

## Packet Reordering Metrics

Показатели разупорядочения пакетов

### Статус документа

В этом документе содержится спецификация стандарта для протокола, предложенного сообществу Internet. Документ служит приглашением к дискуссии в целях развития и совершенствования протокола. Текущее состояние стандартизации протокола вы можете узнать из документа Internet Official Protocol Standards (STD 1). Документ может распространяться без ограничений.

### Авторские права

Copyright (C) The IETF Trust (2006).

### Аннотация

Этот документ определяет показатели для оценки нарушения порядка доставки пакетов. Приводятся мотивы введения новых показателей и обсуждаются вопросы измерения, включая данные о контексте, требуемые для измерений. Сначала определяется одиночная (singleton) фиксация нарушения порядка, которая затем служит основой для выборки количественных значений степени разупорядочения в нескольких измерениях для характеристики сети или устройства получателя. Дополнительные показатели количественно оценивают частоту нарушения порядка и «расстояние» между отдельными нарушениями. Затем определяется метрика, ориентированная на влияние нарушения порядка на протокол TCP. Приведено несколько примеров оценки с использованием разных показателей. В приложении даны расширенные определения для оценки разупорядоченности при наличии фрагментов.

## Оглавление

1. Введение.....	2
1.1. Мотивация.....	2
1.2. Цели и задачи.....	3
1.3. Контекст, требуемый для всех показателей.....	3
2. Используемые соглашения.....	3
3. Показатель одиночного нарушения порядка.....	3
3.1. Название показателя.....	4
3.2. Параметры показателя.....	4
3.3. Определение.....	4
3.4. Определение пропуска номеров.....	4
3.5. Оценка нарушения порядка во времени или байтах.....	5
3.6. Обсуждение.....	5
4. Показатели выборки.....	5
4.1. Доля разупорядоченных пакетов.....	5
4.1.1. Имя показателя.....	5
4.1.2. Параметры показателя.....	5
4.1.3. Определение.....	5
4.1.4. Обсуждение.....	6
4.2. Степень нарушения порядка.....	6
4.2.1. Имя показателя.....	6
4.2.2. Обозначение и параметры показателя.....	6
4.2.3. Определение.....	6
4.2.4. Обсуждение.....	6
4.3. Разупорядочение по времени задержки.....	6
4.3.1. Имя показателя.....	7
4.3.2. Параметры показателя.....	7
4.3.3. Определение.....	7
4.3.4. Обсуждение.....	7
4.4. Разупорядочение по смещению байтов.....	7
4.4.1. Имя показателя.....	7
4.4.2. Параметры показателя.....	7
4.4.3. Определение.....	7
4.4.4. Обсуждение.....	8
4.5. Разрыв в нарушении порядка.....	8
4.5.1. Имена показателей.....	8

4.5.2. Параметры.....	8
4.5.3. Определение разрыва разупорядоченности.....	8
4.5.4. Определение интервала между нарушениями порядка.....	8
4.5.5. Обсуждение.....	8
4.6. Работа без нарушения порядка.....	9
4.6.1. Имена показателей.....	9
4.6.2. Параметры.....	9
4.6.3. Определение.....	9
4.6.4. Обсуждение и иллюстрация.....	9
5. Показатель, нацеленный на оценку получателя - TCP.....	10
5.1. Имя показателя.....	10
5.2. Обозначение параметров.....	10
5.3. Определения.....	10
5.4. Обсуждение.....	10
6. Вопросы измерений и реализации.....	11
6.1. Пассивные измерения.....	12
7. Примеры оценки порядка прибытия.....	12
7.1. Один пакет с нарушением порядка.....	12
7.2. Нарушение порядка двумя пакетами.....	12
7.3. Пример с тремя разупорядоченными пакетами.....	13
7.4. Нарушение порядка множеством пакетов.....	14
8. Вопросы безопасности.....	14
8.1. Отказ в обслуживании (DoS).....	14
8.2. Конфиденциальность данных пользователей.....	14
8.3. Влияние на показатели.....	14
9. Взаимодействие с IANA.....	14
10. Нормативные документы.....	16
11. Дополнительная литература.....	16
12. Благодарности.....	16
Приложение А. Пример реализации на языке С (информативный).....	17
Приложение В. Оценка порядка фрагментов (информативное).....	18
В.1. Имя показателя.....	18
В.2. Дополнительные параметры показателей.....	18
В.3. Определение.....	18
В.4. Замечания к показателям выборки при фрагментации.....	19
Приложение С. Отказ от ответственности и лицензия.....	19

## 1. Введение

Упорядоченное прибытие - это свойство, обнаруживаемое для пакетов, проходящих по своему пути, когда порядковый номер в каждом прибывающем пакете возрастает по сравнению с предшественником. В протоколах Internet Protocol (IP) [RFC791] [RFC2460] нет механизма упорядоченной доставки и протоколам вышележащих уровней (над IP) следует быть готовыми к обработке потерь и нарушениям порядка доставки. В этом документе определяются показатели разупорядочения пакетов.

Уникальный порядковый идентификатор в каждом пакете, например, инкрементируемый целочисленный номер сообщения, задаёт исходный порядок пакетов. Обнаружение разупорядочения на приёмной стороне основано на сравнении порядка поступления пакетов с их исходной нумерацией [Cia03].

Этот показатель согласуется с [RFC2330] и классифицирует полученные пакеты с номером меньше полученного ранее как разупорядоченные (out-of-order). Например, если пакеты поступают в порядке 1,2,4,5,3, пакет 3 будет разупорядоченным. Это эквивалентно определению разупорядочения Рахон [Рах98], где «запоздавшие» пакеты объявляются разупорядоченными. Другим вариантом является выделение «преждевременных» пакетов (4 и 5 в пример), но это можно отличить от потери пакета лишь по прибытии пакета 3. Фокусировка на просроченных пакетах позволяет обеспечить ортогональность (независимость) от показателей потери пакетов. Показатель очень похож на проверку пространства последовательностей принятых сегментов в [RFC793]. Более ранние работы по определению упорядоченной доставки включают [Cia00], [Ben99], [Lou01], [Bel02], [Jai02], [Cia03].

### 1.1. Мотивация

Показатели разупорядоченности актуальны для большинства приложений, особенно при оценке сетевой поддержки медиапотоков в реальном масштабе времени. Степень разупорядочения может вызывать отбрасывание пакетов функциями, расположенными выше уровня IP.

Порядок пакетом может меняться в процессе передачи и некоторые характеристики пути могут повышать вероятность нарушения порядка.

- При наличии двух (или более) путей со слабо различающимся временем передачи потока пакетов проходящие по разным путям пакеты могут приходиться с нарушением порядка. Множество путей может применяться для балансирования нагрузки или возникать в результате нестабильности маршрутов.
- Наличие в сетевом устройстве нескольких процессоров, обслуживающих один порт (или параллельные каналы), может приводить к нарушению порядка доставки.
- Протокол повтора передачи L2 для компенсации ошибок может нарушать порядок доставки.
- Обслуживание буферизованных пакетов не в порядке их поступления может нарушать порядок доставки.
- Если пакеты потока распределяются по разным буферам (например, по результатам оценки характеристик трафика) с различающимися характеристиками занятости и скорости обслуживания меняет порядок доставки.

Когда одна или несколько указанных характеристик путей присутствуют постоянно, разупорядочение также может сохраняться постоянно. Установившееся разупорядочение обычно вызывает нарушение порядка для существенной части пакетов. Эту форму разупорядочения проще всего обнаружить, минимизируя интервалы (spacing) между тестовыми пакетами. Постоянное разупорядочение может возникать в результате нестабильности сети - временные петли в маршрутизации могут приводить к сильному нарушению порядка. Это состояние характеризуется долгими периодами упорядоченной доставки с возникающим время от времени нарушением порядка иногда с экстремальной корреляцией. Однако не ожидается доставки пакетов в полностью случайном порядке, когда, например, последний или первый пакет в выборке с одинаковой вероятностью могут прийти первым к получателю. Таким образом, ожидается по меньшей мере минимальная степень упорядоченности доставки пакетов, как это происходит в реальных сетях.

Возможности восстановить порядок у получателя, скорее всего, будут ограничены. Практические хосты имеют приёмные буферы с конечным размером (по числу пакетов, байтов или продолжительности), например, для компенсации вариаций задержки. После первоначального обнаружения разупорядоченности полезно количественно оценить степень нарушения или задержки во всех значимых измерениях.

## 1.2. Цели и задачи

Приведённые ниже определения предназначены для достижения указанных ниже целей.

1. Обнаружение наличия или отсутствия нарушения порядка пакетов.
2. Количественная оценка степени разупорядоченности (Для этого в документе определено несколько параметров, поскольку в протоколах и приложениях применяются разные процедуры приёма. Так как влияние разупорядочения зависит от этих процедур, параметр количественной оценки ключевых аспектов поведения одного получателя может не подходить для других получателей. Если в отчёте указаны все показатели, определённые ниже, это даст широкое представление об условиях нарушения порядка.).

Показатели нарушения порядка **должны**:

- иметь одно или несколько применений, таких как разработка получателей или характеристика сети, а также убедительную актуальность с точки зрения заинтересованного сообщества;
- быть вычисляемыми «на лету»;
- работать даже при дублировании или потере пакетов из потока.

Желательно, чтобы показатели разупорядоченности имели хотя бы один из приведённых ниже атрибутов:

- способность объединять результаты из разных сегментов пути для оценки нарушения порядка на всем пути;
- простота расчётов и понимания;
- соответствие устройству протокола TCP;
- соответствие требованиям производительности в реальном масштабе времени.

Текущий набор показателей удовлетворяет всем приведённым выше требованиям и обеспечивает все атрибуты кроме конкатенации (за исключением случаев, когда измерения на сегментах показывают отсутствие нарушений порядка и можно считать, что полный путь из этих сегментов также не вносит нарушений). Однако выполнение требований ограничивает набор показателей теми, которые дают чёткое представление о характеристиках сети и устройстве получателя. Это вряд ли будет исчерпывающим набором для охвата влияния нарушения порядка на приложения и могут быть найдены дополнительные измерения.

## 1.3. Контекст, требуемый для всех показателей

Важнейшим аспектом всех показателей является их неразрывная связь с условиями измерений. Нарушение порядка пакетов невозможно надёжно определить, пока не указан точный контекст измерения. Поэтому все определения показателей нарушения порядка включают указанные ниже параметры.

1. Идентификаторы Packet of Type-P [RFC2330] для потока пакетов, включая транспортные адреса источника и получателя, а также любые другие данные, которые могут влиять на различную трактовку пакетов.
2. Набор параметров потока для дисциплины отправки, таких как уникальные параметры периодических потоков (как в [RFC3432]), потоков в стиле TCP (как в [RFC3148]) или потоков Пуассона (как в [RFC2330]). Параметры потока включают размер пакетов, заданный фиксированным значением или картиной (pattern) размеров (что из них применимо).

При указании показателя в отчёте **должно** включаться описание этих параметров для обеспечения контекста.

## 2. Используемые соглашения

Ключевые слова **необходимо** (MUST), **недопустимо** (MUST NOT), **требуется** (REQUIRED), **нужно** (SHALL), **не нужно** (SHALL NOT), **следует** (SHOULD), **не следует** (SHOULD NOT), **рекомендуется** (RECOMMENDED), **возможно** (MAY), **необязательно** (OPTIONAL) в данном документе должны интерпретироваться в соответствии с [RFC2119]. Хотя RFC 2119 был написан для протоколов, ключевые слова в этом документе используются так же. Они служат для обеспечения сопоставимости результатов измерений двух разных реализаций и выявления случаев, когда реализация может нарушать работу сети.

В этом документе символы  $\leq$  означают «меньше или равно», а  $\geq$  - «больше или равно».

## 3. Показатель одиночного нарушения порядка

Модель IPPM [RFC2330] выделяет одиночные измерения (singleton), выборки (sample) и статистику (statistics).

Одноэлементной (singleton) метрикой называются показатели, которые по сути неделимы (атомарны). Например, единичный экземпляр «общей пропускной способности» от одного хоста к другому можно определить как singleton-метрику, даже несмотря на то, что измерение включает множество пакетов Internet.

Для оценки порядка пакетов требуются некоторые концепции поддержки. Первая связана с алгоритмом (функцией), создающей серию строго монотонно возрастающих идентификаторов, применяемых для пакетов у отправителя для организации однозначного порядка передачи (функция  $g(x)$  возрастает строго монотонно, если для любого  $x > y$  выполняется условие  $g(x) > g(y)$ ). Уникальный последовательный идентификатор может быть просто возрастающим по порядку целым числом или порядковым номером, как указано ниже. Процедура перехода порядкового номера через максимум описана в разделе 6. Вопросы измерений и реализации.

Второй концепцией поддержки является сохраняемое значение «ожидаемого следующим» порядкового номера пакета. При обычных условиях значение NextExp является порядковым номером предыдущего пакета, увеличенным на 1 для нумерации сообщений (обычно получатель воспроизводит алгоритм отправителя и последовательность номеров, поэтому можно определить «следующий ожидаемый номер»).

Каждый пакет в потоке можно оценить с помощью этого показателя одиночных измерений (singleton metric).

### 3.1. Название показателя

Type-P-Reordered

### 3.2. Параметры показателя

- Src - IP-адрес хоста.
- Dst - IP-адрес хоста.
- SrcTime - время отправки пакета источником (или время в линии - wire time).
- s - уникальный порядковый номер пакета, заданный на стороне источника (в единицах сообщений).
- NextExp - следующий ожидаемый получателем порядковый номер (в единицах сообщений). Сохраняемое значение NextExp определяется из предыдущего принятого пакета.

Необязательные параметры

- PayloadSize - число байтов в информационном поле, которое указывают, когда значение SrcByte основано на числе переданных байтов.
- SrcByte - порядковый номер пакета, основанный на количестве переданных в пакетах данных.

### 3.3. Определение

Если обнаружено, что пакет s (передан в момент SrcTime) нарушает порядок при сравнении со значением NextExp, для него устанавливается Type-P-Reordered = TRUE. В ином случае Type-P-Reordered = FALSE, как определено ниже.

Type-P-Reordered принимает значение TRUE, если  $s < \text{NextExp}$  (пакет нарушает порядок). Значение NextExp в этом случае не меняется.

Type-P-Reordered принимает значение FALSE, если  $s \geq \text{NextExp}$  (пакет не ). В этом случае устанавливается  $\text{NextExp} = \text{set to } s+1$  для сравнения со следующим пакетом.

Поскольку NextExp не может уменьшаться, это обеспечивает критерий для обнаружения нарушающих порядок пакетов.

Приведённое определение можно записать в псевдокоде на момент получения пакета с порядковым номером s.

```
if s >= NextExp then /* s сохраняет порядок */
    NextExp = s + 1;
    Type-P-Reordered = False;
else /* когда s < NextExp */
    Type-P-Reordered = True
```

### 3.4. Определение пропуска номеров

Пакеты с  $s > \text{NextExp}$  являются особым случаем упорядоченной доставки. Это условие показывает разрыв последовательности из-за потери пакетов или нарушения их порядка. Чтобы счесть разрыв в номерах прерыванием разупорядочения должны быть получены пакеты с нарушением порядка (4. Показатели выборки).

Определим два случая для упорядоченных пакетов.

При  $s = \text{NextExp}$  исходная нумерация сохраняется и пропуска номеров нет.

Значение  $s > \text{NextExp}$  говорит, что некоторые пакеты исходной последовательности ещё не прибыли и в нумерации оказываются пропуск, связанный с пакетом s. Пропущенные номера  $s - \text{NextExp}$  указывают число пакетов, отсутствующих в результате потери или нарушения порядка.

Приведённое определение можно записать в псевдокоде на момент получения пакета с порядковым номером s.

```
if s >= NextExp, then /* s сохраняет порядок */
    if s > NextExp then
        SequenceDiscontinuity = True;
        SeqDiscontinuitySize = s - NextExp;
    else
        SequenceDiscontinuity = False;
        NextExp = s + 1;
        Type-P-Reordered = False;
else /* когда s < NextExp */
    Type-P-Reordered = True;
```

`SequenceDiscontinuity = False;`

Наличие разрыва в номерах (и его размер) определяется условиями, вызывающими потерю или нарушение порядка на пути измерений. Отметим, что переупорядочение пакетов может возникать в одной точке и утрачиваться (восстановление порядка) на последующем пути, но наблюдатель в точке приёма пакетов не будет знать об этом.

### 3.5. Оценка нарушения порядка во времени или байтах

Можно воспользоваться другими измерениями (время, переданные данные - payload) в определении переупорядочения в параграфе 3.3, если значение SrcTime и SrcBytes уникальны и надежны. Разрывы в нумерации легко определить и обнаружить путём нумерации сообщений, однако это усложняется при использовании времени или байтов. Это препятствует использованию упомянутых измерений, поскольку обнаружение пропусков в нумерации играет ключевую роль для описанных ниже показателей выборки.

Можно обнаружить пропуски в номерах с помощью нумерации байтов данных (payload), но это подходит лишь для случаев, когда тестовое устройство точно знает, какое значение присвоит NextExp при получении пакета. Это возможно при наличии у получателя полной картины размеров данных или при создании этой картины с использованием генератора псевдослучайных чисел и общей «затравки» (seed). Если размер данных не меняется, нумерация байтов ведёт к неоправданному усложнению по сравнению с нумерацией сообщений.

Можно обнаружить пропуски в периодических потоках с «нумерацией» по времени отправки, но здесь возникают практические сложности с точностью планирования передачи и синхронизацией часов.

Время и число байтов остаются важной основой для характеристики уровня переупорядочения, как описано в параграфах 4.3. Разупорядочение по времени задержки и 4.4. Разупорядочение по смещению байтов.

### 3.6. Обсуждение

Любой прибывающий пакет из последовательности, устанавливающей NextExp, можно проверить на предмет нарушения порядка. Если значение NextExp не определено (Undefined), поскольку прибывший пакет является первым в последовательности, пакет считается упорядоченным (Type-P-Reordered=FALSE).

Этот показатель позволяет собрать пакет из фрагментов до проверки упорядоченности. В принципе, можно применять показатель Type-P-Reordered для оценки порядка доставки фрагментов, но тогда в каждом фрагменте нужны сведения о порядке. Более подробно это рассматривается ниже - Приложение В. Оценка порядка фрагментов (информативное)

При получении дубликатов (несколько неповреждённых копий) они должны помечаться и для последующего анализа выбирается лишь первый пакет (копии считаются разупорядоченными в соответствии с данным выше определением). Это оказывает такое же влияние на хранение, как более ранние показатели IPPM, и соответствует прецеденту [RFC2679]. В разделе 6. Вопросы измерений и реализации предлагается минимизировать размер хранилища.

## 4. Показатели выборки

В этом разделе определяются показатели, применимые к выборкам пакетов из одной последовательности нумерации. При возникновении разупорядочения крайне желательно установить его уровень, т. е. отношение числа разупорядоченных пакетов к числу остальных. Здесь определяется несколько показателей для количественной оценки степени нарушения порядка, для каждого из которых имеется своё применение.

Параметры в последующих параграфах ориентированы на характеристики сети, но относятся также к устройству получателя, где интересно восстановление порядка. Сначала рассматривается просто доля разупорядоченных пакетов в выборке.

### 4.1. Доля разупорядоченных пакетов

#### 4.1.1. Имя показателя

Type-P-Reordered-Ratio-Stream

#### 4.1.2. Параметры показателя

Набор параметров включает одиночные параметры Type-P-Reordered - уникальные параметры пуассоновских потоков (как в [RFC2330]), периодических потоков (как в [RFC3432]) или потоков в стиле TCP (как в [RFC3148]), размер пакетов или картину размеров, а также:

- T0 - время старта;
- Tf - время завершения;
- dT - время ожидания прибытия для каждого пакета (в секундах);
- K - общее число пакетов в потоке, переданном от источника к получателю;
- L - общее число принятых пакетов (прибывших в интервале T0 - Tf+dT) из числа K переданных; идентичные копии - дубликаты удаляются, поэтому L <= K;
- R - отношение числа разупорядоченных пакетов к числу принятых, как определено ниже.

Отметим, что параметр dT фактически является порогом для объявления пакета потерянному. Показатель IPPM Packet Loss Metric [RFC2680] отвергает рекомендацию значения этого порога, заявляя: «на практике требуется понимать сроки существования пакетов».

#### 4.1.3. Определение

С учётом потока пакетов от источника к получателю доля переупорядоченных потоков в выборке составит

$$R = (\text{число пакетов с Type-P-Reordered=TRUE}) / (L)$$

Эту долю можно выразить в % (умножив на 100). Отметим, что в случае дубликатов учитывается лишь первая копия.

#### 4.1.4. Обсуждение

Когда `Type-P-Reordered-Ratio-Stream = 0`, для этой выборки не требуется других показателей разупорядочения. Поэтому ценность этого показателя заключается в простой возможности суммировать результаты для выборок без нарушения порядка.

## 4.2. Степень нарушения порядка

В этом разделе определяется степень нарушения порядка пакетов и связывается конкретное прерывание порядка с каждым разупорядоченным пакетом. Раздел наследует определённые выше параметры.

### 4.2.1. Имя показателя

`Type-P-Packet-Reordering-Extent-Stream`

### 4.2.2. Обозначение и параметры показателя

Напомним, что  $K$  - число пакетов, переданных источником, а  $L$  - число принятых получателем пакетов. Каждому пакету назначается порядковый номер  $s$  (от 1 до  $K$ ) в порядке отправки пакетов ( $u$  источника).

Пусть  $s[1], s[2], \dots, s[L]$  - исходные порядковые номера, связанные с пакетами в порядке их прибытия.  $s[i]$  можно считать вектором, где индекс  $i$  - номер прибытия пакета с порядковым номером  $s$ . Теоретически любой порядковый номер от источника может быть связан с любым номером прибытия, но это маловероятно.

Рассмотрим неупорядоченный пакет (`Type-P-Reordered=TRUE`) с индексом прибытия  $i$  и номером отправки  $s[i]$ . Имеется набор индексов  $j$  ( $1 \leq j < i$ ), таких, что  $s[j] > s[i]$ .

Новые параметры включают:

- $i$  - номер прибытия пакета, где  $i-1$  представляет пакет, прибывший перед  $i$ ;
- $j$  - набор из одного или нескольких индексов прибытия, где  $1 \leq j < i$ ;
- $s[i]$  - исходные порядковые номера ( $s$ ) в порядке отправки;
- $e$  - степень разупорядочения (`Reordering Extent`), указанную числом пакетов, как определено ниже.

### 4.2.3. Определение

Степень разупорядочения  $e$  для пакета  $s[i]$  определяется как  $i-j$  для наименьшего значения  $j$ , где  $s[j] > s[i]$ .

Неформально степень разупорядочения определяет максимальную дистанцию (в пакетах) от неупорядоченного пакета до ближайшего принятого пакета с большим порядковым номером. Если пакет получен по порядку, разупорядочение будет неопределённым. Первый пакет является упорядоченным по определению и имеет неопределённую степень разупорядочения.

Комментарий к определению. Для некоторых случаев порядка прибытия назначение простой позиции (дистанции) степени разупорядоченности ведёт к переоценке размера хранилища у получателя, требуемого для восстановления порядка. Более точной и сложной процедурой расчёта размера хранилища для восстановления порядка было бы вычитание (исключение) принятых ранее пакетов, которые получатель мог бы передать на вышележащий уровень (4.4. Разупорядочение по смещению байтов). С учётом смещения (*bias*) это определение считается достаточным особенно для случаев, когда нужны расчёты «на лету».

### 4.2.4. Обсуждение

Пакет с индексом  $j$  ( $s[j]$  в параграфе 4.2.3. Определение) представляет собой прерывание разупорядочения, связанное с пакетом  $s$  с индексом  $i$  ( $s[i]$ ). Определение формализовано ниже.

Отметим, что  $K$  в потоке могут быть частью большего потока, но  $L$  по-прежнему остаётся суммарным числом пакетов, полученных из числа  $K$  переданных пакетов.

Если получателю нужно восстановить порядок, его возможности буферизации определяют способность обработать пакеты с нарушением порядка. Для ситуаций с одним разупорядоченным пакетом, степень разупорядоченности  $e$  указывает число пакетов, которые нужно сохранить в буфере получателя в ожидании восстановления порядка. В более сложных случаях степень разупорядоченности может приводить к переоценке требуемого для буферизации пространства (см. 4.4. Разупорядочение по смещению байтов и примеры из раздел 7. Примеры оценки порядка прибытия). Если же получатель по какой-либо причине очищает буфер, степень разупорядочения не будет определять поведение, предполагая взамен, что получатель предпримет исчерпывающую попытку восстановить порядок.

Хотя степень разупорядоченности в первую очередь количественно определяет смещение по числу позиций прибытия, это может быть полезно для определения части разупорядоченных пакетов, для которых можно или невозможно восстановить порядок в типовом приёмном буфере на основе лишь порядка поступления (без помощи ретрансляции).

Степень разупорядоченности выборок можно отобразить на гистограмме для представления частоты нарушений.

## 4.3. Разупорядочение по времени задержки

Разупорядоченный пакета можно назначить смещения, указывающие их задержки с точки зрения времени в буфере, которое получатель должен обеспечить для восстановления порядка. Показатель смещения рассчитывается лишь для разупорядоченных пакетов, указанных одиночным показателем нарушения порядка (3. Показатель одиночного нарушения порядка).

### 4.3.1. Имя показателя

Type-P-Packet-Late-Time-Stream

### 4.3.2. Параметры показателя

В дополнение к параметрам, определенным для Type-P-Reordered-Ratio-Stream, задаются:

- DstTime - время прихода к получателю каждого пакета из потока, который можно связать с индексом  $i$  или пакетом  $s[i]$ ;
- LateTime( $s[i]$ ) - смещение пакета  $s[i]$  (в секундах), как определено ниже.

### 4.3.3. Определение

Время задержки рассчитывается по часам получателя. Когда полученный пакет  $s[i]$  разупорядочен со степенью  $e$ ,

$LateTime(s[i]) = DstTime(i) - DstTime(i-e)$

Кроме того, используя нотацию, похожую на принятую в параграфе 4.2, это можно записать в форме

$LateTime(s[i]) = DstTime(i) - DstTime(j)$ , для  $\min\{j | 1 \leq j < i\}$  при  $s[j] > s[i]$ .

### 4.3.4. Обсуждение

Показатель смещения может помочь в предсказании, когда разупорядоченные пакеты будут полезны в общей системе буферизации получателя с ограничениями. Предел может быть задан временем хранения до завершения цикла (как в буферах компенсации вариаций задержки).

Отметим, что показатель IPDV (IP Packet Delay Variation) [RFC3393] дает вариации задержки в одном направлении относительно предшествующего пакета в потоке от источника. Задержка и IPDV служат индикатором наличия в приёмном буфере достаточного пространства для адаптации к поведению сети и восстановления порядка. При потере более раннего пакета в последовательности значение IPDV обязательно становится неопределённым, а LateTime может стать единственным способом оценки полезности пакета.

В случае буферов компенсации вариаций задержки есть обстоятельства, при которых получатель маскирует потери предполагаемым временем получения запоздавшего пакета. Однако при получении этого пакета с нарушением порядка значение LateTime определяет фактическую полезность этого пакета. IPDV больше не применяется, поскольку получатель организует новое планирование (play-out) с дополнительной задержкой в буфере для адаптации к подобным событиям в будущем (это требует лишь минимальной обработки).

Комбинация потерь и нарушения порядка влияет на показатель LateTime. Если представлена последовательность прибытия 1, 10, 5 (пакеты 2, 3, 4 и 6 - 9 потеряны), LateTime не будет точно указывать, насколько запоздал пакет 5 по сравнению с предполагаемым временем. IPDV [RFC3393] не учитывает такие ситуации, поскольку отсутствуют соседние пакеты. В предположении периодического потока [RFC3432] ожидаемое время прибытия можно определить для всех пакетов, но это будет по сути показателем измерения вариации задержки в одной точке (как определено в рекомендациях ITU-T [I.356] и [Y.1540]), а не показателем разупорядочения.

Результаты выборки LateTime можно указать на гистограмме для представления числа значений времени буферизации, требуемых для размещения разупорядоченных пакетов, и разрешить настройку буфера на этой основе. Информативной может быть кумулятивная функция распределения (cumulative distribution function или CDF) времени буферизации в зависимости от процента размещаемых в буфере разупорядоченных пакетов.

## 4.4. Разупорядочение по смещению байтов

Разупорядоченным пакетам можно назначить значение смещения, указывающее в байтах объем памяти, которая потребуется получателю для их размещения. Показатель смещения рассчитывается лишь для разупорядоченных пакетов в соответствии с измерением одиночных нарушений из раздела 3. Показатель одиночного нарушения порядка.

### 4.4.1. Имя показателя

Type-P-Packet-Byte-Offset-Stream

### 4.4.2. Параметры показателя

Применяются определённые выше параметры, включая необязательные параметры SrcByte и PayloadSize, а также:

- ByteOffset( $s[i]$ ) - смещение пакета  $s[i]$  в байтах.

### 4.4.3. Определение

Смещение ByteOffset для разупорядоченного пакета  $s[i]$  - сумма размеров данных (payload) в пакетах, соответствующим приведённым ниже критериям:

- прибытие до разупорядоченного пакета  $s[i]$ ;
- порядковый номер отправки  $s > s[i]$ .

Пакет, соответствующий обоим критериям, обычно буферизуется, пока не поступит следующий за ним пакет. Отметим, что эти критерии применяются ко всем пакетам (независимо от нарушения порядка).

Для разупорядоченного пакета  $s[i]$  со степенью разупорядочения  $e$

$ByteOffset(s[i]) = \text{Sum}[\text{соответствующий пакет}]$   
 $= \text{Sum}[\text{PayloadSize}(\text{пакет с } i-1, \text{ если он соответствует}),$   
 $\text{PayloadSize}(\text{пакет с } i-2, \text{ если он соответствует}), \dots$   
 $\text{PayloadSize}(\text{пакет с } i-e \text{ всегда соответствует})]$

С использованием представленной выше нотации

```
ByteOffset(s[i]) =
    Sum[данные s[j], где s[j]>s[i] и i > j >= i-e]
```

#### 4.4.4. Обсуждение

Отметим, что размер буфера для компенсации разупорядочения сильно зависит от тестового потока в части дистанции между тестовыми пакетами и их размера, особенно при переменном размере. В этих и иных обстоятельствах может оказаться наиболее полезным характеризовать смещение размером сохраняемых пакетов с использованием показателя Type-P-packet-Byte-Offset-Stream. Этот показатель может помочь в предсказании полезности разупорядоченных пакетов в общей системе буферизации получателя с конечным размером. Предел указывается числом байтов, которые можно поместить в буфер.

Результаты выборки ByteOffset можно указать на гистограмме для представления для представления числа размеров буфера, требуемых для разупорядоченных пакетов, и разрешить настройку буфера на основе этих значений. Информативной может быть CDF размера буфера в зависимости от процента размещаемых в буфере разупорядоченных пакетов.

### 4.5. Разрыв в нарушении порядка

#### 4.5.1. Имена показателей

Type-P-Packet-Reordering-Gap-Stream

Type-P-Packet-Reordering-GapTime-Stream

#### 4.5.2. Параметры

Применяются определённые выше параметры с добавлением соглашения о том, что  $i' > i$  и  $j' > j$ , определения

- Gap(s[j']) - интервал между нарушениями порядка (Reordering Gap) для пакета s[j'] (число сообщений)

и **необязательного** параметра

- GapTime(s[j']) интервал между нарушениями порядка (Reordering Gap) для пакета s[j'] (в секундах).

#### 4.5.3. Определение разрыва разупорядоченности

Все разупорядоченные пакеты связанные с пакетом, на котором порядок восстановлен, определяемым как упорядоченный пакет s[j], прибывший с наименьшим значением  $j$  ( $1 \leq j < i$ ), для которого  $s[j] > s[i]$ .

Отметим, что обнаружение прерывания нарушения порядка пакетом s[j] вызывает разрыв последовательности с  $s > \text{NextExp}$  при оценке показателя одиночного пропуска номеров, как описано в параграфе 3.4. Определение пропуска номеров.

Напомним, что  $i - e = \min(j)$ . Последующие разупорядоченные пакеты могут быть связаны с тем же s[j] или другим прерыванием нарушения порядка. Этот факт используется в определении интервала без нарушений (Reordering Gap).

#### 4.5.4. Определение интервала между нарушениями порядка

Интервалом между нарушениями порядка является дистанция между последовательными прерываниями разупорядочения. Показатель Type-P-Packet-Reordering-Gap-Stream присваивает значение Gap(s[j']) (всем) пакетам в потоке (и значение GapTime(s[j']), если оно используется).

Если выполняются все указанные ниже условия

определено, что пакет s[j'] прерывает нарушение порядка на основе приёма разупорядоченного пакета s[i'] со степенью разупорядочения  $e'$ ;

уже было обнаружено предшествующее прерывание нарушения порядка s[j] на основе прибытия разупорядоченного пакета s[i] со степенью разупорядочения  $e$ ;

$i' > i$

не было обнаружено прерывания разупорядоченности между  $j$  и  $j'$ ,

тогда интервал между нарушениями порядка (Reordering Gap) для пакета s[j'] определяется разностью между позициями прибытия при прерывании разупорядочения, как показано ниже.

$$\text{Gap}(s[j']) = (j') - (j)$$

Интервалы можно также указать временем

$$\text{GapTime}(s[j']) = \text{DstTime}(j') - \text{DstTime}(j)$$

В противном случае

Gap(s[j']) (и GapTime(s[j'])) для пакета s[j'] имеет значение 0.

#### 4.5.5. Обсуждение

Когда можно различить отдельные разрывы разупорядочения, их число можно указать (вместе с описанием разрыва, таким как число разупорядоченных пакетов, связанных с разрывом, а также их степени разупорядоченности и смещения). Интервалы между разрывами нарушения порядка в выборках можно указать на гистограмме для представления частоты разных разрывов. Указание режима, среднего значения, диапазона и т. п. также может обобщать распределения.

Показатель Gap может помочь при сопоставлении частоты разрывов нарушения порядка с их причинами. Величина разрыва также полезна разработчикам получателей, показывая период разрывов. Сочетание разрывов и степени разупорядочения показывает, будут ли получатели обрабатывать наложение разупорядочения пакетов.

## 4.6. Работа без нарушения порядка

В этом разделе определяется показатель, основанный на числе упорядоченных пакетов между разупорядоченными.

### 4.6.1. Имена показателей

Type-P-Packet-Reordering-Free-Run-x-numruns-Stream

Type-P-Packet-Reordering-Free-Run-q-squruns-Stream

Type-P-Packet-Reordering-Free-Run-p-numpkts-Stream

Type-P-Packet-Reordering-Free-Run-a-accpkts-Stream

### 4.6.2. Параметры

Используются определённые выше параметры, а также:

- г - счётчик запусков;
- х - число запусков, а также число разупорядоченных пакетов;
- а - аккумулятор упорядоченных пакетов;
- р - число пакетов (по завершении потока  $p=(x+a)=L$ );
- q - сумма квадратов, учтённых запусков.

### 4.6.3. Определение

По мере прибытия пакетов выборки к получателю они учитываются счётчиком Reordering-Free. Отметим, что в соответствии с этим определением минимальный размер цикла равен 0. Пример псевдокода приведён ниже.

```

r = 0; /* r - счётчик */
x = 0; /* x - число запусков */
a = 0; /* a - аккумулятор упорядоченных пакетов */
p = 0; /* p - число пакетов */
q = 0; /* q - сумма квадратов учтённых запусков */

while(прибывает пакет с номером s)
{
    p++;
    if (s >= NextExp) /* s упорядочен */
        then r++;
        a++;
    else /* s нарушает порядок */
        q+= r*r;
        r = 0;
        x++;
}

```

Прибытие каждого упорядоченного пакета увеличивает счётчик запусков и аккумулятор упорядоченных пакетов, а каждый разупорядоченный пакет сбрасывает счётчик запусков после учёта его значения в сумме квадратов.

Каждое прибытие неупорядоченного пакета ведёт к новому подсчёту запусков. Длительная работа счётчика связана с продолжительным поступлением упорядоченных пакетов, а короткие интервалы работы счётчика указывают частое или многопакетное разупорядочение.

Процент упорядоченных пакетов составляет  $100 \cdot a/p$ , средняя продолжительность работы без нарушения порядка -  $a/x$ . Счётчик q показывает отклонение работы без нарушения порядка от среднего значения путём сравнения  $q/a$  с  $a/x$  ( $(q/a)/(a/x)$ ).

### 4.6.4. Обсуждение и иллюстрация

Параметры Type-P-packet-Reordering-Free-Run-Stream дают краткую сводку характеристик нарушения порядка, включая среднюю продолжительность работы без нарушения и вариации этих периодов, поэтому основным применением этого показателя является оценка работы сети.

Для 36 пакетов с 3 периодами из 11 упорядоченных пакетов мы имеем

```

p = 36
x = 3
a = 33
q = 3 * (11*11) = 363
средний интервал работы без нарушений = 11
q/a = 11
(q/a) / (a/x) = 1,0

```

Для 36 пакетов с тремя интервалами работы без ошибок, из которых два имеют размер 1, а третий - 31, мы имеем

```

p = 36
x = 3
a = 33
q = 1 + 1 + 961 = 963
средний интервал работы без нарушений = 11
q/a = 29.18
(q/a) / (a/x) = 2,65

```

Вариативность продолжительности работы без нарушения порядка проявляется в разнице между значениями q (сумма квадратов продолжительности интервалов без ошибок) и сравнении средней продолжительности интервалов с отношением  $(q/a)/(a/x)$  - 1 означает одинаковые интервалы работы без нарушения порядка.

## 5. Показатель, нацеленный на оценку получателя - TCP

В этом разделе описан показатель, содержащий информацию, связанную с влиянием нарушения порядка на TCP. Однако для оценки производительности TCP тестовый поток **должен** быть близок к интересующему отправителю TCP. В [RFC3148] перечислены конкретные аспекты алгоритмов контроля перегрузок, которые должны быть заданы. Кроме того, в соответствии с RFC 3148 для показателей Bulk Transfer Capacity (ёмкости объемной передачи) **следует** иметь инструменты, различающие 3 варианта нарушения порядка пакетов (см. 3.3. Определение). Показатели выборки, определённые выше, удовлетворяют требованиям классификации пакетов со слабым или значительным нарушением порядка. Определённый здесь показатель добавляет возможность оценки способности нарушения порядка приводить к превышению порога DUP-ACK, вызывающему алгоритм Fast Retransmit. Дополнительные инструменты для TCP указаны в [Mat03].

### 5.1. Имя показателя

Type-P-Packet-n-Reordering-Stream

### 5.2. Обозначение параметров

Пусть  $n$  будет положительным целым числом (параметр), а  $k$  - положительным целым числом, равным количеству переданных пакетов (размер выборки). Пусть  $l$  - неотрицательное целое число, представляющее пакеты с нарушением порядка из числа переданных  $k$  пакетов (отметим, что  $k$  и  $l$  не связаны, потери могут сделать  $l < k$ , а дубликаты -  $l > k$ ). Свяжем с каждым переданным пакетом номер от 1 до  $k$  в порядке их передачи.

Пусть  $s[1], s[2], \dots, s[l]$  - исходные порядковые номера пакетов в порядке их получения на приёмной стороне.

### 5.3. Определения

Определение 1. Принятый пакет  $i$  ( $n < i \leq l$ ) с исходным номером  $s[i]$ , считается  $n$ -разупорядоченным тогда и только тогда, когда для всех  $j$  таких, что  $i-n \leq j < i$ , выполняется условие  $s[j] > s[i]$ .

Утверждение. Если в соответствии с этим определением пакет  $n$ -разупорядочен и  $0 < n' < n$ , пакет также  $n'$ -разупорядочен.

Примечание. Это определение представлено кодом C в Приложении A. Код обнаруживает и сообщает о  $n$ -разупорядочении при  $n$  от 1 до заданного значения (в коде MAXN = 100). Предполагается, что относящееся к TCP значение  $n$  является порогом TCP ACK (3 в соответствии с п. 2 в параграфе 3.2 [RFC 2581]).

Это определение не назначает  $n$  всем разупорядоченным пакетам, как определено для одиночного измерения, в частности при переупорядочении последовательных блоков (в приёмной последовательности  $s=\{1,2,3,7,8,9,4,5,6\}$  пакеты 4 - 6 разупорядочены, но лишь пакет 4 является  $n$ -разупорядоченным с  $n=3$ ).

Определение 2. Степенью  $n$ -разупорядоченности выборки является  $m/l$ , где  $m$  - число  $n$ -разупорядоченных пакетов в выборке.

Определение 3. Степенью монотонного переупорядочения выборки является степень 1-разупорядоченности.

Определение 4. Выборку считают упорядоченной, если степень монотонной переупорядоченности равна 0.

Примечание. Как следует из приведённого выше утверждения, если монотонная разупорядоченность группы равна 0,  $n$ -разупорядочение группы равно 0 для всех  $n$ .

### 5.4. Обсуждение

Степень  $n$ -разупорядоченности можно выразить в процентах, умножив значение из определения 2 на 100.

Показатель  $n$ -разупорядоченности полезен для сопоставления с порогом дубликатов ACK на данном пути. Например, если на пути наблюдается не более чем 5-разупорядочение, порог DUP-ACK = 6 может предотвратить ненужные повторы передачи. Важными особыми случаями являются  $n=1$  и  $n=3$ .

- При  $n=1$  отсутствие 1-разупорядочения означает, что получатель видит монотонный рост порядкового номера по сравнению с предыдущим пакетом.
- При  $n=3$  отправитель NewReno TCP будет повторно передавать 1 в ответ на экземпляр 3-разупорядочения и, следовательно, станет считать этот пакет потерянным при контроле перегрузки (отправитель будет снижать вдвое своё окно перегрузки [RFC2581]). Значение 3 принято по умолчанию для протоколов SCTP (Stream Control Transport Protocol) [RFC2960] и DCCP (Datagram Congestion Control Protocol) [RFC4340] при использовании с Congestion Control ID 2: TCP-like Congestion Control [RFC4341].

$N$ -разупорядочение выборки можно представить на гистограмме для указания частоты при каждом значении  $n$ .

Отметим, что определение  $n$ -разупорядочения не позволяет точно предсказать число повторных пакетов, необоснованно переданных отправителем TCP в некоторых обстоятельствах, например в случае близко расположенных одиночных случаев нарушения порядка. На поведение отправителя влияет время и расположение.

Назначение  $n$ -разупорядочения иногда эквивалентно степени нарушения порядка  $e$ , но  $n$ -разупорядочение отличается в двух аспектах:

1.  $n$  - число предшествующих пакетов с последовательными номерами прибытия к получателю;
2. пакеты с нарушением порядка (Type-P-Reordered=TRUE) могут не быть  $n$ -разупорядоченными, но иметь степень  $e$  (см. примеры).

## 6. Вопросы измерений и реализации

Результаты тестов будут зависеть от временных интервалов между измерительными пакетами (как в источнике, так и при транспортировке, где эти интервалы могут меняться). Очевидно, что пакеты, передаваемые нечасто (например, через 10 секунд), вряд ли будут переупорядочены.

Чтобы измерить нарушение порядка для приложения в соответствии с определёнными здесь показателями, **рекомендуется** использовать такую же картину отправки, которая применяется в интересующем приложении. В любом случае, метод генерации пакетов, включая все параметры потока, **должен** точно указываться в отчёте об измерении.

- Чтобы сделать выводы о приложениях TCP, **требуется** применять потоки в стиле TCP как в [RFC3148].
- Для приложений в реальном масштабе времени **рекомендуются** периодические потоки как в [RFC3432].

Показатели из разделов 3 и 4 можно сообщать с другими метриками IPPM, используя потоки Пуассона [RFC2330]. Пуассоновские потоки представляют «беспристрастную выборку» производительности сети для показателей потери и задержки пакетов. Однако было бы некорректно делать выводы об указанных выше категориях применения с использованием показателей нарушения порядка, измеренных с помощью потоков Пуассона.

Разработчики тестовых потоков могут предпочесть периодическую отправку, чтобы поддерживать известный сдвиг во времени и упростить анализ результатов (как описано в [RFC3432]). В таких случаях **рекомендуется** выбирать интервал для периодических потоков так, чтобы воспроизвести ближайший ожидаемый интервал между исходными пакетами. Тестировщики должны понимать, что потоки, передаваемые с максимальной скоростью канала, **должны** быть ограничены по продолжительности, и **должны** рассматривать потерю пакетов как указание перегрузки, вызванной потоком, останавливая дальнейшее тестирование.

При намерении сравнивать результаты независимых измерений разупорядочения **рекомендуется** применять одинаковые параметры тестовых потоков в каждой измерительной системе.

Размер пакетов также может изменяться в попытке обнаружить случаи параллельной обработки (они могут вызывать устойчивое переупорядочение). Например, импульс максимально длинных (MTU) пакетов со скоростью линии, за которым следует импульс наиболее коротких пакетов, может быть примером эффективного шаблона детектирования. Возможны и иные картины (шаблоны) размеров.

Критерий необратимого порядка и все показатели, описанные выше, сохраняют действие и полезны в случаях, когда поток пакетов сталкивается с потерями или одновременно с потерями и разупорядочением. Иными словами, потери сами по себе не ведут к признанию последующих пакетов разупорядоченными.

Поскольку в определении показателей могут применяться порядковые номера с конечным диапазоном значений, возможно достижение максимального номера и сброс в 0 во время измерения. По определению значение NextExp не может снижаться и все пакеты, полученные после перехода через 0, будут считаться переупорядоченными. Сброса порядкового номера можно избежать, применяя сочетание размера счётчика и продолжительности теста, при котором достижение максимального номера невозможно (и отсчёт начинается с 0 при запуске). Кроме того, нумерация на уровне сообщений обеспечивает более медленный рост номеров. В некоторых случаях может быть желательно метрологическое смягчение проблемы (например, при длительных тестах). Для этого служат указанные ниже меры.

1. Должна быть проверка перехода через максимум. Различия в порядковых номерах последовательных пакетов более чем на половину диапазона практически невозможны, поэтому столь большие разрывы следует связывать с переходом через максимум.
2. Используемые в расчётах порядковые номера представляются с достаточной точностью (разрядностью). Номера корректируются (эквивалент добавления значащей цифры) при обнаружении достижения максимума.
3. Переупорядоченные пакеты с номерами в конце диапазона также должны обнаруживаться для применения корректировки с учётом перехода через максимум.

В идеале тестовый инструмент должен иметь возможность использовать все предшествующие пакеты в любой точке тестового потока. На практике возможность определения степени разупорядочения будет ограничиваться размером хранилища предшествующих пакетов. Сохранение лишь указывающих разрывы пакетов (и позиции их прибытия) позволит сократить требуемый размер хранилища.

Другим решением является применение скользящего окна истории пакетов, где размер окна задаёт верхняя граница приемлемой степени нарушения порядка. Эта граница может состоять из нескольких пакетов или из пакетов за несколько секунд в зависимости от планируемого анализа. При отбрасывании всей информации за пределами этого окна степень разупорядочения или n-разупорядочения может быть выражена как превышающая размеры окна, если сведения о разрыве разупорядочения были отброшены и вычисление интервала (Gap) невозможно.

Для игнорирования дубликатов пакетов также требуется пространство хранения. Здесь отслеживание порядковых номеров отсутствующих пакетов может снизить требования к размеру хранилища. Отсутствующие в конечном итоге могут быть сочтены потерянными или разупорядоченными (если они придут). Списка отсутствующих пакетов и наибольшего принятого (на данный момент) порядкового номера (NextExp - 1) достаточно для определения дубликатов (при условии управляемого размера хранилища для пакетов, отсутствующих в результате потерь).

Важно отметить, что в практических сетях IP также может быть ограниченная возможность «хранить» пакеты даже при возникновении временных петель в маршрутизации. Таким образом, максимальное хранилище для показателей переупорядочения (и их сложности) будет приближаться к числу пакетов в выборке (K) лишь в случае, когда время передачи K пакетов мало по сравнению с максимальным временем их нахождения в сети. Другим возможным ограничением для хранилища является максимальный размер поля порядкового номера при условии, что большинство номеров тестовых пакетов на практике не достигают максимальных номеров (использование поля целиком).

В заключение отметим, что определение степени нарушения порядка и интервалов между случаями разупорядочения осложняется при наличии перекрывающихся или вложенных событий. Сложности тестового инструментария и разупорядочения связаны напрямую.

## 6.1. Пассивные измерения

Как и для других показателей IPPM, определения рассчитаны прежде всего на активные измерения.

Предполагая, что требуемые сведения (номер сообщения) включены в данные пакета (возможно в прикладные заголовки, такие как RTP), показатели нарушения порядка можно оценить и в пассивных измерениях. Можно также оценить порядок в любой точке пути от источника к получателю, понимая, что результаты могут отличаться от результатов измерений у получателя (где можно оценить влияние переупорядочения на приложения).

Можно применить эти показатели для оценки переупорядочения в потоке отправителя TCP. В этом случае порядковые номера источника основываются на смещении байтов или нумерации сегментов. Поскольку поток может включать повторы передачи в результате потерь или нарушения порядка, должны быть приняты меры против учёта повторных пакетов как разупорядоченных. Дополнительная нумерация  $s$  или SrcTime поможет избежать неоднозначности при активном измерении, а (необязательные) временные метки TCP [RFC1323] - при пассивном.

## 7. Примеры оценки порядка прибытия

В этом разделе приведены примеры, показывающие работу критерия необратимости порядка и  $n$ -переупорядочения в сравнении, а также важность количественной оценки переупорядочения во времени, байтах и позиции (смещении).

Пакеты здесь указываются исходными порядковыми номерами, если не указано иное. Например, «пакет 4» означает пакет, переданный источником с номером 4, и читателю следует обращаться при необходимости к таблицам номеров.

### 7.1. Один пакет с нарушением порядка

В таблице 1 показан случай нарушения порядка для пакета 4. Пакеты указаны в порядке их прибытия с нумерацией по сообщениям. Все пакеты имеют PayloadSize=100 байтов и SrcByte=( $s \times 100$ )-99 для  $s=1,2,3,4,\dots$

Таблица 1. Пример с нарушением порядка для пакета 4. Порядок отправки источником ( $s @Src$ ): 1,2,3,4,5,6,7,8,9,10.

$s @Dst$	NextExp	Src Time	Dst Time	Delay	IPDV	Dst Order	Byte Offset	Late Time
1	1	0	68	68		1		
2	2	20	88	68	0	2		
3	3	40	108	68	0	3		
5	4	80	148	68	-82	4		
6	6	100	168	68	0	5		
7	7	120	188	68	0	6		
8	8	140	208	68	0	7		
4	9	60	210	150	82	8	400	62
9	9	160	228	68	0	9		
10	10	180	248	68	0	10		

$s$  Порядковый номер отправки пакета.

NextExp Значение NextExp при получении пакета (до обновления).

SrcTime Временная метка отправки пакета (мсек).

DstTime Временная метка прибытия пакета (мсек).

Delay Задержка пакета в одном направлении (мсек).

IPDV Вариация задержки (IP Packet Delay Variation или IPDV) в мсек.  $IPDV = Delay(SrcNum) - Delay(SrcNum - 1)$ .

DstOrder Порядок прибытия пакетов к получателю.

Byte Offset Смещение переупорядоченного пакета (в байтах).

LateTime Задержка переупорядоченного (мсек).

Можно видеть, что при получении пакета 4 значение NextExp=9 и пакет считается переупорядоченным. Степень переупорядочения рассчитывается, как показано ниже.

В нотации  $\langle s[1], \dots, s[i], \dots, s[L] \rangle$  принятые пакеты представляются как

$$s = 1, 2, 3, 5, 6, 7, 8, 4, 9, 10$$

$$i = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10$$

Применим определение Type-P-Packet-Reordering-Extent-Stream

когда  $j=7, 8 > 4$  и степень разупорядоченности составляет не менее 1;

когда  $j=6, 7 > 4$  и степень разупорядоченности составляет не менее 2;

когда  $j=5, 6 > 4$  и степень разупорядоченности составляет не менее 3;

когда  $j=4, 5 > 4$  и степень разупорядоченности составляет не менее 4.

когда  $j=3, 3 < 4$  и 4 является максимальной степенью переупорядочения  $e=4$  (в предположении отсутствия более раннего разрыва переупорядочения, как в этом примере).

Далее рассчитывается время задержки (Late Time,  $210-148=62$  мсек) с использованием DstTime по сравнению с прибытием пакета 5. Если у получателя есть буфер компенсации вариаций задержки, вмещающий более 4 пакетов или хотя бы хранение в течение 62 мсек, пакет 4 может быть полезен. Отметим, что задержка в одном направлении и IPDV указывают необычное поведение пакета 4. Если бы пакет 4 прибыл на 62 мсек раньше, он был бы упорядоченным.

Если все пакеты содержат 100 байтов данных (payload), Byte Offset будет иметь значение 400 байтов.

В соответствии с определениями параграфа 5.1 пакет 4 обозначается как 4-переупорядоченный.

### 7.2. Нарушение порядка двумя пакетами

Таблица 2. Пример с переупорядоченными пакетами 5 и 6. Порядок отправки ( $s @Src$ ): 1,2,3,4,5,6,7,8,9,10.

$s @Dst$	NextExp	Src Time	Dst Time	Delay	IPDV	Dst Order	Byte Offset	Late Time
----------	---------	----------	----------	-------	------	-----------	-------------	-----------

1	1	0	68	68		1		
2	2	20	88	68	0	2		
3	3	40	108	68	0	3		
4	4	60	128	68	0	4		
7	5	120	188	68	-22	5		
5	8	80	189	109	41	6	100	1
6	8	100	190	90	-19	7	100	2
8	8	140	208	68	0	8		
9	9	160	228	68	0	9		
10	10	180	248	68	0	10		

В таблице 2 показан случай, когда пакеты 5 и 6 прибывают после пакета 7 и являются переупорядоченными. Задержки их прибытия невелики (189-188=1, 190-188=2).

В нотации  $\langle s[1], \dots, s[i], \dots, s[l] \rangle$  принятые пакеты представляются как

$$s = 1, 2, 3, 4, 7, 5, 6, 8, 9, 10$$

$$i = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10$$

Рассмотрим сначала пакет 5

когда  $j=5$ ,  $7 > 5$  и степень разупорядоченности составляет не менее 1;

когда  $j=4$ , мы имеем  $4 < 5$ , поэтому максимальной степенью разупорядоченности будет 1 и  $e=1$ .

Рассмотрим пакет 6

когда  $j=6$ ,  $5 < 6$  и степень разупорядоченности ещё не определена;

когда  $j=5$ ,  $7 > 6$  и степень разупорядоченности составляет не менее  $i-j=2$ ;

когда  $j=4$ ,  $4 < 6$  и максимальной степенью разупорядоченности будет 2 и  $e=2$ .

С каждым из этих пакетов можно также связать разрыв переупорядочения. Минимум составляет  $j=5$  (для обоих пакетов) в соответствии с параграфом 4.2.3. Пакет 6 связан с тем же разрывом переупорядочения, что и пакет 5 — разрыв переупорядочения пакетом 7.

В этом случае степень переупорядочения  $e$  ведёт к переоценке размера памяти, требуемой для восстановления порядка. Фактически нужна память лишь для одного пакета (7), но  $e=2$  для пакета 6.

В соответствии с определениями раздела 5 пакет 5 обозначается как 1-переупорядоченный, а пакет 6 не считается  $n$ -переупорядоченным.

Гипотетическая пара отправитель-получатель может без необходимости повторить пакет 5, поскольку он 1-переупорядочен (в соответствии с singleton-показателем). Хотя пакет 6 может не передаваться без необходимости повторно, получатель не может продвигать пакет 7 на вышележащий уровень, пока не придёт пакет 6. Следовательно, показатель одиночного измерения корректно определяет переупорядоченность пакета 6.

### 7.3. Пример с тремя разупорядоченными пакетами

Таблица 3. Пример с переупорядоченными пакетами 4 - 6. Порядок отправки ( $s @Src$ ): 1,2,3,4,5,6,7,8,9,10,11.

$s @Dst$	NextExp	Src Time	Dst Time	Delay	IPDV	Dst Order	Byte Offset	Late Time
1	1	0	68	68		1		
2	2	20	88	68	0	2		
3	3	40	108	68	0	3		
7	4	120	188	68	-88	4		
8	8	140	208	68	0	5		
9	9	160	228	68	0	6		
10	10	180	248	68	0	7		
4	11	60	250	190	122	8	400	62
5	11	80	252	172	-18	9	400	64
6	11	100	256	156	-16	10	400	68
11	11	200	268	68	0	11		

В таблице 3 представлен случай, когда три пакета подряд имеют длительное время доставки (пакеты с  $s = 4, 5, 6$ ). Значения Delay, Late time, Byte Offset очень хорошо отражают это и показывают вариации степени переупорядочения, тогда как IPDV показывает изменение интервала между пакетами 4, 5, 6.

Гистограмма степени переупорядоченности имеет вид

Степень	1	2	3	4	5	6	7
Частота	0	0	0	1	1	1	0

В нотации  $\langle s[1], \dots, s[i], \dots, s[l] \rangle$  принятые пакеты представляются как

$$s = 1, 2, 3, 7, 8, 9, 10, 4, 5, 6, 11$$

$$i = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11$$

Сначала рассчитаем  $n$ -переупорядочение, начав с пакета 4

когда  $n=1$ ,  $7 \leq j < 8$  и  $10 > 4$  пакет является 1-переупорядоченным;

когда  $n=2$ ,  $6 \leq j < 8$  и  $9 > 4$  пакет является 2-переупорядоченным;

когда  $n=3$ ,  $5 \leq j < 8$  и  $8 > 4$  пакет является 3-переупорядоченным;

когда  $n=4$ ,  $4 \leq j < 8$  и  $7 > 4$  пакет является 4-переупорядоченным;

когда  $n=5$ ,  $3 \leq j < 8$  и  $3 < 4$ , 4 является максимальным  $n$ -переупорядочением.

Далее рассмотрим пакет 5[9]

когда  $n=1$ ,  $8 \leq j < 9$  и  $4 < 5$ , поэтому пакет с  $i=9$  не считается  $n$ -переупорядоченным (то же и для пакета 6).

Рассмотрим, связаны ли переупорядоченные пакеты 5 и 6 с тем же разрывом разупорядоченности, что и пакет 4. Тест из параграфа 4.2.3 показывает минимальное значение  $j=4$  для всех трёх пакетов. Все они связаны с разрывом разупорядочения пакетом 7.

Этот пример снова показывает, что определение n-переупорядочения указывает 1 пакет (4) с достаточной степенью n-переупорядочения, что может вызвать ненужные повторы передачи отправителем New Reno TCP (с порогом DUP-ACK 3 или 4). Кроме того, переупорядоченное прибытие пакетов 5 и 6 позволит процессу-получателю передать пакеты 7 - 10 вверх по стеку протоколов (единичное значение Type-P-Reordered = TRUE для пакетов 5 и 6 и они связаны с одним разрывом переупорядочения).

## 7.4. Нарушение порядка множеством пакетов

Таблица 4. Пример с переупорядочением нескольких пакетов. Порядок отправки (s @Src): 1,2,3,4,5,6,7,8,9,10,11,12,13,14,15,16.

```

Разрыв          Разрыв
|-----Gap-----|
s = 1, 2, 3, 6, 7, 4, 5, 8, 9, 10, 12, 13, 11, 14, 15, 16
i = 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16

r = 1, 2, 3, 4, 5, 0, 0, 1, 2, 3, 4, 5, 0, 1, 2, 3, ...
число запусков, n = 1 2 3
заверш. счёта r = 5 0 5

```

(Эти значения рассчитываются после прихода пакета.)

Пакет 4 имеет степень переупорядочения e=2, пакет 5 - e=3, пакет 11 - e=2. Имеется два разрыва переупорядочения, один из которых связан с пакетом 6 (где j=4), другой - с пакетом 12 (где j'=11).

В соответствии с определением интервала без переупорядочения (Reordering Gap)

```

Gap(s[j']) = (j') - (j)
Gap(Packet 12) = (11) - (4) = 7

```

Здесь также три «запуска» без переупорядочения продолжительностью 5, 0 и 5.

Различия между этими двумя показателями с несколькими событиями очевидны здесь. Интервал (Gap) - это дистанция между двумя последовательными фактами переупорядочения, тогда как работа без нарушения порядка - это дистанция между упорядоченными пакетами.

## 8. Вопросы безопасности

### 8.1. Отказ в обслуживании (DoS)

Для этого показателя нужен поток пакетов, передаваемых от одного хоста (источник) к другому (получатель) через промежуточные сети. Этот метод можно использовать для организации DoS-атак, направленных на получателя и/или промежуточные сети.

Администраторам источника, получателя и промежуточных сетей следует заключить двух- или многосторонние соглашения о времени, продолжительности и частоте выполняемых выборок. Использование методов измерений сверх согласованных условий может приводить к незамедлительному отбрасыванию или отклонению пакетов, а также к другим процедурам, определенным затрагиваемыми сторонами.

### 8.2. Конфиденциальность данных пользователей

При активном использовании этого метода создаются потоки для выборки, а не отбираются пакеты пользователей, поэтому конфиденциальность данных не нарушается. При пассивных измерениях должны вноситься ограничения на просмотр содержимого пакетов (только заголовки). Поскольку данные пользователей могут временно сохраняться для анализа размера пакетов, **должны** приниматься меры по сохранению конфиденциальности этих сведений. В большинстве случаев хеширование обеспечит получение значений, подходящих для сравнения содержимого пакетов.

### 8.3. Влияние на показатели

Можно идентифицировать принадлежность пакета или потока пакетов к выборке и на основе этого обработать пакеты по-иному у получателя и/или в промежуточной точке (например, увеличивая или снижая задержку) для влияния на результаты измерений. Можно также создать дополнительные пакеты, которые будут выглядеть частью выборки показателей. Эти дополнительные пакеты будут вероятно сочтены дубликатами или за дубликаты будут приняты исходные пакеты (если они придут позже), что повлияет на результаты выборки.

Требования к устойчивости отбора данных к вмешательству злоумышленников и механизмы обеспечения такой устойчивости приведены в других документах IPPM. Набор требований к протоколу данных можно найти в [RFC3763], а спецификация протокола активных измерений в одном направлении (One-Way Active Measurement Protocol или OWAMP) представлена в [RFC4656]. Посвященные безопасности разделы двух документов OWAMP достаточно обширны и к ним следует обращаться для получения дополнительных сведений.

## 9. Взаимодействие с IANA

Определённые здесь показатели включены в реестр IANA IPPM METRICS REGISTRY, как описано в исходной версии реестра [RFC4148].

Агентство IANA зарегистрировало приведённые ниже показатели в IANA-IPPM-METRICS-REGISTRY-MIB

```

ietfReorderedSingleton OBJECT-IDENTITY
    STATUS      current
    DESCRIPTION
        "Type-P-Reordered"
    REFERENCE
        "Reference RFC 4737, Section 3"
    ::= { ianaIppmMetrics 34 }

ietfReorderedPacketRatio OBJECT-IDENTITY
    STATUS      current
    DESCRIPTION

```

```
"Type-P-Reordered-Ratio-Stream"
REFERENCE
"Reference RFC 4737, Section 4.1"
::= { ianaIppmMetrics 35 }

ietfReorderingExtent OBJECT-IDENTITY
STATUS      current
DESCRIPTION
"Type-P-Packet-Reordering-Extent-Stream"
REFERENCE
"Reference RFC 4737, Section 4.2"
::= { ianaIppmMetrics 36 }

ietfReorderingLateTimeOffset OBJECT-IDENTITY
STATUS      current
DESCRIPTION
"Type-P-Packet-Late-Time-Stream"
REFERENCE
"Reference RFC 4737, Section 4.3"
::= { ianaIppmMetrics 37 }

ietfReorderingByteOffset OBJECT-IDENTITY
STATUS      current
DESCRIPTION
"Type-P-Packet-Byte-Offset-Stream"
REFERENCE
"Reference RFC 4737, Section 4.4"
::= { ianaIppmMetrics 38 }

ietfReorderingGap OBJECT-IDENTITY
STATUS      current
DESCRIPTION
"Type-P-Packet-Reordering-Gap-Stream"
REFERENCE
"Reference RFC 4737, Section 4.5"
::= { ianaIppmMetrics 39 }

ietfReorderingGapTime OBJECT-IDENTITY
STATUS      current
DESCRIPTION
"Type-P-Packet-Reordering-GapTime-Stream"
REFERENCE
"Reference RFC 4737, Section 4.5"
::= { ianaIppmMetrics 40 }

ietfReorderingFreeRunx OBJECT-IDENTITY
STATUS      current
DESCRIPTION
"Type-P-Packet-Reordering-Free-Run-x-numruns-Stream"
REFERENCE
"Reference RFC 4737, Section 4.6"
::= { ianaIppmMetrics 41 }

ietfReorderingFreeRunq OBJECT-IDENTITY
STATUS      current
DESCRIPTION
"Type-P-Packet-Reordering-Free-Run-q-sqruns-Stream"
REFERENCE
"Reference RFC 4737, Section 4.6"
::= { ianaIppmMetrics 42 }

ietfReorderingFreeRunp OBJECT-IDENTITY
STATUS      current
DESCRIPTION
"Type-P-Packet-Reordering-Free-Run-p-numpkts-Stream"
REFERENCE
"Reference RFC 4737, Section 4.6"
::= { ianaIppmMetrics 43 }

ietfReorderingFreeRuna OBJECT-IDENTITY
STATUS      current
DESCRIPTION
"Type-P-Packet-Reordering-Free-Run-a-accpkts-Stream"
REFERENCE
"Reference RFC 4737, Section 4.6"
::= { ianaIppmMetrics 44 }

ietfReordering OBJECT-IDENTITY
STATUS      current
DESCRIPTION
"Type-P-Packet-n-Reordering-Stream"
REFERENCE
"Reference RFC 4737, Section 5"
::= { ianaIppmMetrics 45 }
```

## 10. Нормативные документы

- [RFC791] Postel, J., "Internet Protocol", STD 5, [RFC 791](#), September 1981.
- [RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, [RFC 2119](#), March 1997.
- [RFC2330] Paxson, V., Almes, G., Mahdavi, J., and M. Mathis, "Framework for IP Performance Metrics", [RFC 2330](#), May 1998.
- [RFC2460] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", [RFC 2460](#), December 1998.
- [RFC3148] Mathis, M. and M. Allman, "A Framework for Defining Empirical Bulk Transfer Capacity Metrics", RFC 3148, July 2001.
- [RFC3432] Raisanen, V., Grotefeld, G., and A. Morton, "Network performance measurement with periodic streams", RFC 3432, November 2002.
- [RFC3763] Shalunov, S. and B. Teitelbaum, "One-way Active Measurement Protocol (OWAMP) Requirements", RFC 3763, April 2004.
- [RFC4148] Stephan, E., "IP Performance Metrics (IPPM) Metrics Registry", BCP 108, RFC 4148, August 2005.
- [RFC4656] Shalunov, S., Teitelbaum, B., Karp, A., Boote, J., and M. Zeckauskas, "A One-way Active Measurement Protocol (OWAMP)", [RFC 4656](#), September 2006.

## 11. Дополнительная литература

- [Bel02] J. Bellardo and S. Savage, "Measuring Packet Reordering," Proceedings of the ACM SIGCOMM Internet Measurement Workshop 2002, November 6-8, Marseille, France.
- [Ben99] J.C.R. Bennett, C. Partridge, and N. Shectman, "Packet Reordering is Not Pathological Network Behavior," IEEE/ACM Transactions on Networking, vol. 7, no. 6, pp. 789-798, December 1999.
- [Cia00] L. Ciavattoni and A. Morton, "Out-of-Sequence Packet Parameter Definition (for Y.1540)", Contribution number T1A1.3/2000-047, October 30, 2000, <http://home.comcast.net/~acmacm/IDcheck/0A130470.doc>.
- [Cia03] L. Ciavattoni, A. Morton, and G. Ramachandran, "Standardized Active Measurements on a Tier 1 IP Backbone," IEEE Communications Mag., pp. 90-97, June 2003.
- [I.356] ITU-T Recommendation I.356, "B-ISDN ATM layer cell transfer performance", March 2000.
- [Jai02] S. Jaiswal et al., "Measurement and Classification of Out-of-Sequence Packets in a Tier-1 IP Backbone," Proceedings of the ACM SIGCOMM Internet Measurement Workshop 2002, November 6-8, Marseille, France.
- [Lou01] D. Loguinov and H. Radha, "Measurement Study of Low-bitrate Internet Video Streaming", Proceedings of the ACM SIGCOMM Internet Measurement Workshop 2001 November 1-2, 2001, San Francisco, USA.
- [Mat03] M. Mathis, J. Heffner, and R. Reddy, "Web100: Extended TCP Instrumentation for Research, Education and Diagnosis", ACM Computer Communications Review, Vol 33, Num 3, July 2003, <http://www.web100.org/docs/mathis03web100.pdf>.
- [Pax98] V. Paxson, "Measurements and Analysis of End-to-End Internet Dynamics," Ph.D. dissertation, U.C. Berkeley, 1997, <ftp://ftp.ee.lbl.gov/papers/vp-thesis/dis.ps.gz>.
- [RFC793] Postel, J., "Transmission Control Protocol", STD 7, [RFC 793](#), September 1981.
- [RFC1323] Jacobson, V., Braden, R., and D. Borman, "TCP Extensions for High Performance", [RFC 1323](#), May 1992.
- [RFC2581] Allman, M., Paxson, V., and W. Stevens, "TCP Congestion Control", RFC 2581, April 1999.
- [RFC2679] Almes, G., Kalidindi, S., and M. Zeckauskas, "A One-way Delay Metric for IPPM", [RFC 2679](#), September 1999.
- [RFC2680] Almes, G., Kalidindi, S., and M. Zeckauskas, "A One-way Packet Loss Metric for IPPM", [RFC 2680](#), September 1999.
- [RFC2960] Stewart, R., Xie, Q., Morneault, K., Sharp, C., Schwarzbauer, H., Taylor, T., Rytina, I., Kalla, M., Zhang, L., and V. Paxson, "Stream Control Transmission Protocol", [RFC 2960](#), October 2000.
- [RFC3393] Demichelis, C. and P. Chimento, "IP Packet Delay Variation Metric for IP Performance Metrics (IPPM)", RFC 3393, November 2002.
- [RFC4340] Kohler, E., Handley, M., and S. Floyd, "Datagram Congestion Control Protocol (DCCP)", [RFC 4340](#), March 2006.
- [RFC4341] Floyd, S. and E. Kohler, "Profile for Datagram Congestion Control Protocol (DCCP) Congestion Control ID 2: TCP-like Congestion Control", [RFC 4341](#), March 2006.
- [TBABAJO2] T. Banka, A. Bare, A. P. Jayasumana, "Metrics for Degree of Reordering in Packet Sequences", Proc. 27th IEEE Conference on Local Computer Networks, Tampa, FL, Nov. 2002.
- [Y.1540] ITU-T Recommendation Y.1540, "Internet protocol data communication service - IP packet transfer and availability performance parameters", December 2002.

## 12. Благодарности

Авторы хотели бы поблагодарить Matt Zeckauskas, Jon Bennett (автор двух параграфов для работы без нарушения порядка) и Matt Mathis за полезные дискуссии. Спасибо David Newman, Henk Uijterwaal, Mark Allman, Vern Paxson, Phil Chimento за их рецензии и предложения, а Michal Przybylski за то, что он поделился опытом реализации в рассылке ippm-list. Anura Jayasumana и Nischal Piratla предоставили свою незавершённую работу [TBABAJO2]. Авторы с благодарностью отмечают фундамент, заложенный авторами модели оценки производительности IP [RFC2330].

**Приложение А. Пример реализации на языке С (информативный)**

Ниже представлены два примера кода С для определений нарушений порядка.

Пример 1. n-reordering

```
#include <stdio.h>

#define MAXN 100

#define min(a, b) ((a) < (b)? (a): (b))
#define loop(x) ((x) >= 0? x: x + MAXN)

/*
 * Считывается и возвращается новый порядковый номер. При отсутствии
 * номера возвращает сигнал EOF (хотя бы 1 раз). В этом примере номера
 * приходят от stdin, в реальном тесте они приходят из сети.
 */

int
read_sequence_number()
{
    int    res, rc;
    rc = scanf("%d\n", &res);
    if (rc == 1) return res;
    else return EOF;
}

int
main()
{
    int    m[MAXN];          /* Имеем m[j-1] == число пакетов с
                            * j-переупорядочением. */
    int    ring[MAXN];      /* Последний увиденный номер. */
    int    r = 0;          /* Кольцевой указатель для следующей записи. */
    int    l = 0;          /* Число считанных порядковых номеров. */
    int    s;              /* Последний считанный номер. */
    int    j;

    for (j = 0; j < MAXN; j++) m[j] = 0;
    for (;(s = read_sequence_number()) != EOF;l++,r=(r+1)%MAXN) {
        for (j=0; j<min(l, MAXN)&&s<ring[loop(r-j-1)];j++) m[j]++;
        ring[r] = s;
    }

    for (j = 0; j < MAXN && m[j]; j++)
        printf("%d-reordering = %f%%\n", j+1, 100.0*m[j]/(l-j-1));
    if (j == 0) printf("no reordering\n");
    else if (j < MAXN) printf("no %d-reordering\n", j+1);
    else printf("only up to %d-reordering is handled\n", MAXN);
    exit(0);
}

/* Пример 2. Сравнение singleton и n-reordering =====
Author: Jerry Perser 7-2002 (измен. асм 12-2004)
Compile: $ gcc -o jpbboth file.c
Usage:   $ jpbboth 1 2 3 7 8 4 5 6 (порядок пакетов задаёт команда)
Отметим, что при копировании строка 59 может потребовать правки
*/
```

```
#include <stdio.h>

#define MAXN 100
#define min(a, b) ((a) < (b)? (a): (b))
#define loop(x) ((x) >= 0? x: x + MAXN)

/* Глобальные счётчики */
int receive_packets=0;      /* число полученных пакетов */
int reorder_packets_A1=0;  /* число полученных пакетов (singleton) */
int reorder_packets_Stas=0; /* число полученных пакетов(n-reordering)*/

/* Функция проверки нарушения порядка текущим пакетом
 * 0 = упорядочен
 * 1 = переупорядочен
 */
int testorder1(int seqnum) // A1
{
    static int NextExp = 1;
    int iReturn = 0;

    if (seqnum >= NextExp) {
        NextExp = seqnum+1;
    } else {
        iReturn = 1;
    }
    return iReturn;
}
```

```

}
int testorder2(int seqnum) // Станислав
{
    static int ring[MAXN]; /* Последний увиденный номер. */
    static int r = 0; /* Кольцевой указатель для следующей записи. */
    int l = 0; /* Число прочитанных номеров. */
    int j;
    int iReturn = 0;

    l++;
    r = (r+1) % MAXN;
    for (j=0; j<min(l, MAXN) && seqnum<ring[loop(r-j-1)]; j++)
        iReturn = 1;
    ring[r] = seqnum;
    return iReturn;
}
int main(int argc, char *argv[])
{
    int i, packet;
    for (i=1; i< argc; i++) {
        receive_packets++;
        packet = atoi(argv[i]);
        reorder_packets_Al += testorder1(packet); // singleton
        reorder_packets_Stas += testorder2(packet); //n-reord.
    }
    printf("Received packets = %d, Singleton Reordered = %d, n-
reordered = %d\n", receive_packets, reorder_packets_Al,
reorder_packets_Stas );
    exit(0);
}

```

Ссылка

ISO/IEC 9899:1999 (E) с поправками ISO/IEC 9899:1999/Cor.1:2001 (E). Опубликовано также в The C Standard: Incorporating Technical Corrigendum 1, British Standards Institute, ISBN: 0-470-84573-2, Hardcover, 558 pages, September 2003.

## Приложение В. Оценка порядка фрагментов (информативное)

В разделе 3 сказано, что сборка фрагментов выполняется до проверки порядка доставки, но можно оценить порядок доставки и до сборки фрагментов, используя похожие процедуры. В этом приложении приведены определения и процедуры для обнаружения переупорядочения в потоке пакетов, содержащем фрагменты.

### В.1. Имя показателя

Имя показателя Type-P-Reordered сохраняется, но нужны дополнительные параметры. В этом приложении предполагается, что фрагментирующее пакеты устройство передаёт их в порядке роста смещений для фрагментов. Ранние версии Linux передавали фрагменты в обратном порядке и это следует проверять.

### В.2. Дополнительные параметры показателей

MoreFrag - состояние флага More Fragments в заголовке IP.

FragOffset - смещение от начала фрагментированного пакета в 8-октетных блоках (из заголовка IP).

FragSeq# - порядковый номер фрагмента из заголовка IP фрагментированного пакета, для которого сейчас проверяется порядок. При значении 0 оценка фрагмента не производится.

NextExpFrag - следующий фрагмент, ожидаемый получателем в 8-октетных блоках. При установке значения 0 фрагмент не оценивается.

Предполагается, что порядковый номер пакета (s) совпадает с порядковым номером заголовка IP, а значение NextExp не меняется при упорядоченной доставке фрагментов и обновляется лишь при поступлении последнего фрагмента или полного пакета.

Отметим, что пакеты с потерянными фрагментами, **должны** считаться потерянными и статус переупорядочения любых доставленных фрагментов **должен** исключаться из показателей выборки.

### В.3. Определение

Type-P-Reordered обычно имеет значение False (пакет упорядочен) при выполнении приведённых ниже условий.

- Порядковый номер  $s \geq \text{NextExp}$ .
- Смещение фрагмента  $\text{FragOffset} \geq \text{NextExpFrag}$ .

Однако более эффективно будет точно определить условия переупорядочения и устанавливать Type-P-Reordered = False, если они не выполняются.

Type-P-Reordered определяется как True (пакет переупорядочен) при указанных ниже условиях (NextExp не меняется).

Случай 1:  $s < \text{NextExp}$

Случай 2:  $s < \text{FragSeq\#}$

Случай 3:  $s \geq \text{NextExp}$ ,  $s = \text{FragSeq\#}$  и  $\text{FragOffset} < \text{NextExpFrag}$

Это определение можно проиллюстрировать псевдокодом, представленным ниже, который можно несколько упростить.

```

NextExp=0;
NextExpFrag=0;
FragSeq#=0;

```

```

while(пакеты прибывают с s, MoreFrag, FragOffset)
{
if (s>=NextExp AND MoreFrag==0 AND s>=FragSeq#){
/* обычный пакет или последний фрагмент при упорядоченной доставке */
NextExp = s+1;
FragSeq# = 0;
NextExpFrag = 0;
Reordering = False;
}
if (s>=NextExp AND MoreFrag==1 AND s>FragSeq#>=0){
/* прибыл фрагмент нового пакета, возможно с большим чем у текущего
фрагментированного пакета порядковым номером */
FragSeq# = s;
NextExpFrag = FragOffset+1;
Reordering = False;
}
if (s>=NextExp AND MoreFrag==1 AND s==FragSeq#){
/* прибыл фрагмент «текущего пакета» */
if (FragOffset >= NextExpFrag){
NextExpFrag = FragOffset+1;
Reordering = False;
}
else{
Reordering = True; /* фрагмент переупорядочен */
}
}
if (s>=NextExp AND MoreFrag==1 AND s < FragSeq#){
/* случай прибытия запоздалого фрагмента приведён лишь для иллюстрации
иллюстрации, поскольку он избыточен */
Reordering = True;
}
else { /* когда s < NextExp или (MoreFrag==0 AND s < FragSeq#) */
Reordering = True;
}
}
}

```

Рабочая версия кода будет включать проверку доставки всех фрагментов перед последующим использованием статуса Reordered, например, в показателях выборки.

## В.4. Замечания к показателям выборки при фрагментации

Все фрагменты с одним порядковым номером источника имеют общее время отправки источником.

Оценку с нумерацией потока байтов можно упростить, если просто добавлять смещение фрагмента к SourceByte первого пакета (с нулевым смещением фрагмента), учитывая счёт в 8-октетных блоках.

## Приложение С. Отказ от ответственности и лицензия

В отношении документа в целом и любых его частей (включая псевдокод и код С) авторы не дают каких-либо гарантий и не несут ответственности за любой ущерб в результате использования. Авторы предоставляют безотзывное разрешение каждому использовать, изменять и распространять документ любым способом, который не умаляет прав любого другого лица на использование, изменение и распространение документа при условии, что распространяемые производные работы не содержат вводящих в заблуждение сведений об авторах или версии. Производные работы не обязательно лицензировать на тех же условиях.

### Адреса авторов

**Al Morton**  
AT&T Labs  
Room D3 - 3C06  
200 Laurel Ave. South  
Middletown, NJ 07748 USA  
Phone +1 732 420 1571  
E-Mail: [acmorton@att.com](mailto:acmorton@att.com)

**Len Ciavattone**  
AT&T Labs  
Room A2 - 4G06  
200 Laurel Ave. South  
Middletown, NJ 07748 USA  
Phone +1 732 420 1239  
E-Mail: [lencia@att.com](mailto:lencia@att.com)

**Gomathi Ramachandran**  
AT&T Labs  
Room C4 - 3D22

200 Laurel Ave. South  
Middletown, NJ 07748 USA  
Phone +1 732 420 2353  
E-Mail: [gomathi@att.com](mailto:gomathi@att.com)

**Stanislav Shalunov**  
Internet2  
1000 Oakbrook DR STE 300  
Ann Arbor, MI 48104  
Phone: +1 734 995 7060  
E-Mail: [shalunov@internet2.edu](mailto:shalunov@internet2.edu)

**Jerry Perser**  
Veriwave  
8770 SW Nimbus Ave.  
Suite B  
Beaverton, OR 97008 USA  
Phone: +1 818 338 4112  
E-Mail: [jperser@veriwave.com](mailto:jperser@veriwave.com)

### Перевод на русский язык

Николай Малых

[nmalykh@protokols.ru](mailto:nmalykh@protokols.ru)

**Copyright (C) The IETF Trust (2006).**

К этому документу применимы права, лицензии и ограничения, указанные в BCP 78, и, за исключением указанного там, авторы сохраняют свои права.

Этот документ и содержащаяся в нем информация представлены "как есть" и автор, организация, которую он/она представляет или которая выступает спонсором (если таковой имеется), Internet Society и IETF отказываются от каких-либо гарантий (явных или подразумеваемых), включая (но не ограничиваясь) любые гарантии того, что использование представленной здесь информации не будет нарушать чьих-либо прав, и любые предполагаемые гарантии коммерческого использования или применимости для тех или иных задач.

**Интеллектуальная собственность**

IETF не принимает какой-либо позиции в отношении действительности или объема каких-либо прав интеллектуальной собственности (Intellectual Property Rights или IPR) или иных прав, которые, как может быть заявлено, относятся к реализации или использованию описанной в этом документе технологии, или степени, в которой любая лицензия, по которой права могут или не могут быть доступны, не заявляется также применение каких-либо усилий для определения таких прав. Сведения о процедурах IETF в отношении прав в документах RFC можно найти в BCP 78 и BCP 79.

Копии раскрытия IPR, предоставленные секретариату IETF, и любые гарантии доступности лицензий, а также результаты попыток получить общую лицензию или право на использование таких прав собственности разработчиками или пользователями этой спецификации, можно получить из сетевого репозитория IETF IPR по ссылке <http://www.ietf.org/ipr>.

IETF предлагает любой заинтересованной стороне обратить внимание на авторские права, патенты или использование патентов, а также иные права собственности, которые могут потребоваться для реализации этого стандарта. Информацию следует направлять в IETF по адресу [ietf-ipr@ietf.org](mailto:ietf-ipr@ietf.org).

**Подтверждение**

Финансирование функций RFC Editor обеспечено Internet Society.