

Протокол DCCP (Datagram Congestion Control Protocol; RFC-4336, -4340)



Транспортный протокол, который использует двунаправленные уникастные соединения с управлением перегрузкой для ненадежной доставки дейтограмм. DCCP предназначен для приложений, которые реализуют поточную схему TCP, но имеют приоритет для своевременной доставки данных с сохранением порядка кадров или требуют надежности, или которым нужен механизм подавления перегрузки, отличный от TCP

- До настоящего времени приложения использовали либо TSP, чья надежность и гарантия упорядочения доставки давалась за счет неопределенно большой задержки, или UDP и независимого механизма управления перегрузкой (или вообще с отсутствием подавления перегрузки).
- Протокол DCCP будет предоставлять стандартный способ управления перегрузкой и позволит использовать механизм **ECN** (Explicit Congestion Notification)
- Протокол DCCP предназначен для приложений, которые не требуют параметров **SCTP** [Stream Control Transmission Protocol, RFC 2960], таких как упорядоченная доставка при нескольких потоках.
- Протокол DCCP обеспечивает надежное согласование параметров при установлении соединения.
- В DCCP было решено не использовать менеджер управления перегрузкой [RFC 3124], который допускает несколько одновременных потоков между отправителем и получателем.

- Протокол DCCP предназначен для приложений, которые реализуют поточную схему TSP, но имеют приоритет для своевременной доставки данных с сохранением порядка кадров или требуют надежности, или которым нужен механизм подавления перегрузки, отличный от TSP
- До настоящего времени такие приложения использовали либо TSP, чья надежность и гарантия упорядочения доставки давалась за счет неопределенно большой задержки, или UDP и независимого механизма управления перегрузкой (или вообще с отсутствием

- Одной из целей DCCP было максимальное облегчение для UDP приложений перехода на DCCP, когда он будет внедрен.

Протокол DCCP имеет следующие характеристики:

- Реализует поток дейтограмм с подтверждением получения, но без повторной отправки.
- ненадежный диалог установления и разрыва соединения.
- Надежное согласование параметров.
- Выбор механизмов подавления перегрузки, дружественных по отношению к TCP, включая TCP-подобное управление перегрузкой (CCID 2) и управление потоком, дружественное TCP [RFC 3448] (CCID 3). CCID 2 использует разновидность TCP-механизма управления перегрузкой, и приемлемо для потоков, которые стремятся воспользоваться преимуществами доступной полосы, CCID 3 пригодно для потоков, которые требуют более стабильной скорости передачи.

- Опции, которые говорят отправителю с высокой надежностью, какие пакеты достигли получателя, были ли эти пакеты помечены ECN [RFC 3168 и RFC 3540], повреждены, или отброшены во входном буфере получателя.
- Управление перегрузкой, снабженное встроенной индикацией явной перегрузки ECN (Explicit Congestion Notification).
- Механизмы, позволяющие серверу избежать поддержки состояний неподтвержденных попыток соединений.
- Выявление MTU пути [RFC 1191].

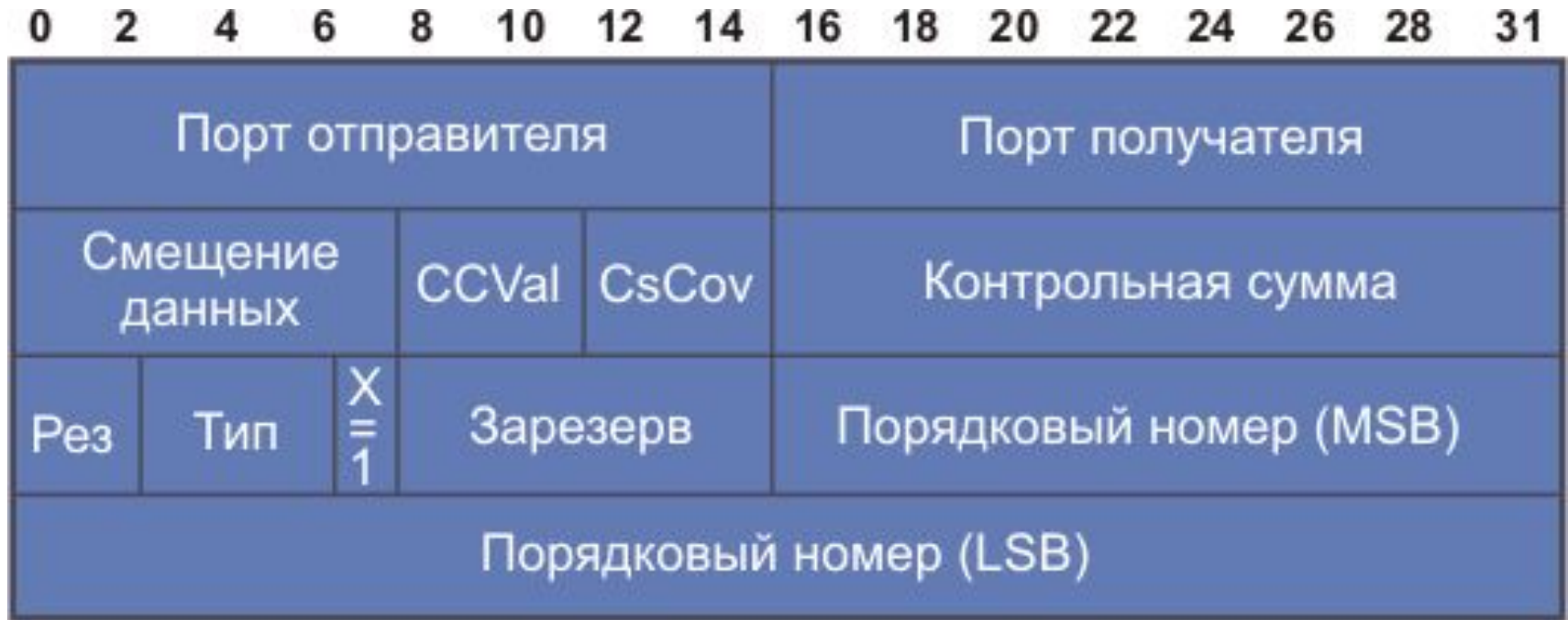
Отличия DCCP от TCP

- **Поток пакетов.** DCCP является протоколом для потоков пакетов, а не потоков байт.
- **Ненадежность.** DCCP никогда не пересылает дейтограммы повторно.
- **Порядковые номера пакетов.** Порядковые номера относятся к пакетам, а не байтам. Каждому пакету, посылаемому DCCP, присваивается новый порядковый номер, это относится и к пакетам подтверждений. Это позволяет получателю пакетов DCCP детектировать потери подтверждений; смотри раздел “Sequence Number Validity” в [DCCP].
- **Обширное пространство для опций** (до 1020 байт).
- **Согласование параметров.** Это является базовым механизмом, с помощью которого партнеры согласуют значения параметров или свойства соединения.

- **Выбор управления перегрузкой.** Партнеры могут использовать разные механизмы управления перегрузкой. В соединении $A \leftarrow B$, информационные пакеты, посланные от $A \rightarrow B$ могут использовать алгоритм **CCID 2**, а пакеты данных от $B \rightarrow A$ могут использовать **CCID 3**.
- **Различные форматы подтверждений.** CCID-соединения определяет то, какой объем данных должен быть передан в *ack*. В CCID 2 (TCP-подобном), посылается один *ack* на 2 пакета, а каждый *ack* должен оповещать, какие конкретно пакеты получены (опция Ack Vector); в CCID 3 (TFRC), посылается в среднем один *ack* за время RTT, а *ack* должны сообщать как минимум о длинах последних интервалов потерь.
- **Отсутствие приемного окна.** DCCP является протоколом управления перегрузкой, а не протоколом управления потоками.
- **Разделение различных видов потерь.** Опция потерянных данных (Data Dropped) позволяет одному из партнеров сообщить, что пакет был потерян из-за повреждения,

- **Определение подтверждения.** В TSP получение пакетов подтверждается, только когда они ставятся в очередь для передачи приложению. В протоколе DCCP это делается не так. Получение пакета подтверждается, когда обработаны поля его опций.
- **Встроенная поддержка мобильности.**

Заголовок DCCP



Если **X** равно нулю, передаются только младшие (LSB) 24 бита порядкового номера, а базовый заголовок имеет длину 12 байт. Смещение (8 бит) от начала заголовка пакета DCCP первого октета прикладных данных (выражается в 32-битных словах).

Checksum Coverage (**CsCov** – покрытие контрольным суммированием)

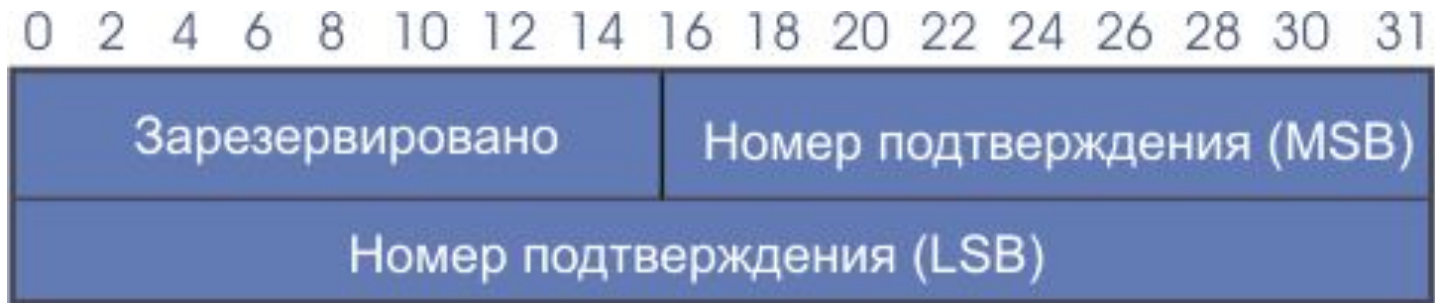
Базовый заголовок пакетов DSCP при X=0

Порт отправителя			Порт получателя		
Data Offset		CCval	CsCov	Контрольная сумма	
Рез.	Тип	X=0	Порядковый номер		

Если $X=1$, в заголовке используется 48-разрядные порядковые номера

Каждому механизму управления перегрузкой, поддерживаемому DSCP, присвоен идентификатор, или CCID: номер в диапазоне от 0 до 255

Формат подзаголовок номера подтверждения



Формат пакета DCCP-запроса

0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 31

Базовый DCCP-заголовок с $X=1$ (16 байт)
с *тип*=0 (DCCP-запрос)

Код сервиса

Опции и заполнитель

Прикладные данные

Формат пакета DCCP-отклика

0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 31

Базовый DCCP-заголовок с $X=1$ (16 байт)
с *тип* = 1 (DCCP-Response)

Опции и заполнитель

Прикладные данные

Формат пакета DCCP-Data

0 2 4 6 8 10 12 14 16 18 20 22 24 26 28 30 31

Базовый DCCP-заголовок с $X=1$ (16 или 12 байт)
с *тип* =2 (DCCP-Data)

Опции и заполнитель

Прикладные данные

TFRC

- Протокол **TFRC** (TCP Friendly Rate Control; RFC-3448, -4342, -4828) предоставляет механизм управления перегрузкой для уникастных потоков в Интернет. Реализованный способ управления перегрузкой может быть использован, например, в транспортном протоколе RTP, в приложении с контролем перегрузки в виртуальном канале точка-точка. RFC-3448, -4342, -4828.

Выражение для пропускной способности

TFRC спроектирован так, чтобы обеспечивать справедливое распределение ресурсов при конкуренции с другими TCP потоками, где под "справедливой" подразумевается ситуация, когда поток получает до двух раз большую полосу, чем TCP при тех же условиях. Таким образом, TFRC следует использовать, только когда приложение требует почти постоянной пропускной способности, в частности, избегая двойного снижения пропускной способности при потере одного пакета, как это происходит в случае TCP. Для приложений, которым нужно передать как можно больше данных за ограниченное время, следует порекомендовать TCP, или, если надежность не требуется, можно применить схемы управления перегрузкой Additive-Increase, Multiplicative-Decrease (**AIMD**).

$$X = \frac{s}{R \times \text{SQRT}(2 \times b \times p / 3) + (t_RTO \times (3 \times \text{SQRT}(3 \times b \times p / 8) \times p \times (1 + 32 \times p^2)))}$$

где **X** - скорость передачи в байтах в сек,

s длина пакета в байтах,

R = RTT в секундах,

p вероятность потери сегментов (между 0 и 1.0).

t_RTO является значением времени таймаута повторной передачи в секундах ($t_RTO = 4 \times RTT$),

b – число пакетов, подтверждаемых одним TCP ack.