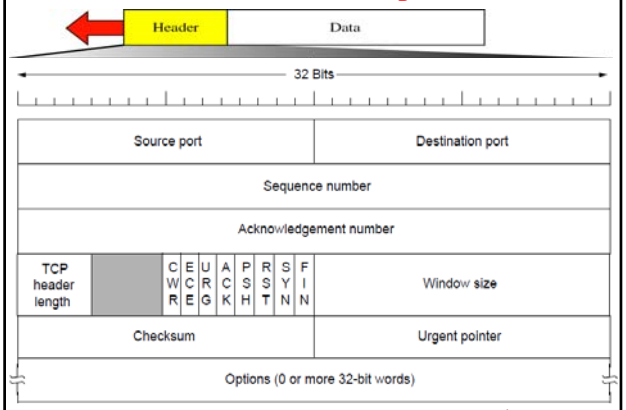


Figure 6-36 Tanenbaum, A.S. & Wetherall D.J., Computer Networks, 5th ed., Pearson, 2011

14

TCP сегмент: Формат (прод.)

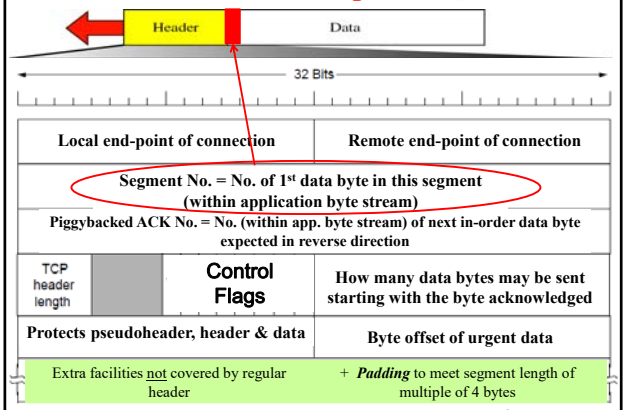


Figure 6-36 Tanenbaum, A.S. & Wetherall D.J., Computer Networks, 5th ed., Pearson, 2011

15

TCP: Контролни флагове

CWR	ECE	URG	ACK	PSH	RST	SYN	FIN

URG - *Urgent pointer* полето е валидно
ACK - *Acknowledgment number* полето е валидно
PSH - заявка за избутване на данните (*push*)
RST - за възобновяване на съединението
SYN - за първоначално установяване на съединението
FIN - за разпадане на съединението
ECE - изрично уведомяване за задръствания (*ECN-Echo*)
CWR - прозорецът на задръстванията е намален (*Congestion Window Reduced*)

16

Установяване на TCP съединение: 3-кратно ръкостискане

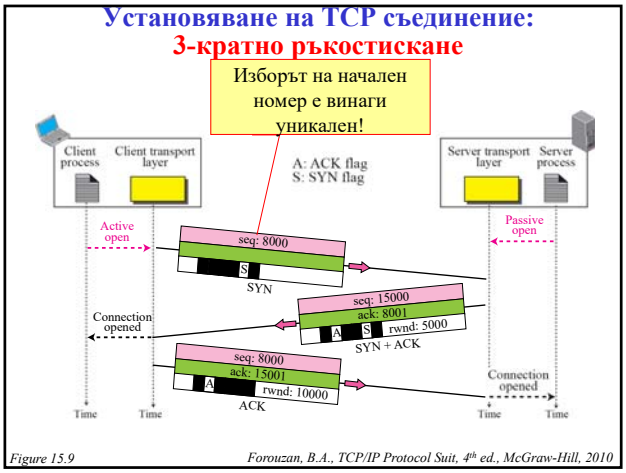


Figure 15.9 Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

25

Разпадане на TCP съединение: 3-кратно ръкостискане (със симетрично разпадане)

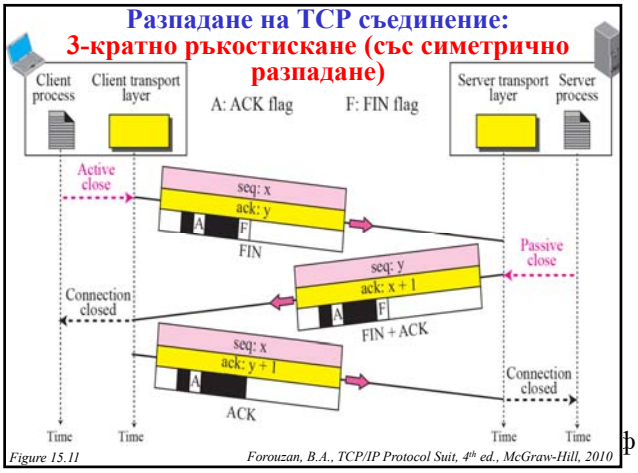


Figure 15.11 Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

27

Разпадане на TCP съединение: 4-кратно ръкостискане (с асиметрично разпадане)

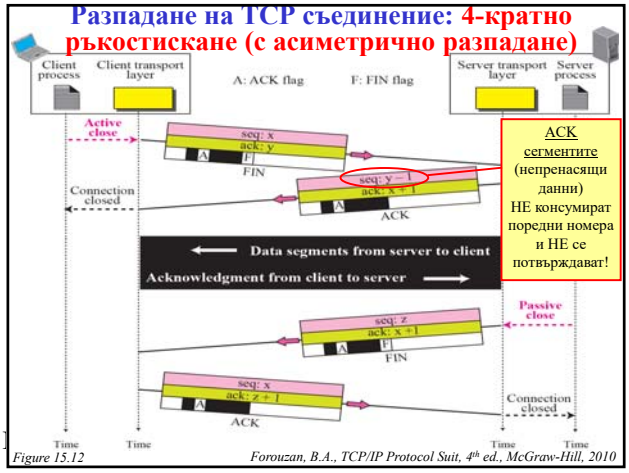


Figure 15.12 Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

28

TCP контрол на потока

29

TCP контрол на потока: Обратна връзка

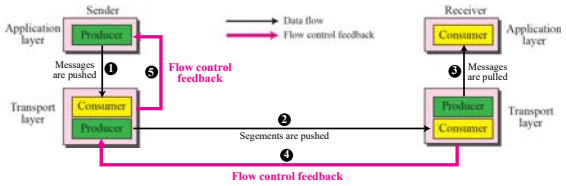
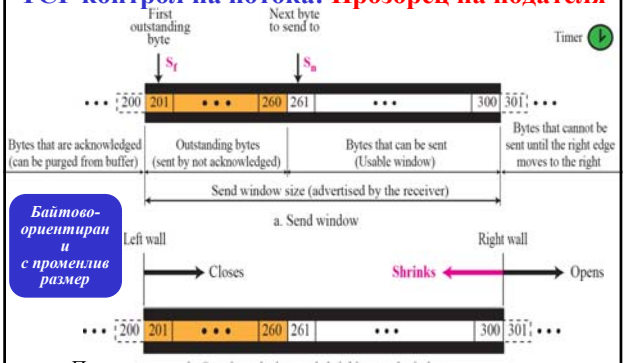


Figure 15.24 *Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010*

30

TCP контрол на потока: Прозорец на подателя



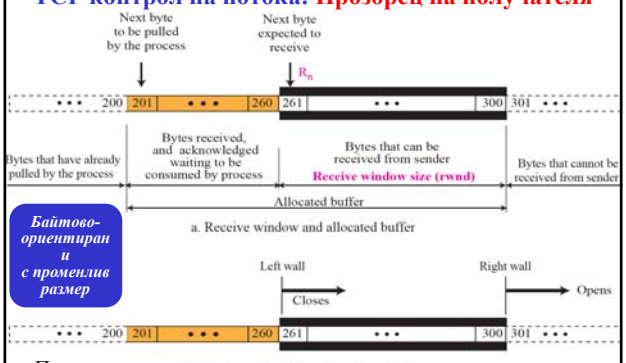
Байтово-ориентиран и с променлив размер

- Подател
  - НЕ е длъжен да изпраща целия прозорец наведнъж
  - НЕ е длъжен да изпраща данните веднага след пристигането им от горния слой

Figure 15.22 *Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010*

31

TCP контрол на потока: Прозорец на получателя



Байтово-ориентиран и с променлив размер

- Получател
  - НЕ е длъжен да изпраща веднага потвърждение
  - Може да изпрати потвърждение по всяко време

Figure 15.23 *Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010*

32

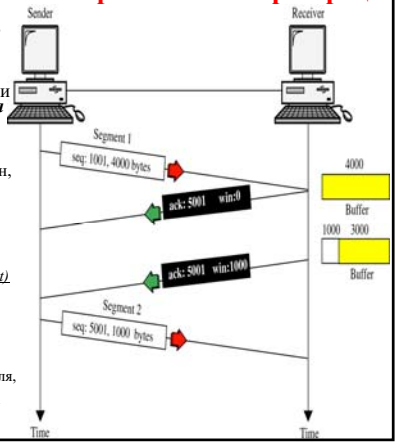
TCP контрол на потока: Пример

Figure 15.25 *Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010*

33

TCP: Управление на прозореца

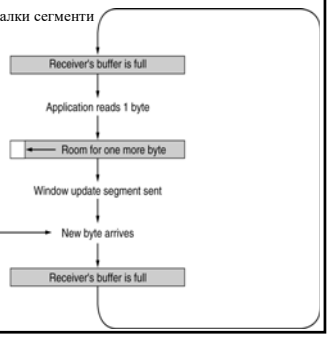
- Не е пряко обвързано с ACK механизъм
  - Специфична особеност на протоколите от транспортния слой
- Размерът на прозореца може да се увеличава или намалява от ***получателя*** динамично
  - Въз основа на размера на буфера му
  - Ако буферът му е пълен, получателят анонсира нулев размер на прозореца.
  - Подателят спира да предава сегменти
  - 2 изключения
    - Спешни данни *(urgent)* могат да се предават (например, за прекратяване на отдалечен процес)
    - Специален сегмент-проба се изпраща периодично за да се предупреди получателя, че подателят все още очаква анонсиране на ненулев размер на прозореца.



34

TCP: Синдром на глупавия прозорец

- Възниква, когато данните се обработват бавно от приложенията.
- ***Синдром, създаден от <u>подателя.</u>***
  - Когато предаващото приложение генерира данните бавно
    - Т.е. само по няколко байта наведнъж.
  - <u>Решение:</u> Алгоритм на Nagle
    - <u>Цел</u>: подателят да не изпраща малки сегменти
- ***Синдром, създаден от <u>получателя.</u>***
  - Когато получаващото приложение консумира данните бавно
  - <u>2 решения:</u>
    - Алгоритм на Clark
    - Забавено потвърждение
    - <u>Цел</u>: получателят да не иска малки сегменти



35

Синдром, създаден от подателя: Алгоритм на Nagle (прилаган от подателя)

- Стъпка 1
  - Изпращане на 1. порция данни, получени от предаващото приложение.
    - Дори, ако това е само 1 байт.
- Стъпка 2
  - Акумулиране на данни в изходящия буфер <u>докато</u>:
    - <u>Получателят изпрати обратно потвърждение (ACK)</u>, или
    - Се съберат <u>достатъчно данни</u> за запълване на:
      - <u>Сегмент с максимален размер (MSS)</u>, или
      - <u>½ от прозореца на подателя</u>
  - Изпращане на сегмент
- Стъпка 3
  - Повторяне на стъпка 2 до края на комуникацията

36

Синдром, създаден от получателя: Алгоритм на Clark (прилаган от получателя)

- Изпращане на потвърждение, веднага след пристигане на сегмент.
  - За да се подпомогне точното изчисление/актуализиране на RTT от страна на подателя
- Но се анонсира <u>нулев размер на прозореца</u>, <u>докато</u> във входящия буфер <u>не се освободи достатъчно място</u>, равно на:
  - <u>1 MSS</u>, или
  - <u>½ от буфера</u>

$$\text{available buffer space} \geq \text{MIN}\left(\frac{\text{buffer size}}{2}, \text{maximum segment size}\right)$$

37

Синдром, създаден от получателя: Забавено потвърждение (прилагано от получателя)

- Забавя се изпращането на потвърждение
- Докато не се освободи достатъчно място във входящия буфер

- **<u>Предимство</u>**
  - Намален трафик
    - Получателят не е длъжен да потвърждава всеки сегмент (особено, ако няма данни за предаване в обратната посока)
- **<u>Недостатък</u>**
  - Забавено потвърждение може да принуди подателя да изпрати отново някой сегмент
    - Затова потвържденията се бавят не повече от 500 ms

38

TCP контрол на грешките

39

TCP: Контрол на грешките

- Базиран на **<u>PAR</u>** схема
  - <u>Положителна квитанция с повторно предаване</u>
  - **P**ositive **A**cknowledgement with **R**etransmission
- Всеки TCP сегмент съдържа ***контролна сума*** в заглавната си част
  - За защита от грешки на целия TCP сегмент (вкл. *псевдозаглавната част*)
- Получател
  - Ако сегментът <u>не е повреден</u>, го потвърждава с ACK.
    - Кумулативни потвърждения
    - *Piggybacking*
    - Сегментите, използвани само за потвърждение (т.е. без данни), <u>не</u> консумират поредни номера и <u>не</u> се потвърждават.
  - Ако сегментът е <u>повреден</u>, го отхвърля <u>без</u> изпращане на отрицателна квитанция *(NAK)*.
  - Ако сегментът пристигне <u>непоред</u>, отлага ACK, докато не пристигнат всички липсващи сегменти, запълващи празнината.
- Подател
  - Ако таймерът се нулира преди пристигането на ACK, предава повторно сегмента.
    - Тъй като или сегментът е повреден или изгубен, или потвърждението му се изгубило.

40

TCP: Контролна сума

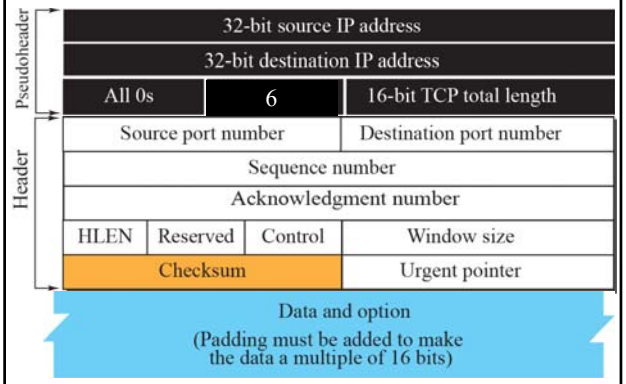
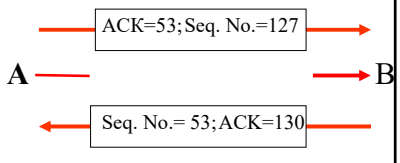


Figure 15.7 Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

41

TCP: Piggybacking



• **ACK=53** специфицира номера на **следващия байт данни**, който **A** очаква да получи от **B**.

• **Seq. No.=127** специфицира номера на **първия байт данни**, съдържащи се в сегмента, изпращан от **A** (т.е. **номера на сегмента**).

• В отговор **B** изпраща своя байт 53 и последващите го байтове данни, капсулирани в сегмент.

• В същото време **B** потвърждава успешното приемане на байтове, изпратени от **A** (т.е. байт 130 е следващият байт, очакван от **B**).

42

TCP: Нормално функциониране

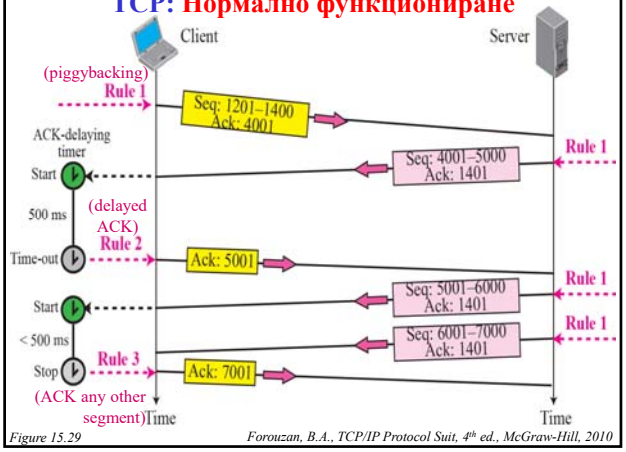


Figure 15.29 Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

43

TCP: Повреден сегмент

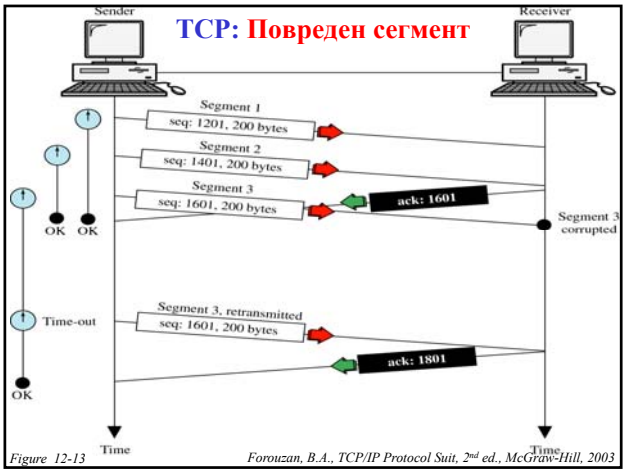


Figure 12-13 Forouzan, B.A., TCP/IP Protocol Suit, 2nd ed., McGraw-Hill, 2003

44

TCP: Изгубен сегмент

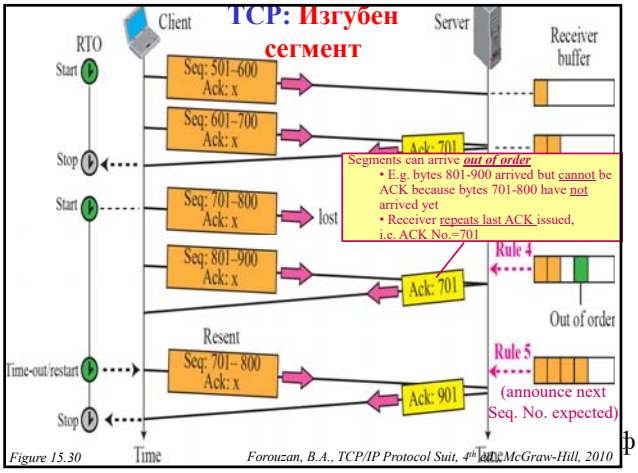


Figure 15.30 Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

45

TCP изгубен сегмент: Бързо повторно предаване

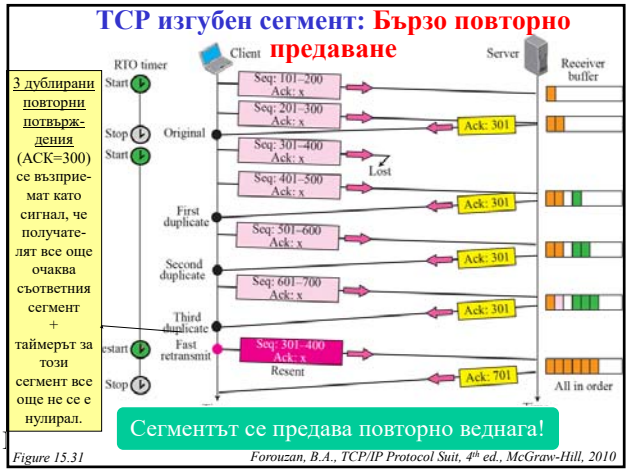


Figure 15.31 Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

46

TCP: Изгубено потвърждение
(коригирано чрез успешно получаване на следващите сегменти)

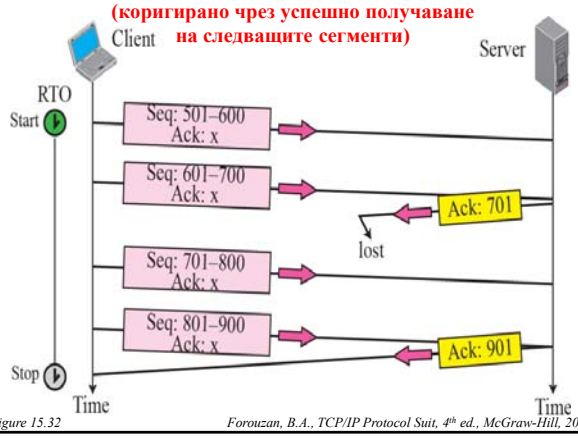


Figure 15.32 Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

47

TCP: Изгубено потвърждение
(коригирано чрез повторно предаване на сегмент)

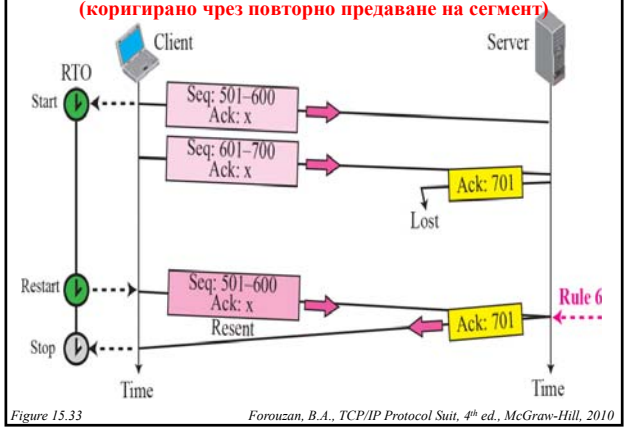


Figure 15.33 Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

48

TCP контрол на грешките: Проблеми

- Сегментите могат да се ***забавят при транзита***
  - Може да се наложи повторното им предаване
  - Но с ***различен диапазон от байтове*** в сравнение с първоначалното им предаване
- Изисква се внимателно администриране
  - За да се ***следи*** кои байтове са били правилно доставени досега

49

TCP контрол на задръстванията

61

TCP: Контрол на задръстванията

- Динамично манипулиране на размера на прозореца
- Подателят поддържа 2 прозореца
  - Прозорец, зададен от получателя *(receiver window, **rwnd**)*
    - Ако подателят се придържа към него, няма да възникнат проблеми поради препълване на буфер, но могат да възникнат проблеми поради задръствания в мрежата.
  - Прозорец, предизвикан от задръствания *(congestion window, **cwnd**)*
- Във всеки един момент подателят използва по-малкия от тези 2 прозореца
  - Т.е. ***swnd*** = **MIN** ***(rwnd, cwnd)***
- Основно действие
  - Когато таймерът се занули, подателят започва да предава по-малко (по обем) данни.
    - Намалява натоварването на мрежата
    - С цел облекчаване на задръстванията

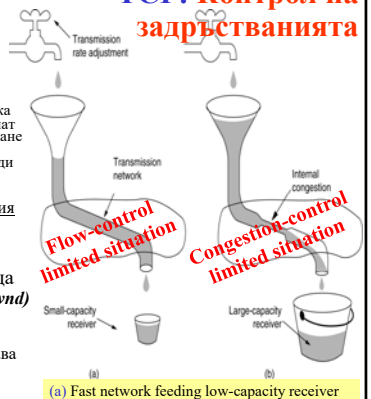


Figure 6-22 Tanenbaum, A.S. & Wetherall D.J., Computer Networks, 5th ed., Pearson, 2011

62

TCP: Контрол на задръстванията (прод.)

- Основно допускане от страна на TCP
  - Изтичане на времето за изчакване *(timeout)* се причинява от задръствания, а не поради грешки в IP пакетите.
- Вярно в кабелните мрежи
  - Комуникационните линии в днешно време са много надеждни (напр. при използване на влакнесто-оптични кабели)
  - Много рядко пакетите пристигат с грешки
- Погрешно в безжичните мрежи
  - Изключително ненадеждни
  - Пакети се губят или пристигат с грешки винаги, по всяко време.
  - Друг подход е необходим за контрол на задръстванията

63

TCP контрол на задръстванията: Фази

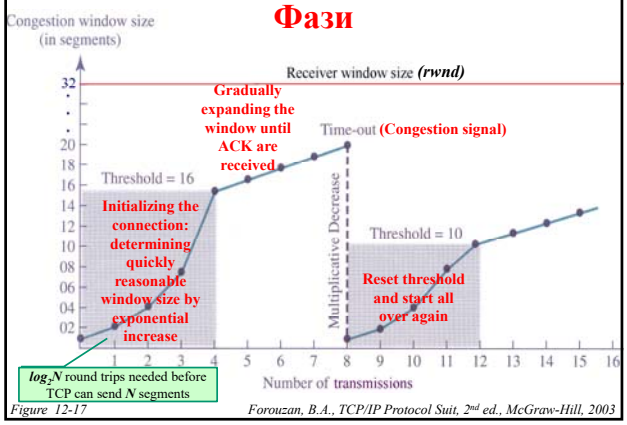


Figure 12-17 Forouzan, B.A., TCP/IP Protocol Suit, 2nd ed., McGraw-Hill, 2003

64

TCP контрол на задръстванията: 1. фаза – Бавен старт (*Slow Start*)

Подател:

- Първо изпраща **1 MSS** (сегмент с MAX размер)
- Ако получи потвърждение преди *timeout*
  - Удвоява ***cwnd***
  - Изпраща 2 MSS
- Ако пак получи потвърждение преди *timeout*
  - Удвоява ***cwnd***
  - Изпраща 4 MSS
- …
- Увеличава експоненциално ***cwnd*** до достигане на прага ***(ssthresh)***
  - ***ssthresh*** първоначално е равен на ½ от *rwnd*
- След това преминава към 2. фаза (вж. следващия слайд)

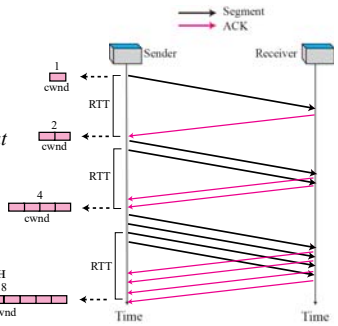
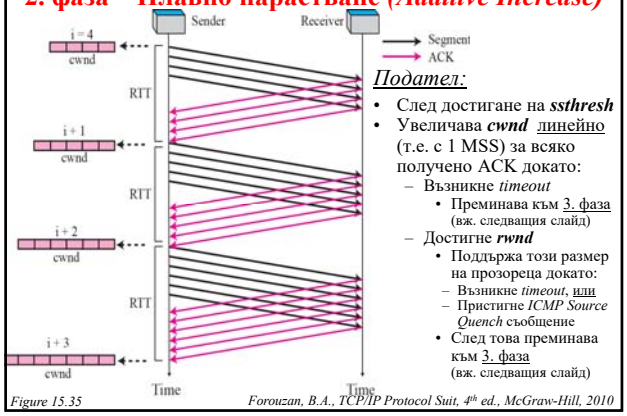


Figure 15.34 Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

65

TCP контрол на задръстванията: 2. фаза – Плавно нарастване *(Additive Increase)*



Подател:

- След достигане на ***ssthresh***
- Увеличава ***cwnd*** линейно (т.е. с 1 MSS) за всяко получено ACK докато:
  - Възникне *timeout*
    - Преминава към 3. фаза (вж. следващия слайд)
  - Достигне ***rwnd***
    - Поддържа този размер на прозореца докато:
      - Възникне *timeout*, или
      - Пристигне *ICMP Source Quench* съобщение
    - След това преминава към 3. фаза (вж. следващия слайд)

Figure 15.35 Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

66

TCP контрол на задръстванията: 3. фаза – Рязко свиване на прозореца *(Multiplicative Decrease)*

- Агресивно
  - Защото на мрежата ѝ е трудно да се възстанови от претоварването
- *Подател:*
  - Ако възникне *timeout*
  - Намалява ***ssthresh*** до ½ от ***cwnd***
  - Връща се към 1. фаза (***Slow Start***)
    - Т.е. възвръща първоначалната стойност на ***cwnd*** (=1 MSS)

67

TCP контрол на задръстванията: Резюме

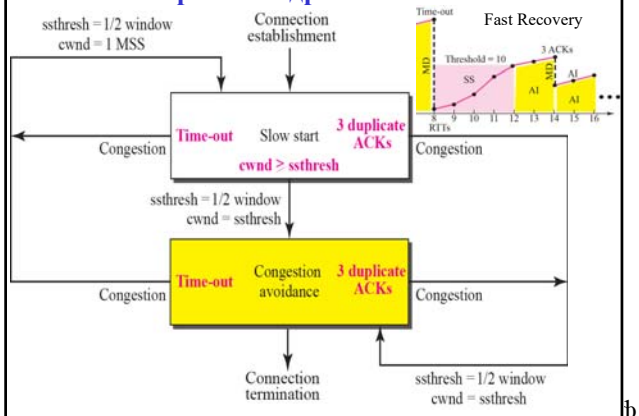


Figure 15.36 Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

68

TCP: Опции

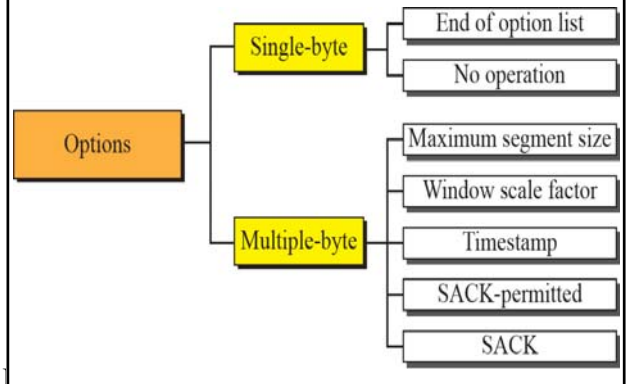


Figure 15.41 Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010

69

TCP: Дизайн

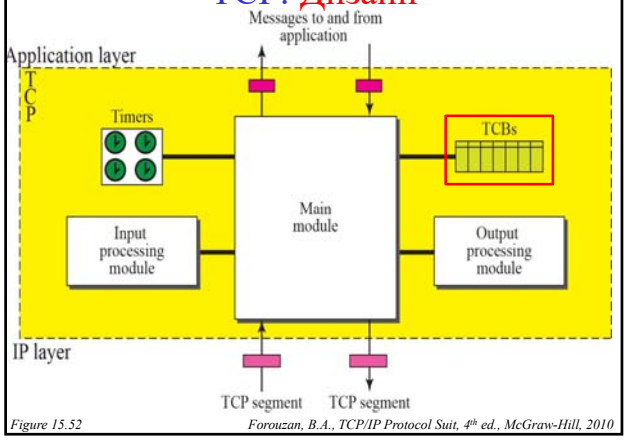
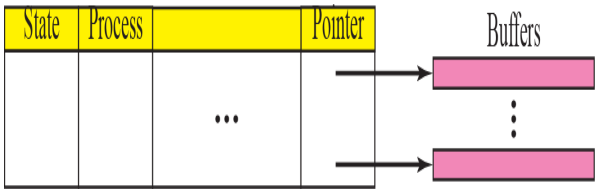


Figure 15.52 *Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010*

76

TCP дизайн: Transmission Control Blocks (TCBs)



- Съдържат **информация за състоянието** на всяко едно активно съединение
- ***Състояние*** на съединението според диаграмата за преход между отделните състояния (фази) на TCP протокола
- ***Процес*** – специфицира процеса като клиент или сървър
- ***Други полета*** – IP адреси, портове и интерфейси, участващи в съединението, текущ размер на прозореца, поредни номера, RTT, RTO, размер на буфера, ...
- ***Указател*** – към буфера, където се съхраняват данните.

Figure 15.53 *Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010*

77

TCP: Реализации

Measure	RFC 1122	TCP Tahoe	TCP Reno	NewReno
RTT Variance Estimation	✓	✓	✓	✓
Exponential RTO Backoff	✓	✓	✓	✓
Karn's Algorithm	✓	✓	✓	✓
Slow Start	✓	✓	✓	✓
Dynamic Window Sizing on Congestion	✓	✓	✓	✓
Fast Retransmit		✓	✓	✓
Fast Recovery			✓	✓
Modified Fast Recovery				✓

- Бързо възстановяване *(fast recovery)*
  - В случай на бързо повторно предаване, прозорецът се намалява наполовина и директно се преминава към 2. фаза (плавно нарастване на прозореца)
- Модифицирано бързо възстановяване
  - Подобрена реакция при загуба на 2 сегмента от един прозорец

Table 22.5 *Stallings, W., Data and Computer Communications, 9th ed., Prentice Hall, 2011*

79

Stream Control Transmission Protocol (SCTP)

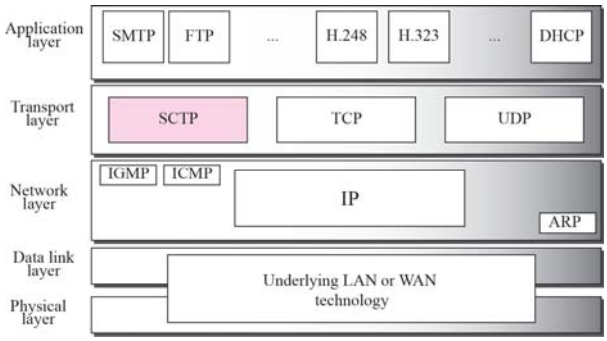


Figure 16.1 *Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010*

84

SCTP

- Дефиниран в RFC 2960
- Първоначално стандартизиран от IETF SIGTRAN WG за транспортиране на SS7 сигнализация през IP мрежи
- По-късно еволюира до транспортен протокол с общо предназначение, осигуряващ надеждно, пълно-дуплексно съединение с подобрени опции за доставка.
  - Необходим за нови приложения като IP телефония (VoIP), ISDN over IP, media gateway control, ...

Protocol	*Port Number*	*Description*
IUA	9990	ISDN over IP
M2UA	2904	SS7 telephony signaling
M3UA	2905	SS7 telephony signaling
H.248	2945	Media gateway control
H.323	1718, 1719, 1720, 11720	IP telephony
SIP	5060	IP telephony

Table 16.1 *Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010*

85

SCTP: Прилики с TCP

- ***Логическо съединение***
  - *Асоциация*
- ***Пълен дуплекс***
- ***Надежден транспорт***
- ***Поредни номера*** за контрол на потока и на грешките
  - Същия диапазон като TCP, т.е. $(0 , 2^{32}-1)$.
- ***Piggybacking***
- ***Контролна сума***
  - **32** бита (SCTP) / **16** бита(TCP)
- ***Бързо повторно предаване*** в случай на:
  - **4** SACKs (SCTP) / **3** ACKs (TCP)
- ***Бързо възстановяване***
- ***Контрол на задръстванията*** (същите фази като TCP)
  - Но прилагани поотделно за всеки поток *(stream)*

86

SCTP: Разлики с TCP

- ***Ориентиран към съобщения, а не към байтове.***
  (запазва границите на съобщенията на приложния слой)
- **Номерира парчета от данни** *(chunks)*
  - TCP номерира байтове
- ***Последователни номера се използват само за потвърждение на данни (data chunks)***
  - *Control chunks* се потвърждават с други *control chunks*
- ***Потвърждава последния получен номер, не следващия очакван номер.***
  - Т.е. ACK_X означава 'Изпрати ми X+1'
- ***Няма опции в заглавната част***
  - Опциите се задават чрез дефиниране на нови *chunk* видове
- Поддръжка на ***множество потоци от данни (multi-streaming)*** – ако един поток е блокиран, останалите продължават да доставят данни, аналогично на магистралните платна.
  - TCP използва само 1 поток от данни
- ***Multi-homing*** (хостовете могат да дефинират няколко IP адреса / мрежови интерфейси във всеки край на комуникацията за постигане на отказоустойчивост; ако един от пътищата за доставка се провали, започва използването на друг)
  - TCP използва само по 1 IP адрес във всеки край на комуникацията

87

SCTP: Multi-streaming

- Възможност за предаване едновременно на няколко независими потока данни, принадлежащи към една и съща асоциация.
- Поток
  - Има свои собствени характеристики
  - До голяма степен е независим от другите потоци
  - Може да е:
    - ***Строго подреден и надежден***
    - ***Надежден*** – данните се доставят към приложението, веднага след пристигането им.
    - ***Частично надежден*** – приложението определя "време на живот" за данните; данните не се предават повторно.
    - ***Неподреден*** – подобно на UDP; данните са независими по отношение на другите данни, изпратени по-рано; затова се запазват в буфер веднага при пристигането им.

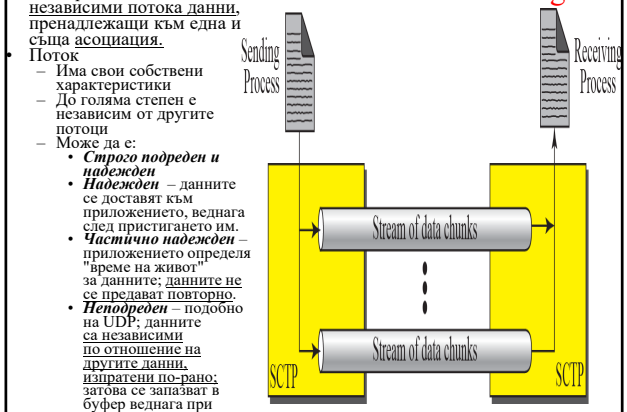


Figure 16.2 *Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010*

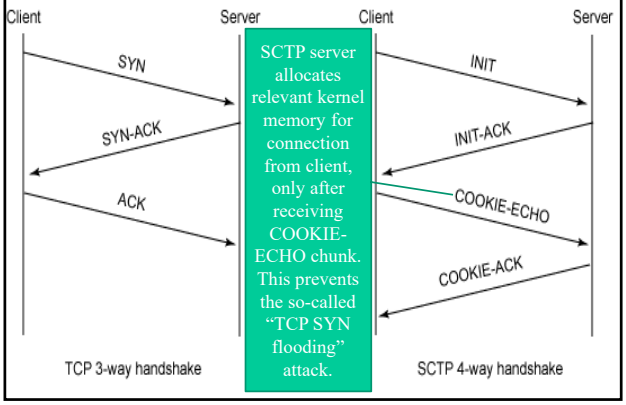
88

SCTP vs. TCP & UDP

Feature	UDP	TCP	SCTP
Connection oriented	No	Yes	Yes
Reliable transport	No	Yes	Yes
Unreliable transport	Yes	No	Yes
Preserve message boundaries	Yes	No	Yes
Ordered delivery	No	Yes	Yes
Unordered delivery	Yes	No	Yes
Data checksum	Yes	Yes	Yes
Checksum size (bits)	16	16	32
Path MTU	No	Yes	Yes
Flow-, error-, congestion control	No	Yes	Yes
Multiple streams	No	No	Yes
Multi-homing support	No	No	Yes

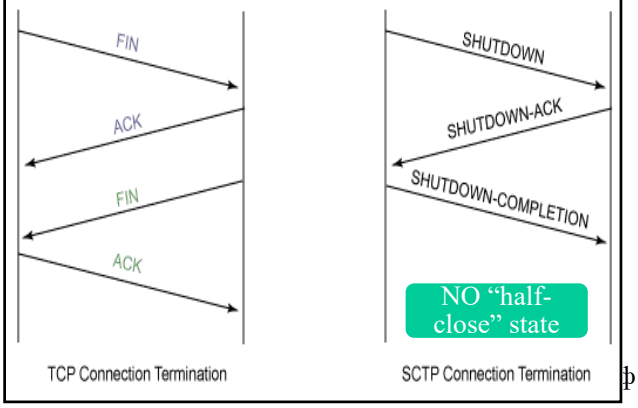
89

SCTP установяване на съединение: 4-кратно ръкостискане



90

SCTP разпадане на съединение: 3-кратно ръкостискане



91

SCTP: Пакет

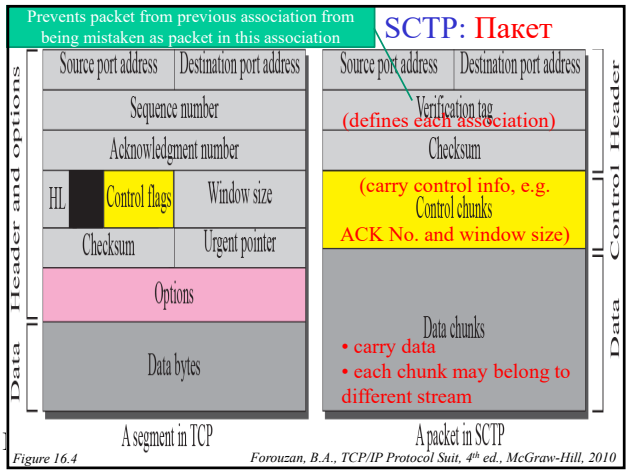


Figure 16.4 *Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010*

92

SCTP: Пакети, парчета *(chunks)* и потоци

- Асоциацията може да изпрати много пакети
- Пакетът може да съдържа няколко парчета данни *(chunks)*
- Парчета могат да принадлежат към различни потоци

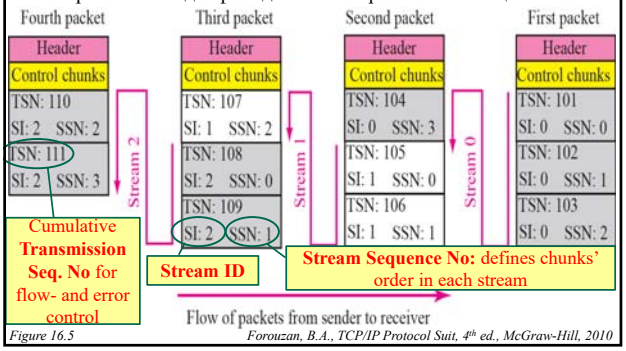
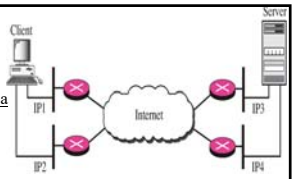


Figure 16.5 *Forouzan, B.A., TCP/IP Protocol Suit, 4th ed., McGraw-Hill, 2010*

94

SCTP: Multi-homing



- Хостове
  - Могат да използват <u>няколко IP адреса</u>
  - <u>Обвързани</u> едновременно <u>с една асоциация</u>
- SCTP асоциация
  - Може да използва множество налични IP адреси
  - Setup (*INIT* chunk) – всяка крайна точка посочва <u>списък с налични IP адреси</u>
  - Setup (*INIT-ACK* chunk) – крайните точки избират двойка валидни IP адреси като <u>основни *(primary)*</u>, между които се осъществява <u>предаването на данните по подразбиране</u> (всички други адреси се маркират като вторични / *secondary*)
- Път за предаване
  - Наблюдаван чрез изпращане на <u>HEARTBEAT</u> chunks, за <u>изпробване достижимостта</u> на даден адрес.
  - Счита се за <u>неактивен</u>, ако предаването по него се провали многократно (<u>данните се предават повторно към избран вторичен адрес</u>).
  - Приложението се информира за състоянието на пътя (в случай на промени)
    - Приложението може да инструктира SCTP да използва друг път за основен

95