

Network Working Group  
Request for Comments: 1990  
Obsoletes: 1717  
Category: Standards Track

K. Sklower  
University of California, Berkeley  
B. Lloyd  
G. McGregor  
Lloyd Internetworking  
D. Carr  
Newbridge Networks Corporation  
T. Coradetti  
Sidewalk Software  
August 1996

## The PPP Multilink Protocol (MP) Протокол PPP Multilink (MP)

### Статус документа

Этот документ содержит проект стандартного протокола Internet для сообщества Internet и служит приглашением к дискуссии в целях совершенствования и развития протокола. Текущее состояние стандартизации и статус протокола можно узнать из документа Internet Official Protocol Standards (STD 1).

### Тезисы

Этот документ предлагает метод расщепления, сборки и упорядочения дейтаграмм, передаваемых через множество логических каналов данных. Исходным мотивом этой работы послужило желание воспользоваться множеством каналов ISDN, но она в равной степени применима в любой ситуации, когда две системы соединены множеством каналов PPP, включая асинхронные. Параллельная работа каналов обеспечивается с помощью новых опций и протоколов PPP [2].

Различия между имеющейся спецификацией PPP Multilink (RFC 1717) и данным документом указаны в разделе 11. Все системы, где реализованы дополнительные ограничения, заданные в этом документе, будут совместимы с реализациями RFC 1717.

### Благодарности

Авторы выражают особую признательность Fred Baker из ACC, Craig Fox из Network Systems, Gerry Meyer из Spider Systems, Dan Brennan из Penril Datability Networks, Vernon Schryver из SGI (за всеобъемлющее обсуждение вопросов заполнения) и членам рабочих групп IP over Large Public Data Networks и PPP Extensions за полезные дискуссии по теме работы.

## Оглавление

1. Введение.....	2
1.1. Мотивация.....	2
1.2. Функциональное описание.....	2
1.3. Соглашения.....	2
2. Общий обзор.....	2
3. Формат пакетов.....	4
3.1. Заполнение.....	5
4. Компромисс между размером буферов и потерями фрагментов.....	5
4.1. Обнаружение потери фрагментов.....	5
4.2. Требования к размеру буфера.....	6
5. Расширения PPP LCP.....	6
5.1. Типы конфигурационных опций.....	6
6. Инициирование использования мультисканальных заголовков.....	9
7. Закрытие каналов связи.....	9
8. Взаимодействие с другими протоколами.....	9
9. Вопросы безопасности.....	9
10. Литература.....	9
11. Отличия от RFC 1717.....	9
11.1. Согласование параметров мультисканала.....	10
11.2. Определение стартового порядкового номера.....	10
11.3. Принятое по умолчанию значение MRRL.....	10
11.4. Запрет Config-Nak для EID.....	10
11.5. Однородность порядковых номеров.....	10
11.6. Начало и прекращение использования мультисканальных заголовков.....	10
11.7. ручная настройка и добавление в связку.....	10
12. Адреса авторов.....	10

## 1. Введение

### 1.1. Мотивация

Интерфейсы ISDN BRI и PRI обеспечивают возможность организации одновременно множества каналов между системами. Это позволяет предоставлять пользователям полосу канала по запросу (bandwidth on demand) с дополнительной оплатой используемых ресурсов. Предыдущие предложения (например, Leifer и др., [1]) для передачи протоколов Internet по каналам ISDN ставили целью обеспечение такой возможности.

Существуют варианты синхронизации множества потоков на битовом уровне (BONDING-предложения), но эти варианты не получили широкого распространения и могут требовать использования дополнительного оборудования. Таким образом, реализация решения на программном уровне может быть полезна, по крайней мере, в качестве предварительной меры.

Существуют другие варианты, в которых предоставляется полоса по запросу, типа использования коммутируемых линий 28800 бит/с для резервирования выделенных линий или использования дополнительных X.25 SVC, в которых размер окна ограничен 2 в силу международных соглашений.

Простейший из возможных алгоритмов чередования пакетов между каналами на основе их доступности (такой подход может называться алгоритмом Bank Teller) может приводить к нежелательным побочным эффектам в виде изменения порядка доставки пакетов.

Путем использования 4-байтовых порядковых номеров в заголовке и простых правил синхронизации можно распределять пакеты между параллельными каналами таким путем, что порядок пакетов не будет нарушаться или, по крайней мере, вероятность такого события будет достаточно мала.

### 1.2. Функциональное описание

Обсуждаемый здесь метод похож на многоканальный протокол, описанный в стандарте ISO 7776 [4], но предлагает дополнительную возможность расщепления и последующей сборки пакетов, что ведет к снижению задержки и может увеличивать эффективное значение максимального размера принимаемого блока (MRU<sup>1</sup>). Кроме того, здесь не требуется подтверждений на канальном уровне, хотя поддерживается такая возможность.

Многоканальность основана на согласовании опции LCP, позволяющем системе показать своему партнеру возможность объединения множества физических каналов в одну «связку» (bundle). Использование нескольких связок может потребоваться взаимодействующим системам лишь в исключительных случаях.

Многоканальность согласуется на этапе первоначального согласования опции LCP. Система показывает своему партнеру желание использовать многоканальное соединение путем передачи опции multilink в процессе начального согласования опции LCP. Это согласование указывает перечисленные ниже свойства.

1. Предлагающая опцию система может объединять множество физических каналов в один логический.
2. Система может принимать модули данных вышележащего протокола (PDU<sup>2</sup>), фрагментированные с использованием описанного ниже заголовка multilink, и собирать из них исходные PDU для обработки.
3. Система способна принимать PDU размером N октетов, где N задается как часть опции, если N превышает MRU на отдельном физическом канале.

После согласования многоканальности передающая система может отправлять PDU, инкапсулированные и/или фрагментированные с использованием заголовка multilink.

### 1.3. Соглашения

Ниже перечислены соглашения, применяемые в документе для указания уровня требований.

- **MUST (должно)**, **SHALL (нужно)** или **MANDATORY (обязательно)** — элемент является обязательным в соответствии с данной спецификацией.
- **SHOULD (следует)** или **RECOMMENDED (рекомендуется)** — элемент в общем случае нужен, но при необходимости может быть опущен.
- **MAY (можно)** или **OPTIONAL (необязательно)** — элемент является не обязательным и может применяться в соответствии с потребностями реализации.

## 2. Общий обзор

Для организации связи через канал «точка-точка» (point-to-point) каждая из сторон канала PPP должна сначала передать пакеты LCP для настройки канала данных в процессе организации соединения (Link Establishment). После того, как канал будет организован, PPP обеспечивает переход к фазе аутентификации (Authentication), в процессе которой могут применяться протоколы аутентификации для определения идентификаторов, связанных с каждой системой, подключенной к каналу.

Целью работы многоканальной системы является координация множества независимых каналов между фиксированной парой систем для организации виртуального соединения с более широкой пропускной способностью, нежели у любого из составляющих каналов. Агрегатный (композиционный) канал или связка (bundle) обозначается парой идентификаторов для двух систем, соединенных множеством каналов. Идентификатор системы может включать информацию, обеспечиваемую PPP Authentication [3] и процессом согласования LCP. Группируемые в связку каналы могут быть различными физическими каналами (например, асинхронные линии), но могут также являться экземплярами мультиплексируемых соединений (например, ISDN, X.25 или Frame Relay). Каналы могут иметь различные типы (например, синхронные коммутируемые и асинхронные выделенные линии в одной связке).

<sup>1</sup>Maximum receive unit.

<sup>2</sup>Protocol data unit.

Предполагается, что работу мультиканальной системы можно смоделировать как виртуальный объект канального уровня PPP, в котором пакеты, полученные по различным физическим каналам, идентифицируются как принадлежность отдельных сетевых протоколов PPP (Multilink Protocol или MP) и собираются в соответствии с информацией, представленной в заголовке фрагментации. Все пакеты, принятые через каналы, объединенные в связку, представляются одной машине обработки сетевого уровня, независимо от наличия заголовка multilink.

Пакеты, переданные с использованием многоканальных процедур, инкапсулируются в соответствии с правилами PPP. Перечисленные ниже опции должны быть заданы вручную.

- Нет отображения символов асинхронного управления.
- Не используется Magic Number.
- Не используется мониторинг качества канала (Link Quality Monitoring).
- Компрессия адресов и полей управления.
- Компрессия полей протокола.
- Композитные кадры не используются.
- Не используется заполнение с самоописанием (Self-Describing-Padding).

В соответствии с правилами RFC 1661 это означает, что реализация **должна** воспринимать собранные пакеты с ведущими нулями в поле протокола (Protocol) и без них. Хотя ниже явно запрещено включение полей Address и Control (обычно два байта FF 03) во фрагментируемую часть, хороший стиль программирования предполагает восприятие пакетов независимо от наличия этих байтов в соответствии со спецификацией RFC1661.

Как признак хорошей реализации предполагается, что в момент создания канала определяется наличие ведущих нулей путем проверки согласования PFC на первом канале, включенном в связку. Это следует сохранять в течение всего времени существования мультиканального пакета.

Естественно, для разных каналов могут быть согласованы различные варианты этих опций. Как будет показано в дальнейшем, каналы в мультиканальной связке **должны** согласовать самоописывающее заполнение (Self-Describing-Padding) даже в тех случаях, когда предварительно фрагментированные (pre-fragmented) пакеты **недопустимо** дополнять. Поскольку режим компрессии поля протокола (Protocol Field Compression) для каналов связи позволяет передающей системе включать или не включать ведущие нули по своему усмотрению, это является дополнительным механизмом генерации пакетов с четной длиной.

Согласование LCP не выполняется для самой многоканальной связки. Реализации **недопустимо** передавать пакеты LCP Configure-Request, Configure-Reject, Configure-Ack, Configure-Nak, Terminate-Request и Terminate-Ack через мультиканальную процедуру, а принимающая реализация **должна** отбрасывать такие пакеты без уведомления, не генерируя в ответ никаких пакетов PPP, но с возможностью протоколирования приема нежелательных пакетов. В отличие от этого, остальные пакеты LCP, имеющие функции управления, не связанные с изменением принятых по умолчанию параметров для пакета в целом, обрабатываются нормально. Реализация **может** передавать пакеты LCP Code-Reject, Protocol-Reject, Echo-Request, Echo-Reply и Discard-Request.

Эффективное значение MRU для элементов логического канала согласуется посредством опции LCP. Не имеет значения, инкапсулируются или нет пакеты NCP<sup>1</sup> в мультиканальные заголовки и даже не играет роли канал, через который эти пакеты передаются после того, как канал идентифицирует себя как часть многоканальной связки.

Отметим, что сетевые протоколы, которые передаются без использования мультиканальных заголовков, не могут быть последовательными (и, следовательно, будут доставляться любым удобным путем).

Рассмотрим в качестве примера рисунок 1. Link 1 имеет согласованные сетевые уровни NL 1, NL 2 и MP между двумя системами. Затем эти системы согласуют MP через канал Link 2.

#### Сетевой уровень

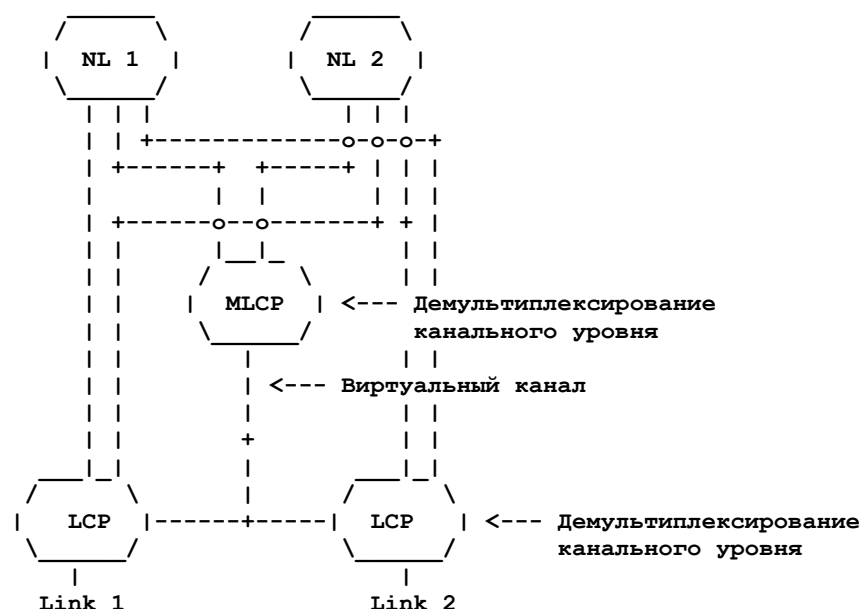


Рисунок 1.

<sup>1</sup>Network Control Protocol - протокол управления сетью.

Кадры, принятые по каналу 1, демультиплексируются на канальном уровне в соответствии с идентификатором протокола PPP и могут быть переданы NL 1, NL 2 или MP. Канал 2 будет принимать кадры со всеми сетевыми протоколами, которые принимает канал 1.

Кадры, принятые MP, подвергаются дальнейшему демультиплексированию на сетевом уровне в соответствии с идентификатором сетевого протокола PPP и передаются NL 1 или NL 2. Любые кадры, принятые MP для любого другого протокола сетевого уровня, отбрасываются с использованием обычного для протокола механизма.

### 3. Формат пакетов

В этом параграфе описана схема отдельных фрагментов, которые являются «пакетами» в MP. Пакеты сетевого протокола сначала инкапсулируются (но не кадрируются) по обычным процедурам PPP а большие пакеты делятся на сегменты, размеры которых приемлемы для множества физических каналов. Хотя спецификация PPP разрешает это, реализациям **недопустимо** включать поля Address и Control в объект, подлежащий фрагментации. Новый заголовок PPP, включающий идентификатор Multilink Protocol и заголовок Multilink, помещается перед каждым разделом (таким образом, первый фрагмент многоканального пакета в PPP будет иметь два заголовка — один для фрагмента, а следующий за ним для самого пакета).

Системы, реализующие многоканальную процедуру, не обязаны фрагментировать мелкие пакеты. Нет также требования соблюдать одинаковый размер сегментов и даже требования разбиения пакетов. Одной из возможных стратегий работы с каналами разной скорости является разбиение пакетов на сегменты, пропорционально скоростям каналов. Другая стратегия может состоять в разбиении на множество одинаковых по размеру фрагментов и распределением их по каналам пропорционально относительной скорости каждого из каналов.

При фрагментации и инкапсуляции PPP multilink используется идентификатор протокола 0x00-0x3d. Вслед за идентификатором протокола размещаются 4 байта с порядковым номером и двумя однобитовыми полями, указывающими первый или последний фрагмент пакета. После согласования дополнительной опции PPP LCP четыре байта заголовка могут быть заменены двумя байтами с использованием для номера лишь 12 битов. Предполагается, что для полей Address, Control и Protocol ID используется сжатие. Отдельный фрагмент будет иметь формат, показанный ниже.

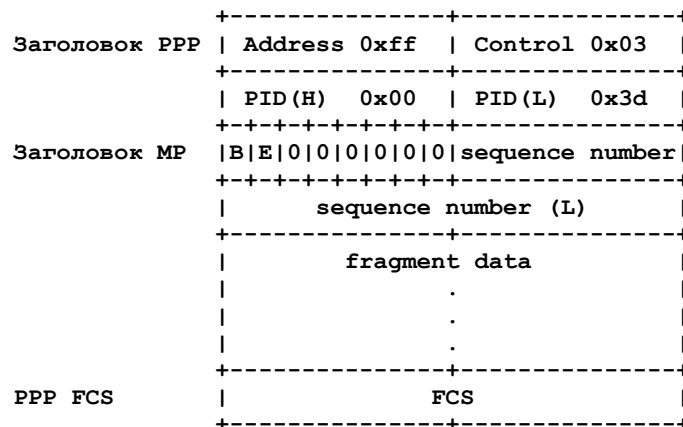


Рисунок 2. Формат фрагмента с длинным номером.

Флаг первого фрагмента (B) устанавливается (1) только для первого фрагмента, полученного для пакета PPP, а в остальных случаях имеет значение 0.

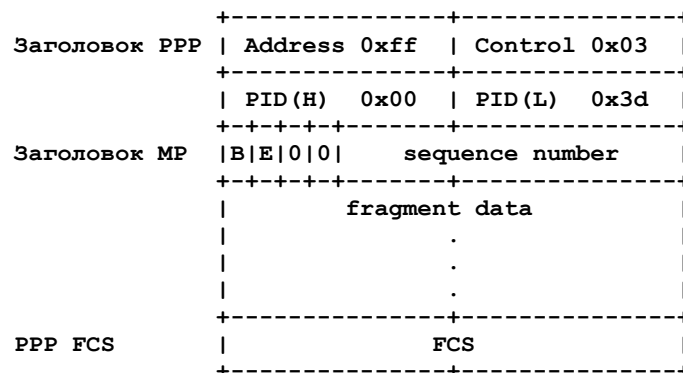


Рисунок 3. Формат фрагмента с коротким номером.

Флаг последнего фрагмента (E) устанавливается (1) только для последнего фрагмента, а в остальных случаях имеет значение 0. Во фрагменте могут быть установлены оба флага (B) и (E).

Поле порядкового номера (24 или 12 битов) увеличивается для каждого следующего фрагмента. По умолчанию применяются 24-битовые номера, но может быть согласовано использование лишь 12 с описанной ниже конфигурационной опцией LCP.

Между флагом последнего фрагмента (E) и порядковым номером размещаются резервные биты, которые в настоящее время не определены и **должны** иметь значение 0. Это 2 бита при использовании коротких номеров и 6 в остальных случаях.

В этом многоканальном протоколе со связкой каналов ассоциируется одна структура сборки, к контексте которой интерпретируются заголовки multilink.

Поле FCS, показанное на рисунках, наследуется от обычного механизма кадрирования включенного в связку канала, по которому будет передаваться пакет. Для восстановленного пакета не предусмотрено значения FCS, если этот пакет фрагментируется.

### 3.1. Заполнение

Системам с поддержкой многоканального протокола **следует** реализовать заполнение с самоописанием (SDP<sup>1</sup>). Система, реализующая SDP, будет по определению включать опцию заполнения в свои начальные сообщения LCP Configure-Request или (для предотвращения задержки Configure-Reject) включать опцию заполнения после получения NAK с опцией.

Система, которая должна дополнять свои передачи, но не использует SDP при отсутствии многоканальности, может по-прежнему не применять SDP, если она обеспечивает подходящий размер фрагментов и дополнять приходится лишь фрагменты с флагом (E). Системе **недопустимо** добавлять заполнение к какому-либо пакету, который не может быть распознан партнером как дополненный. Фрагменты, которые не являются первыми или последними, **недопустимо** заполнять с использованием какого-либо материала, кроме заполнения с самоописанием.

Система **должна** гарантировать, что SDP, как описано в RFC 1570 [11], согласовано на отдельном канале до начала передачи каких-либо многоканальных пакетов, если она может дополнять пакеты, не являющиеся крайними, или будет применять протоколы сжатия или сетевые протоколы, способные пострадать от заполнения, как описано в RFC 1570. При необходимости, система, которая применяет заполнение, **должна** использовать LCP Configure-NAK с целью выбора Configure-Request для SDP от партнера.

Отметим, что сообщения LCP Configure-Request могут передаваться в любой момент по любому из каналов и партнер всегда будет отвечать своим сообщением Configure-Request. Система, которая дополняет передачу, но не использует протоколов кроме многоканального, который может пострадать от заполнения, может задержать отправку партнеру сообщения Configure-Request SDP до того момента, когда ей представится желательным согласовать использование самого многоканального протокола. Это обеспечит взаимодействие систем, использующих заполнение, со старыми партнерами, которые не поддерживают ни Multilink, ни SDP.

## 4. Компромисс между размером буферов и потерями фрагментов

В многоканальной процедуре возможна задержка одного канала относительно других каналов связки. Это может приводить к нарушению порядка получения фрагментов, что усложняет детектирование потери фрагментов. Это усложняет задачу оценки размера буферов на приемной стороне. В этом разделе рассматривается метод детектирования потери фрагментов с учетом минимизации требуемого размера буферов без роста потерь.

### 4.1. Обнаружение потери фрагментов

На каждом канале связки отправитель **должен** передавать фрагменты строго в порядке роста номеров (с учетом модуля пространства номеров). Это требование поддерживает стратегию получателя по обнаружению потери фрагментов путем сравнения порядковых номеров. Порядковые номера не сбрасываются для каждого нового пакета PPP и применяются даже для тех фрагментов, которые содержат пакет PPP целиком (флаги (B) и (E) установлены).

Реализация **должна** указать для первого фрагмента, передаваемого через вновь созданную связку, номер 0. Добавление канала в существующую невидимо для протокола и сбрасывать в такой ситуации порядковый номер **недопустимо**.

Получатель отслеживает порядковые номера на каждом канале связки и сохраняет текущее минимальное значение среди всех каналов связки (обозначим его M). Получатель обнаруживает конец пакета по наличию у фрагмента флага (E). Сборка пакета завершается, если были получены его фрагменты со всеми порядковыми номерами.

Потеря фрагмента детектируется, если M превышает порядковый номер фрагмента с флагом (E), который не был включен в заверленную сборку (т. е. не все фрагменты между первым фрагментом (B) и последним фрагментом (E) были получены). Это обеспечивается правилом роста порядковых номеров в связке. Обнаружение отсутствующего номера говорит о потере фрагмента.

Реализация **должна** предполагать, что фрагменту с флагом (B) непосредственно предшествует по порядковым номерам фрагмент с флагом (E). Таким образом, если потерян пакет с флагом (E) и пакет с номером M имеет флаг (B), реализация **должна** отбросить все фрагменты несобранных пакетов до M-1, но **не следует** на этом основании фрагменты с установленным флагом (B).

Детектирование потерянного фрагмента, порядковый номер которого был определен как U, заставляет получателя отбросить все фрагменты вплоть до наименьшего номера с установленным битом (E), если этот номер не превышает U. Однако величина M может перейти в середину цепочки пакетов, которая может быть успешно завершена.

Фрагменты могут теряться в результате повреждения отдельных пакетов или аварийной потери канала (которая может произойти для одного направления). Данная версия многоканального протокола не требует специальных процедур для обнаружения отказавших каналов. Для достижения этой цели можно применять средства управления качеством канала PPP или периодически передавать эхо-запросы LCP.

Отправителям **следует** избегать простоя отдельных каналов для максимально быстрого определения потери фрагментов на приемной стороне, поскольку значение M не инкрементируется на бездействующих каналах incremented on idle links. Отправителям **следует** выполнять ротацию трафика между каналами, если трафика недостаточно для заполнения каналов с целью предотвращения бездействия.

Потеря финального фрагмента может «замораживать» получателя в ожидании прибытия нового пакета. Вероятность этого можно снизить передачей пустого фрагмента (null) в каждый канал связки, который стал бы бездействующим сразу после отправки фрагмента с флагом (E). Пустой фрагмент содержит лишь заголовок multilink с флагами (B) и (E), т. е. не содержит данных. Реализациям, озабоченным потерей пропускной способности или расходами в расчете на пакет, не требуется передавать пустые фрагменты и они могут отложить их отправку до завершения отсчета таймера с

<sup>1</sup>Self-Describing-Padding.

некоторой вероятностью увеличения в результате задержки на стороне получателя. Получателю **следует** реализовать тот или иной таймер бездействия канала для защиты от бесконечного ожидания.

Правило увеличения порядковых номеров предотвращает перераспределение фрагментов, поставленных в очередь для отказавшего канала, что не является необычным делом для реализаций ISO multilink на основе LAPB [4].

## 4.2. Требования к размеру буфера

Не существует какого-либо объема буферов, который гарантировал бы корректное обнаружение потери фрагментов, поскольку партнер на другой стороне может удерживать фрагмент на одном канале и отправить произвольное число фрагментов в другие каналы. Для обычного случая, когда передача идет по всем каналам, можно показать, что имеется минимальный объем буфера, ниже которого уже не будет возможности корректно детектировать потери. Этот объем зависит от относительных задержек между каналами ( $D[\text{channel-}i, \text{channel-}j]$ ), скорости в каждом канале  $R[c]$ , максимального размера фрагментов, разрешенного для каждого канала  $F[c]$  и общего объема буфером, распределяемого между каналами на передающей стороне.

При использовании PPP задержку между каналами можно определить с помощью запросов и откликов LCP-эхо (в случае разных скоростей в каналах это следует учитывать). «Проскальзывание» для каждого канала можно определить как пропускную способность, умноженную на задержку для этого канала по сравнению с каналом, где задержка максимальна,  $S[c] = R[c] * D[c, c\text{-worst}]$  ( $S[c\text{-worst}]$ , естественно, будет 0!).

Ситуация, которая будет усугублять нарушение порядка номеров, возникает при пиковой загрузке (почти все каналы полностью заняты), когда передающий узел помещает в очередь одного канала возможное число пронумерованных пакетов, затем делает это на втором канале и т. д. Поскольку передающий узел должен буферизовать по крайней мере один фрагмент максимального размера на каждом канале (обычно будет буферизовать не меньше двух), получатель, который выделил буферы меньше  $S[1] + S[2] + \dots + S[N] + F[1] + \dots + F[N]$ , будет сталкиваться с риском некорректного детектирования потери пакетов. Поэтому **следует** выделять буферы по крайней мере вдвое больше.

## 5. Расширения PPP LCP

Если нужна надежная работа через множество каналов, до начала использования протокола MP **должна** быть согласована надежная передача PPP Reliable Transmission [6] (по сути, ISO LAPB) на каждом канале связи.

Независимо от использования надежной доставки на каналах связи, реализация **должна** представлять сигналы о потерях протоколу NCP, работающему на связке каналов.

Компрессия может применяться отдельно на каждом канале или для связки (как логического группового канала). Использование множества сжатых потоков в связке (т. е. отдельно для каждого канала) указывается протоколом Compression Control [5], но с дополнительным идентификатором протокола PPP.

### 5.1. Типы конфигурационных опций

Протокол MP использует дополнительные опции конфигурации LCP:

- Multilink Maximum Received Reconstructed Unit (максимальный размер восстанавливаемого блока);
- Multilink Short Sequence Number Header Format (использование коротких порядковых номеров);
- Endpoint Discriminator (дискриминатор конечной точки)

#### 5.1.1. Опция LCP Multilink MRRU

```

      0           1           2           3
0 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|  Type = 17  | Length = 4  | Max-Receive-Reconstructed-Unit |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Рисунок 4. Опция LCP Multilink MRRU.

Присутствие этой опции LCP указывает, что передающая ее система поддерживает протокол PPP Multilink. Если опция не отвергнута, система будет считать все пакеты на этом канале, как пригодные для обработки общим протокольным модулем со всеми пакетами от того же партнера через любой другой канал, на котором была принята эта опция.

Поле Max-Receive-Reconstructed-Unit размером 2 октета задает максимальное число октетов в информационных полях восстановленных из фрагментов пакетов. Система **должна** быть способна принимать поле Information размером 1500 октетов во всех собранных пакетах PPP, хотя она **может** предпринять попытку согласования большего или меньшего значения. Значение 1500 взято из спецификации опции MRU LCP в PPP и при смене значения в будущих версиях RFC 1661, здесь будут применяться те же правила.

Система **должна** включать опцию LCP MRRU в каждом согласовании LCP, направленном на создание связки или присоединение к имеющейся. Если опция LCP MRRU предлагается на канале, предназначенном для добавления к имеющейся связке, система **должна** предлагать значение Max-Receive-Reconstruct-Unit, выбранное для этой связки.

Системе **недопустимо** передавать какие-либо multilink-пакеты в любой из каналов, пока ее партнер не предложил опцию MRRU LCP и не было подтверждения configure-Ack в последнем согласовании LCP на этом канале. Система **может** включить опцию MRRU LCP в сообщение configure-NAK, если партнер не предложил ее (если партнер в соответствии с правилами PPP не передал configure-Reject для нее).

**Примечание.** Значение MRRU, передаваемое в этой опции, соответствует MRU для связки, если рассматривать ее как объект PPP. Однако правила для опции Multilink MRRU отличаются от правил для LCP MRU, поскольку значение **должно** предлагаться при каждом согласовании LCP и подтверждение этой опции требуется до начала работы по множеству каналов.

### 5.1.2. Формат опции коротких порядковых номеров

```

      0                               1
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Type = 18   | Length = 2   |
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Рисунок 5. Опция короткого номера.

Эта опция сообщает партнеру о желании данной реализации получать фрагменты с короткими 12-битовыми порядковыми номерами. Когда партнер подтверждает эту опцию в configure-Ask он **должен** передавать все multilink-пакеты на всех каналах связи с 12-битовыми порядковыми номерами. В противном случае передается configure-Reject. Если желательно применять 12-битовые номера, эта опция **должна** быть согласована при организации связи и **должна** явно включаться в каждый последующий конфигурационный запрос LCP, когда система предполагает включить канал в связку, использующую 12-битовые номера. Если эта опция не согласовывалась в течение срока действия связи, будут использоваться 24-битовые порядковые номера.

Реализация, желающая передавать multilink-фрагменты с короткими порядковыми номерами, **может** включить опцию коротких номеров в configure-NAK для запроса у партнера встречного запроса на использование коротких номеров. Партнер не обязан отвечать такой опцией.

### 5.1.3. Опция дискриминатора конечной точки

```

      0                               1                               2                               3
      0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+
|   Type = 19   | Length      | Class      | Address ...
+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+---+

```

Рисунок 6. Опция дискриминатора конечной точки.

Опция дискриминатор конечной точки (Endpoint Discriminator) указывает систему, передающую пакет. Эта опция указывает системе, что партнер на этом канале может быть таким же, как партнер на имеющемся другом канале. Если опция отличает партнера от всех других, **должна** создаваться новая связка из согласуемого соединения. Если опция соответствует классу и адресу одного из партнеров на существующем канале, новый канал **должен** присоединяться к связке, содержащей канал с таким партнером, или **должна** создаваться новая связка, в зависимости от дерева решений, показанного ниже, - пп. (1) - (4).

Для защищенного добавления к имеющейся связке должен использоваться протокол аутентификации PPP [3] с целью получения от партнера подлинной информации и предотвращения включения канала в имеющуюся связку на основе фальсифицированной опции дискриминатора.

Эта опция не требуется для многоканальной работы. Если система не получала опции Multilink MRRU, но получила опцию Endpoint Discriminator при отсутствии заданной вручную конфигурации многоканальной работы, реализации **недопустимо** предполагать многоканальную работу на основе только этой опции.

По причине отсутствия требований к проверке подлинности, возможны 4 варианта, перечисленных ниже.

- (1) Нет аутентификации и дискриминатора.

Все новые каналы **должны** включаться в одну связку, если заданная вручную конфигурация не требует иного. Недопустимо также иметь более одной заданной вручную связки между парой систем.

- (2) Дискриминатор без проверки подлинности.

При соответствии дискриминатора канал **должен** добавляться к имеющейся связке, в противном случае **должна** создаваться новая.

- (3) Проверка подлинности без дискриминатора.

При подтверждении подлинности канал **должен** добавляться к имеющейся связке, в противном случае **должна** создаваться новая.

- (4) Дискриминатор и проверка подлинности.

При соответствии дискриминатора и подтверждении подлинности канал **должен** добавляться к имеющейся связке, в остальных случаях **должна** создаваться новая.

Опция содержит поле Class, которое определяет идентификатор адресного пространства, и поле Address, которое указывает уникальный идентификатор в этом пространстве.

Предполагается, что идентификатор указывает физическое устройство, связанное с передающей системой. Для некоторых классов идентификаторы уникальны в глобальном масштабе и не связаны с областью действия конкретного административного домена. В таких случаях уникальность адреса контролируется зависящими от класса правилами выделения значений.

Каждая конечная точка может выбирать идентификаторы класса без ограничений. Поскольку цель заключается в детектировании расхождений между конечными точками, ошибочно сочтенными однотипными, несоответствия класса вполне достаточно. Хотя ни один из классов не указан в качестве рекомендуемого, предпочтительно использовать классы с уникальными в глобальном масштабе адресами.

Поддержка этой опции не является обязательной для каждой системы или партнера. Если опция не указана в Configure-Request, системе **недопустимо** создавать Configure-Nak с этой опцией при любой причине отказа. Вместо этого **следует** вести себя так, будто была принята опция с Class = 0, Address = 0. Если система получает Configure-Nak или Configure-Reject с этой опцией, она **должна** удалить опцию из всех дополнительных сообщений Configure-Request.

Размер определяется полем Length. Для некоторых классов размер фиксирован, для остальных может меняться. Опция недействительна, если поле Length содержит значение, которое меньше минимума для этого класса.

Реализация **может** использовать Endpoint Discriminator для поиска административных или аутентификационных записей в локальной базе данных. Такое применение опции является второстепенным и не допускается, если взамен можно использовать протокол PPP Authentication [3]. Поскольку некоторые классы позволяют партнеру создавать случайный или локально администрируемый адрес, использование опции в качестве ключа базы данных требует предварительного согласования между администраторами.

Описания субполей приведены ниже

**Type**

19 = для Endpoint Discriminator

**Length**

3 + размер поля Address

**Class**

Поле Class имеет размер 1 октет и задает идентификатор пространства адресов. Наиболее свежие значения поля LCP Endpoint Discriminator Class field можно найти в свежей версии документа Assigned Numbers [7]. Текущие значения указаны ниже.

- 0 Null Class (пустой класс).
- 1 Locally Assigned Address (локально назначаемый адрес).
- 2 Адрес IP.
- 3 Глобально администрируемый адрес IEEE 802.1 MAC
- 4 Блок PPP Magic-Number
- 5 Телефонный номер в сети общего пользования

**Address**

Поле Address содержит 1 или несколько октетов, указывающих идентификатор адреса в выбранном классе. Размер и содержимое поля зависит от значения поля Class, как показано ниже.

Класс 0 — пустой класс

Максимальный размер: 0

Этот класс используется по умолчанию, если опции нет в принятом сообщении Configure-Request.

Класс 1 — локально назначаемый адрес.

Максимальный размер: 20

Этот класс определен для обеспечения возможности использования локальных адресов в тех случаях, когда нет подходящего глобального адреса. Предлагается использовать серийные номера устройств. Выбор класса типа адресов не рекомендуется, поскольку они не гарантируют однозначности.

Класс 2 — адрес IP

Фиксированный размер: 4

Адрес этого класса содержит IP-адрес хоста, как определено в [8].

Класс 3 - глобально администрируемый адрес IEEE 802.1 MAC

Фиксированный размер: 6

Адрес этого класса содержит 6 октетов IEEE 802.1 MAC в каноническом формате 802.3 [9]. Биты global/local и multicast/specific в адресе **должны** быть сброшены. Для локально назначенных MAC-адресов следует применять класс 1.

Класс 4 — блок PPP Magic-Number

Максимальный размер: 20

Это не адрес, а блок из 1 - 5 объединенных 32-битовых значений PPP Magic-Number [2]. Это класс служит для автоматической генерации значений, но их уникальность не гарантируется. Конечная точка **должна** применять один и тот же блок, пока хотя бы один канал находится в состоянии LCP Open. Выбор класса типа адресов не рекомендуется, поскольку они не гарантируют однозначности.

Отметим, что значения PPP Magic-Number используются в [2] обнаружения неожиданных соединений конечной точки с собой (loopback). Вероятность генерации двумя точками одинаковых magic-number невелика. Эта вероятность существенно снижается при повторении согласования LCP в поисках несоответствия если партнер может генерировать несогласованные значения magic-number.

Здесь значения magic-number используются для определения принадлежности двух каналов одной или разным конечным точкам. Исходящие от одной точки номера всегда будут совпадать. Вероятность совпадения номеров от разных конечных точек достаточно мала. Для уверенности в отсутствии ложных совпадений, как при обнаружении петель LCP, можно объединить в блок несколько несвязанных magic-number.

Класс 5 — номер в телефонной сети общего пользования

Максимальный размер: 15

Адрес этого класса содержит последовательность октетов, определяемую стандартом I.331 (E.164) и представляющую телефонный номер в международном формате, используемый для доступа к конечной точек через телефонную сеть общего пользования [10].



## 6. Инициирование использования мультиканальных заголовков

Когда использование протокола MP согласовано на канале Y и канал добавлен в связку, содержащую единственный канал X, система **должна** передать пакет с multilink-инкапсуляцией в канал X до начала передачи таких пакетов в Y.

Поскольку каналы могут добавляться в связку и удаляться из нее без нарушения состояния связки, фрагменту следует назначать подходящий (следующий) номер. Как отмечено выше, первый фрагмент передается через v с номером 0.

## 7. Закрытие каналов связки

Каналы связки могут разрываться с помощью обычных процедур PPP LCP с использованием пакетов LCP Terminate-Request и Terminate-Ack на таком канале. Поскольку предполагается, что каналы связки обычно не меняют порядок пакетов, прием подтверждения разрыва служит достаточным основанием для допущения от отсутствия потерь каких-либо пакетов мультиканального протокола.

Получение пакета LCP Terminate-Request на одном из каналов связки не препятствует работе оставшихся каналов.

Пока в связке остаются активные каналы, состояние PPP для связки сохраняется как отдельный объект. Однако если в связке остался единственный канал, а остальные были корректно закрыты (с Terminate-Ack), реализация **может** прекратить использование заголовков multilink.

Если многоканальная процедура применяется вместе с гарантированной доставкой PPP и канал связки не был закрыт корректно, реализации следует ожидать приема пакетов с нарушением правила роста порядковых номеров.

## 8. Взаимодействие с другими протоколами

В общем случае LCP и протокол управления аутентификацией (Authentication Control Protocol) должны быть согласованы для всех каналов пакета. Поля сетевого протокола (Network Protocol) сами по себе и связанные с ними управляющие расширения обычно задаются для пакета в целом.

В некоторых случаях для ряда сетевых протоколов (Network Protocol) существует требование последовательной передачи и, если размеры MRU для каналов не приводят к фрагментации, эти протоколы могут напрямую передаваться через отдельные каналы пакета.

Хотя это явно не одобрялось выше, при наличии нескольких каналов, соединяющих две реализации, и желательности использования независимой нумерации для двух наборов протоколов, если они не блокируют друг друга, можно описать две мультиканальных процедуры, предполагая множество идентификаторов конечных точек для данной системы. Каждый канал, однако, будет оставаться только в одном пакете. One could think of a physical router as housing two logically separate implementations, каждая из которых сконфигурирована независимо от другой.

Простейшим решением будет отказ от включения одного канала в пакет путем передачи пакета Configure-Reject в ответ на опцию Multilink LCP.

## 9. Вопросы безопасности

Работа этого протокола не влияет на обеспечение безопасности с помощью протоколов аутентификации PPP [3]. Для получения дополнительной информации вы можете воспользоваться приведенной ссылкой.

## 10. Литература

- [1] Leifer, D., Sheldon, S., and B. Gorsline, "A Subnetwork Control Protocol for ISDN Circuit-Switching", University of Michigan (unpublished), March 1991.
- [2] Simpson, W., Editor, "The Point-to-Point Protocol (PPP)", STD 51, [RFC 1661](#), Daydreamer, July 1994.
- [3] Lloyd, B., and W. Simpson, "PPP Authentication Protocols", RFC 1334, Lloyd Internetworking, Daydreamer, October 1992.
- [4] International Organisation for Standardization, "HDLC - Description of the X.25 LAPB-Compatible DTE Data Link Procedures", International Standard 7776, 1988
- [5] Rand, D., "The PPP Compression Control Protocol (CCP)", PPP Extensions Working Group, RFC 1962, June 1996.
- [6] Rand, D., "PPP Reliable Transmission", RFC 1663, Novell, July 1994
- [7] Reynolds, J., and J. Postel, "Assigned Numbers", STD 2, RFC 1700<sup>1</sup>, USC/Information Sciences Institute, October 1994.
- [8] Postel, J., Editor, "Internet Protocol - DARPA Internet Program Protocol Specification", STD 5, RFC 791, USC/Information Sciences Institute, September 1981.
- [9] Institute of Electrical and Electronics Engineers, Inc., "IEEE Local and Metropolitan Area Networks: Overview and Architecture", IEEE Std. 802-1990, 1990.
- [10] The International Telegraph and Telephone Consultative Committee (CCITT), "Numbering Plan for the ISDN Area", Recommendation I.331 (E.164), 1988.
- [11] Simpson, W., Editor, "PPP LCP Extensions", RFC 1570, Daydreamer, January 1994.

## 11. Отличия от RFC 1717

В этом разделе рассмотрены различия между данным документом и [RFC 1717](#). Существуют ограничения, связанные с реализациями, которые отсутствуют в RFC 1717. Системы, следующие этим ограничениям, будут полностью совместимы с системами RFC 1717.

<sup>1</sup>В соответствии с [RFC 3232](#) данный документ утратил силу. Информация доступна по [ссылке](#). *Прим. перев.*

### 11.1. Согласование параметров мультиканала

RFC 1717 допускает использование отдельного формата с укороченными порядковыми номерами SSNHF (Short Sequence Number Header Format) или опций MRRU (Maximum Reconstructed Receive Unit - максимальный размер восстанавливаемого принятого блока) для индикации намерения согласовать мультиканал. Настоящая спецификация запрещает использование опций SSNHF самих по себе, но позволяет использовать обе упомянутые опции вместе. Все реализации, которые обеспечивают совместимость с RFC 1717 по остальным требованиям и следуют приведенному ограничению, будут совместимы с реализациями на основе RFC 1717.

### 11.2. Определение стартового порядкового номера

Данная спецификация требует, чтобы первый порядковый номер пакета, переданного после того, как виртуальный канал достигнет открытого состояния, был равен 0.

### 11.3. Принятое по умолчанию значение MRRU

В этой спецификации отсутствует принятое по умолчанию значение MRRU (оно всегда должно согласовываться с некоторыми другими значениями) и указано, что реализации должны поддерживать MRRU с такими же значениями, как принятый по умолчанию размер MRU для протокола PPP.

### 11.4. Запрет Config-Nak для EID

Данная спецификация запрещает использование config-Nak для EID во всех случаях.

### 11.5. Однородность порядковых номеров

Данная спецификация требует использования общей нумерации для всех каналов мультиканального пакета.

### 11.6. Начало и прекращение использования мультиканальных заголовков

Данный документ указывает как должно начинаться использование заголовков Multilink при добавлении канала и при каких условиях можно безопасно прекращать их использование.

### 11.7. ручная настройка и добавление в связку

Документ явно разрешает ручную настройку множества каналов в отсутствие дискриминатора оконечной точки (Endpoint Discriminator) и любой формы аутентификации.

## 12. Адреса авторов

#### Keith Sklower

Computer Science Department  
384 Soda Hall, Mail Stop 1776  
University of California  
Berkeley, CA 94720-1776  
Телефон: (510) 642-9587  
EMail: [sklower@CS.Berkeley.EDU](mailto:sklower@CS.Berkeley.EDU)

#### Brian Lloyd

Lloyd Internetworking  
3031 Alhambra Drive  
Cameron Park, CA 95682  
Телефон: (916) 676-1147  
EMail: [brian@lloyd.com](mailto:brian@lloyd.com)

#### Glenn McGregor

Lloyd Internetworking  
3031 Alhambra Drive  
Cameron Park, CA 95682  
Телефон: (916) 676-1147  
EMail: [glenn@lloyd.com](mailto:glenn@lloyd.com)

#### Dave Carr

Newbridge Networks Corporation  
600 March Road

P.O. Box 13600

Kanata, Ontario,

Canada, K2K 2E6

Телефон: (613) 591-3600

E-Mail: [dcarr@Newbridge.COM](mailto:dcarr@Newbridge.COM)

**Tom Coradetti**

Sidewalk Software

1190 Josephine Road

Roseville, MN 55113

Телефон: (612) 490 7856

E-Mail: [70761.1664@compuserve.com](mailto:70761.1664@compuserve.com)

**Перевод на русский язык**

Николай Малых

[nmalykh@gmail.com](mailto:nmalykh@gmail.com)