

Network Working Group  
Request for Comments: 2328  
STD: 54  
Obsoletes: 2178  
Category: Standards Track

J. Moy  
Ascend Communications, Inc.  
April 1998

## Протокол OSPF версии 2

OSPF Version 2

### Статус документа

Этот документ содержит спецификацию протокола для сообщества Internet и служит запросом для дальнейшего обсуждения и предложений в целях совершенствования протокола. Точные сведения о статусе документа можно найти в Internet Official Protocol Standards ([STD 1](#)). Документ может распространяться свободно.

### Авторские права

Copyright (C) The Internet Society (1998). All Rights Reserved.

### Тезисы

Этот документ описывает вторую версию OSPF – протокола маршрутизации на основе состояний каналов (link-state routing protocol). Протокол предназначен для использования внутри автономных систем (AS). Все OSPF-маршрутизаторы поддерживают идентичные базы данных с описанием топологии AS. На основании данных из базы рассчитывается маршрут путем конструирования дерева кратчайшего пути.

При изменении топологии OSPF быстро пересчитывает маршруты и не порождает значительного трафика при обновлении маршрутной информации. Протокол OSPF обеспечивает поддержку множества путей с одинаковой стоимостью (equal-cost multipath). Поддерживается «маршрутизация областей» (area routing), обеспечивающая дополнительный уровень защиты маршрутизации и снижение служебного трафика при обмене маршрутной информацией. Кроме того, обмен маршрутной информацией осуществляется с использованием средств аутентификации.

Различия между этим документом и RFC 2178 рассмотрены в Приложении G. Все изменения протокола обеспечивают обратную совместимость.

Реализации описываемого протокола совместимы (интероперабельны) с реализациями RFC 2178, RFC 1583 и RFC 1247.

Комментарии к данному документу направляйте по адресу [ospf@gated.cornell.edu](mailto:ospf@gated.cornell.edu).

### Оглавление

Статус документа.....	1
Авторские права.....	1
Тезисы.....	1
1. Введение.....	3
1.1. Обзор протокола.....	3
1.2. Терминология.....	4
1.3. Краткая история маршрутизации на базе состояния каналов.....	5
1.4. Структура документа.....	5
1.5. Благодарности.....	6
2. База данных Link-state: структура и расчеты.....	6
2.1. Представление маршрутизаторов и сетей.....	6
2.1.1. Представление сетей без широковещания.....	7
2.1.2. Пример базы данных о каналах.....	7
2.2. Дерево кратчайших путей.....	9
2.3. Внешние данные.....	9
2.4. Множество равноценных путей.....	10
3. Деление AS на области.....	10
3.1. Магистраль AS.....	10
3.2. Маршрутизация между областями.....	10
3.3. Классификация маршрутизаторов.....	11
3.4. Пример конфигурации областей.....	12
3.5. Поддержка подсетей IP.....	13
3.6. Поддержка «тупиковых» областей.....	14
3.7. Разделение областей.....	14
4. Основные функции.....	14
4.1. Междоменная маршрутизация.....	15
4.2. Внешние маршруты AS.....	15
4.3. Пакеты протокола маршрутизации.....	15
4.4. Основные требования к реализации.....	16
4.5. Дополнительные возможности OSPF.....	17
5. Структуры данных протокола.....	17
6. Структура данных области.....	18
7. Организация отношений смежности.....	19

7.1. Протокол Hello .....	19
7.2. Синхронизация баз данных .....	19
7.3. Выделенный маршрутизатор DR .....	20
7.4. Резервный маршрутизатор DR .....	20
7.5. Граф смежности .....	21
8. Обработка пакетов протокола .....	21
8.1. Передача пакетов .....	21
8.2. Прием пакетов протокола .....	22
9. Структура данных интерфейса .....	23
9.1. Состояния интерфейса .....	24
9.2. События, изменяющие состояние интерфейса .....	25
9.3. Машина состояний интерфейса .....	25
9.4. Назначение DR .....	26
9.5. Передача пакетов Hello .....	27
9.5.1. Передача пакетов Hello в сети NBMA .....	27
10. Структура данных Neighbor .....	28
10.1. Состояния соседей .....	29
10.2. События, меняющие состояние соседа .....	30
10.3. Машина состояний Neighbor .....	30
10.4. Когда организуется смежность .....	32
10.5. Прием пакетов Hello .....	32
10.6. Прием пакетов DD .....	33
10.7. Прием пакетов Link State Request .....	34
10.8. Передача пакетов DD .....	34
10.9. Передача пакетов Link State Request .....	35
10.10. Пример .....	35
11. Структура таблицы маршрутов .....	35
11.1. Просмотр таблицы маршрутов .....	36
11.2. Пример таблицы маршрутизации без областей .....	37
11.3. Пример таблицы маршрутизации с областями .....	37
12. Анонсы состояния каналов (LSA) .....	38
12.1. Заголовок LSA .....	38
12.1.1. Возраст LS .....	38
12.1.2. Опции .....	39
12.1.3. Тип LS .....	39
12.1.4. Link State ID .....	39
12.1.5. Анонсирующий маршрутизатор .....	39
12.1.6. Порядковый номер LS .....	39
12.1.7. Контрольная сумма LS .....	40
12.2. База данных link-state .....	40
12.3. Представление TOS .....	40
12.4. Генерация LSA .....	41
12.4.1. LSA для маршрутизатора .....	42
12.4.1.1. Описание интерфейсов point-to-point .....	42
12.4.1.2. Описание интерфейсов в ширококестельные и NBMA-сети .....	42
12.4.1.3. Описание виртуальных соединений .....	43
12.4.1.4. Описание интерфейсов Point-to-MultiPoint .....	43
12.4.1.5. Примеры router-LSA .....	43
12.4.2. LSA для сети .....	44
12.4.2.1. Примеры network-LSA .....	44
12.4.3. Summary-LSA .....	44
12.4.3.1. Генерация summary-LSA в тупиковые области .....	45
12.4.3.2. Примеры summary-LSA .....	45
12.4.4. AS-external-LSA .....	46
12.4.4.1. Примеры AS-external-LSA .....	46
13. Процедура лавинной рассылки (Flooding) .....	47
13.1. Сравнение возраста LSA .....	47
13.2. Инсталляция LSA в базу данных .....	48
13.3. Следующий этап лавинной рассылки .....	48
13.4. Прием собственных (self-originated) LSA .....	49
13.5. Передача LSA .....	49
13.6. Повторная передача LSA .....	50
13.7. Подтверждение приема состояния канала .....	50
14. Старение базы данных LS .....	50
14.1. Принудительное старение LSA .....	51
15. Виртуальные соединения .....	51
16. Расчет таблицы маршрутизации .....	52
16.1. Расчет кратчайшего пути для области .....	52
16.1.1. Расчет next hop .....	54
16.2. Расчет междоменных маршрутов .....	54
16.3. Просмотр summary-LSA от транзитных областей .....	55
16.4. Расчет внешних маршрутов AS .....	56
16.4.1. Предпочтения для внешнего пути .....	56
16.5. Нарастающие обновления – summary-LSA .....	56
16.6. Нарастающие обновления – AS-external-LSA .....	57
16.7. События, генерируемые при изменении таблицы маршрутов .....	57
16.8. Множество равноценных путей .....	57
Литература .....	57

A. Форматы данных OSPF .....	59
A.1 Инкапсуляция пакетов OSPF .....	59
A.2 Поле Options .....	59
A.3 Форматы пакетов OSPF .....	59
A.3.1 Заголовок пакета OSPF .....	60
A.3.2 Пакет Hello .....	60
A.3.3 Пакет Database Description .....	61
A.3.4 Пакет Link State Request .....	61
A.3.5 Пакет Link State Update .....	61
A.3.6 Пакет Link State Acknowledgment .....	62
A.4 Форматы LSA .....	62
A.4.1 Заголовок LSA .....	62
A.4.2 Router-LSA .....	63
A.4.3 Network-LSA .....	64
A.4.4 Summary-LSA .....	64
A.4.5 AS-external-LSA .....	64
B. Архитектурные константы .....	65
C. Настраиваемые константы .....	66
C.1 Глобальные параметры .....	66
C.2 Параметры области .....	66
C.3 Параметры интерфейса маршрутизатора .....	66
C.4 Параметры виртуального канала .....	67
C.5 Параметры сетей NBMA .....	68
C.6 Параметры сетей Point-to-MultiPoint .....	68
C.7 Параметры Host route .....	68
D. Аутентификация .....	68
D.1 Null-аутентификация .....	68
D.2 Парольная аутентификация .....	68
D.3 Криптографическая аутентификация .....	69
D.4 Генерация сообщений .....	69
D.4.1 Генерация Null-аутентификации .....	69
D.4.2 Генерация простой парольной аутентификации .....	70
D.4.3 Генерация криптографической аутентификации .....	70
D.5 Верификация сообщений .....	70
D.5.1 Проверка Null-аутентификации .....	70
D.5.2 Проверка простой парольной аутентификации .....	70
D.5.3 Проверка криптографической аутентификации .....	70
E. Алгоритм установки Link State ID .....	71
F. Множество интерфейсов в одну сеть/подсеть .....	71
G. Отличия от RFC 2178 .....	72
G.1 Изменение лавинной рассылки .....	72
G.2 Смена предпочтений для внешнего пути .....	72
G.3 Неполное разрешение виртуальных next hop .....	72
G.4 Просмотр таблицы маршрутизации .....	72
Вопросы безопасности .....	72
Адреса авторов .....	72
Полное заявление авторских прав .....	73

## 1. Введение

Этот документ содержит спецификацию протокола OSPF (Open Shortest Path First – по самому короткому пути), служащего для маршрутизации трафика TCP/IP. Протокол OSPF относится к числу внутренних протоколов маршрутизации (Interior Gateway Protocol или IGP) – это означает, что маршрутная информация распространяется между маршрутизаторами одной автономной системы AS. Протокол OSPF работает на основе технологии SPF (link-state – по состоянию канала) в отличие от алгоритмов Bellman-Ford, используемых традиционными протоколами маршрутизации TCP/IP.

Протокол OSPF подготовлен одноименной рабочей группой IETF и предназначен для использования в средах TCP/IP. Протокол включает явную поддержку CIDR и установки меток (tagging) при использовании внешней маршрутной информации. OSPF использует аутентификацию и групповую адресацию (IP multicast) при обмене маршрутными сообщениями. Кроме того, при разработке протокола были приложены значительные усилия по ускорению обработки топологических изменений в сети и снижению уровня служебного трафика.

### 1.1. Обзор протокола

OSPF обеспечивает маршрутизацию пакетов IP исключительно на основе IP-адресов получателей, определенных из заголовка пакетов IP. Пакеты IP маршрутизируются без их изменения (as is), т. е., не используется инкапсуляция в какие-то иные пакеты. OSPF является динамическим протоколом маршрутизации, обеспечивающим быстрое обнаружение топологических изменений в AS (например, сбой маршрутизаторов или каналов) и расчет новых беспетлевых (loop-free) маршрутов. Период схождения (convergence) – расчет нового маршрута – достаточно короток и уровень служебного трафика невелик.

В протоколах на основе состояния каналов (link-state) каждый маршрутизатор поддерживает базу данных с описанием топологии AS. Эти базы называют базами данных о состоянии каналов<sup>1</sup> (link-state database). Базы данных всех маршрутизаторов идентичны. Каждый элемент базы данных представляет собой локальное состояние отдельного маршрутизатора (например, поддерживаемые интерфейсы или доступные соседи). Маршрутизаторы распространяют информацию о своем локальном состоянии путем лавинной маршрутизации (flooding).

<sup>1</sup> Для краткости в переводе используется термин «база данных о каналах» или «база каналов». *Прим. перев.*

Все маршрутизаторы работают в параллель, используя одинаковый алгоритм. На основе базы каналов каждый маршрутизатор строит дерево кратчайших путей, корнем которого является сам маршрутизатор. Это дерево содержит маршруты ко всем адресатам внутри AS. Маршрутная информация внешнего происхождения представляется как листья дерева.

При наличии нескольких путей равной стоимости к одному адресату, трафик поровну распределяется между всеми маршрутами. Стоимость маршрута описывается безразмерной метрикой, представляемой в виде одного числа.

OSPF позволяет группировать сети – такие группы называют областями (area). Топология области невидима для остальной части AS. Такое сокрытие (избыточной) информации позволяет существенно снизить уровень служебного трафика. Кроме того, маршрутизация внутри области определяется исключительно внутренней топологией этой области, что обеспечивает защиту областей от использования некорректной маршрутной информации. Понятие области является обобщением подсетей IP.

OSPF обеспечивает возможность гибкой настройки подсетей IP. Каждый маршрут в OSPF распространяется с указанием адресата и маски подсети. Две разных подсети одной сети IP могут иметь различные размеры (т. е., разные маски) – для обозначения этого обычно используется термин *variable length subnetting* (переменный размер подсетей). Пакеты маршрутизируются по пути с наилучшим (т. е., самым длинным или более конкретным) соответствием. Маршруты к хостам рассматриваются как пути в подсети с маской из одних единиц (all ones или 0xffffffff<sup>2</sup>).

Весь обмен информацией OSPF осуществляется с использованием аутентификации. Это означает, что в маршрутизации внутри AS могут участвовать только уполномоченные маршрутизаторы. Могут использоваться различные схемы аутентификации; в частности, допустимо применение различных схем для каждой подсети IP.

Внешние маршрутные данные (т. е., маршруты, полученные от протоколов внешних шлюзов EGP - например, BGP - [Ref23]) анонсируются через AS. Такие данные сохраняются отдельно от данных OSPF о состоянии каналов. Каждый внешний маршрут может также быть помечен (tagged) анонсирующим его маршрутизатором – это дает возможность передачи дополнительной информации между маршрутизаторами на границе AS.

## 1.2. Терминология

В этом параграфе приведены определения терминов, имеющих специфическое толкование в контексте протокола OSPF и используемых в тексте. Читателям, не знакомым со стеком протоколов Internet рекомендуется обратиться к работе [Ref13], содержащей вводную информацию по протоколу IP.

### *Router (маршрутизатор)*

Устройство сетевого уровня для пересылки пакетов IP. В литературе прошлых лет для обозначения маршрутизаторов часто использовался термин gateway (шлюз).

### *Autonomous System (автономная система)*

Группа маршрутизаторов, обменивающихся маршрутной информацией на основе общего протокола. Используется также сокращение AS.

### *Interior Gateway Protocol (протокол внутренней маршрутизации)*

Протокол маршрутизации, используемый в пределах одной автономной системы. Для обозначения таких протоколов часто используется аббревиатура IGP. Каждая AS имеет один протокол IGP. В разных автономных системах могут использоваться различные протоколы IGP.

### *Router ID (идентификатор маршрутизатора)*

32-битовое целое число, связываемое с каждым маршрутизатором, использующим протокол OSPF. Этот идентификатор является уникальным в масштабе AS.

### *Network (сеть)*

В контексте документа термин сеть может относиться к сети/подсети/«сети сетей» IP (network/subnet/supernet). Одна физическая сеть может включать множество сетей/подсетей IP – мы будем рассматривать их как разные сети. Физические сети «точка-точка» (point-to-point) являются исключением – такая сеть рассматривается как единое целое, независимо от наличия в ней отдельных сетей/подсетей IP.

### *Network mask (маска сети)*

32-битовое число, показывающее диапазон IP-адресов, относящихся к одной сети/подсети/сети сетей IP. В данной спецификации маски указываются шестнадцатеричными числами.

Например, маска сети класса C имеет значение 0xfffff00. Обычно для записи такой маски используют форму 255.255.255.0.

### *Point-to-point networks (сеть на базе соединений точка-точка)*

Сеть, соединяющая между собой 2 маршрутизатора. Примером таких сетей являются сети на основе каналов 56 кбит/с.

### *Broadcast networks (широковещательные сети)*

Сети, поддерживающие множество (более 2) подключенных маршрутизаторов с возможностью адресации одного сообщения всем подключенным маршрутизаторам – широковещательная передача (broadcast).

Соседние маршрутизаторы в таких сетях обнаруживаются динамически с использованием протокола OSPF Hello. Протокол Hello использует преимущества широковещательной передачи. Протокол OSPF может также использовать групповую адресацию (multicast), если она поддерживается. Предполагается, что каждая пара маршрутизаторов в широковещательной сети может обмениваться данными напрямую. Примером широковещательных сетей может служить Ethernet.

### *Non-broadcast networks (сети без широковещания)*

Сети, поддерживающие множество (более 2) подключенных маршрутизаторов, но не имеющие возможностей широковещательной передачи. Для обнаружения соседних маршрутизаторов также используется протокол OSPF Hello, однако, в силу отсутствия широковещания, для обнаружения соседей требуются некоторые настройки. В сетях без широковещания пакеты OSPF, обычно передаваемые с использованием групповой адресации, должны адресоваться непосредственно соседним маршрутизаторам. Примером сетей без широковещания могут служить сети X.25.

Протокол OSPF в сетях без широковещания может работать в двух режимах. Первый режим называется NBMA (non-broadcast multi-access – множественный доступ без широковещания) имитирует работу OSPF в широковещательных сетях. Второй режим называется Point-to-MultiPoint (один со многими), в этом случае сеть без широковещания трактуется как множество соединений «точка-точка»<sup>3</sup>. Сети без широковещания мы будем делить на сети NBMA и Point-to-MultiPoint в зависимости от режима работы OSPF.

### *Interface (интерфейс)*

Соединение между маршрутизатором и одной из подключенных к нему сетей. С интерфейсом связана информация о его состоянии, получаемая от протоколов нижележащих уровней и самого протокола маршрутизации. Сетевой интерфейс имеет

<sup>2</sup> 255.255.255.255. Прим. перев.

<sup>3</sup> Далее для краткости будем называть такие соединения парными. Прим. перев.

адрес IP и маску, если этот интерфейс не относится к безадресным (unnumbered) соединениям point-to-point. Применительно к интерфейсу может также употребляться термин соединение, канал (link).

#### **Neighboring routers (соседние маршрутизаторы)**

Два маршрутизатора, имеющие интерфейсы в общую сеть. Отношения соседства поддерживаются и детектируются (обычно автоматически) с помощью протокола OSPF Hello.

#### **Adjacency (смежность)**

Отношения между соседними маршрутизаторами, организуемые в целях обмен маршрутной информацией. Не всякая пара соседних маршрутизаторов поддерживает отношения смежности.

#### **Link state advertisement (анонс состояния канала)**

Единица данных, описывающая состояние маршрутизатора или сети. Для маршрутизатора анонс включает состояние интерфейсов и отношения смежности. Каждый анонс рассылается с использованием лавинной маршрутизации в масштабах домена маршрутизации. Анонсы состояний всех маршрутизаторов и сетей формируют базу данных о каналах, используемую протоколом. В настоящем документе для обозначения анонсов состояния канала используется аббревиатура LSA.

#### **Hello Protocol (протокол приветствия)**

Часть протокола OSPF, используемая для организации и поддержки отношений соседства между маршрутизаторами. В широковещательных сетях протокол Hello может служить для динамического обнаружения соседних маршрутизаторов.

#### **Flooding (лавинная маршрутизация)**

Часть протокола OSPF, отвечающая за распространение и синхронизацию базы данных о каналах между маршрутизаторами OSPF.

#### **Designated Router (указанный маршрутизатор)**

Каждая широковещательная или NBMA-сеть, в которой есть не менее двух маршрутизаторов, имеет среди них выделенный - Designated Router (DR). Выделенный маршрутизатор генерирует анонсы LSA для сети и выполняет другие специальные действия для обеспечения работы протокола. Маршрутизатор DR задается протоколом Hello.

Концепция выделенного маршрутизатора DR позволяет уменьшить число смежных пар, требуемых для широковещательных сетей и сетей NBMA, что обеспечивает снижение уровня служебного трафика и уменьшение базы данных о каналах.

#### **Lower-level protocols (протоколы нижележащих уровней)**

Протоколы нижележащих уровней, предоставляющих свои услуги протоколу IP и OSPF. Примерами таких протоколов могут служить X.25 или протоколы канального уровня Ethernet.

## **1.3. Краткая история маршрутизации на базе состояния каналов**

OSPF относится к числу протоколов маршрутизации на основе данных о состоянии каналов. Такие протоколы также называют протоколами на базе SPF или протоколами с распределенной базой данных (distributed-database protocol). В этом параграфе приведено краткое описание технологии link-state в контексте ее влияния на протокол OSPF.

Первый протокол маршрутизации по состоянию каналов был разработан для использования в сети ARPANET, работавшей на основе коммутации пакетов. Этот протокол описан в работе [Ref3]. Протокол послужил отправной точкой для разработки всех остальных протоколов link-state. Гомогенная среда ARPANET, содержащая однотипные коммутаторы, соединенные синхронными последовательными каналами, существенно упростила разработку и реализацию первого варианта протокола.

В работе [Ref4] был предложен измененный вариант этого протокола. Изменения позволили повысить устойчивость протокола маршрутизации к сбоям – к таким изменениям относятся, прежде всего, добавление контрольных сумм LSA, позволяющее детектировать повреждения баз данных. В этой работе также были предложены способы снижения уровня служебного трафика протокола link-state за счет использования механизма, позволяющего увеличить на порядок интервал между последовательными передачами LSA.

Алгоритм link-state был также предложен для использования в качестве протокола маршрутизации ISO IS-IS, описанного в работе [Ref2]. Этот протокол обеспечивал методы снижения трафика при работе в широковещательных сетях за счет введения выделенного маршрутизатора DR для каждой широковещательной сети. Этот маршрутизатор обеспечивает генерацию LSA для своей сети.

Рабочая группа IETF OSPF продолжила разработки в этом направлении, создав протокол OSPF. Концепция выделенного маршрутизатора DR была существенно доработана, что позволило добиться дополнительного снижения служебного трафика. Дополнительное снижение расхода полосы было обеспечено также за счет использования групповой адресации. Была разработана схема областей маршрутизации (area routing), обеспечивающая защиту информации, ее сокрытие и сокращение объема. И, наконец, алгоритмы были приспособлены для использования в межсетевой среде TCP/IP.

## **1.4. Структура документа**

В трех первых разделах документа приведен общий обзор возможностей протокола и его функций. Главы 4 - 16 содержат детальное описание различных механизмов работы протокола. Формат пакетов, константы и элементы настройки описаны в приложениях.

Метки типа HelloInterval, встречающиеся в тексте, обозначают константы протокола, которые могут иметь фиксированные или настраиваемые значения. Архитектурные константы описаны в Приложении В, а настраиваемые константы – в Приложении С.

При описании протокола используется множество структур данных. Такое представление позволяет дать более точные объяснения. При реализации протокола требуется точное соблюдение функциональных требований, но сохранение приведенных в спецификации структур данных не является обязательным.

## **1.5. Благодарности**

Автор выражает свою признательность Ran Atkinson, Fred Baker, Jeffrey Burgan, Rob Coltun, Dino Farinacci, Vince Fuller, Phanindra Jujjavarapu, Milo Medin, Tom Pusateri, Kannan Varadhan, Zhaohui Zhang и остальным участникам рабочей группы OSPF за их идеи и поддержку проекта.

Интерфейс OSPF Point-to-MultiPoint основан на результатах работы Fred Baker.

Средства криптографической аутентификации OSPF Cryptographic Authentication разработали Fred Baker и Ran Atkinson.

## **2. База данных Link-state: структура и расчеты**

В следующих параграфах описана организация базы данных о каналах протокола OSPF и расчеты маршрутов, выполняемые для заполнения таблиц маршрутизации.



## 2.1. Представление маршрутизаторов и сетей

База каналов AS описывается направленным графом. Вершины графов показывают маршрутизаторы и сети. Ребро графа соединяет два маршрутизатора, если между ними существует физическая сеть point-to-point. Ребро графа соединяет два маршрутизатора с сетью, если маршрутизатор имеет интерфейс в данную сеть. Сеть может быть транзитной или оконечной (stub). Транзитные сети могут также передавать трафик, который не связан с данной сетью, т. е., отправитель и получатель принадлежат другим сетям. Транзитные сети представляются вершинами графов, имеющими входящее и исходящее ребро. Вершины графов для оконечных сетей имеют только входящее ребро.

Соседство каждого сетевого узла на графе зависит от типа сети (point-to-point, широковещательная, NBMA или Point-to-MultiPoint) и числа маршрутизаторов, имеющих интерфейс в данную сеть. На рисунке 1а показано три варианта. Прямоугольники представляют маршрутизаторы, круги и эллипсы – сети. В обозначениях маршрутизаторов используется префикс RT, для сетей – префикс N. Интерфейсы маршрутизаторов именуются с префиксом I. Линии между маршрутизаторами показывают сети point-to-point. В левой части рисунка показаны сети с подключенными к ним маршрутизаторами, а справа приведено представление в виде графов.

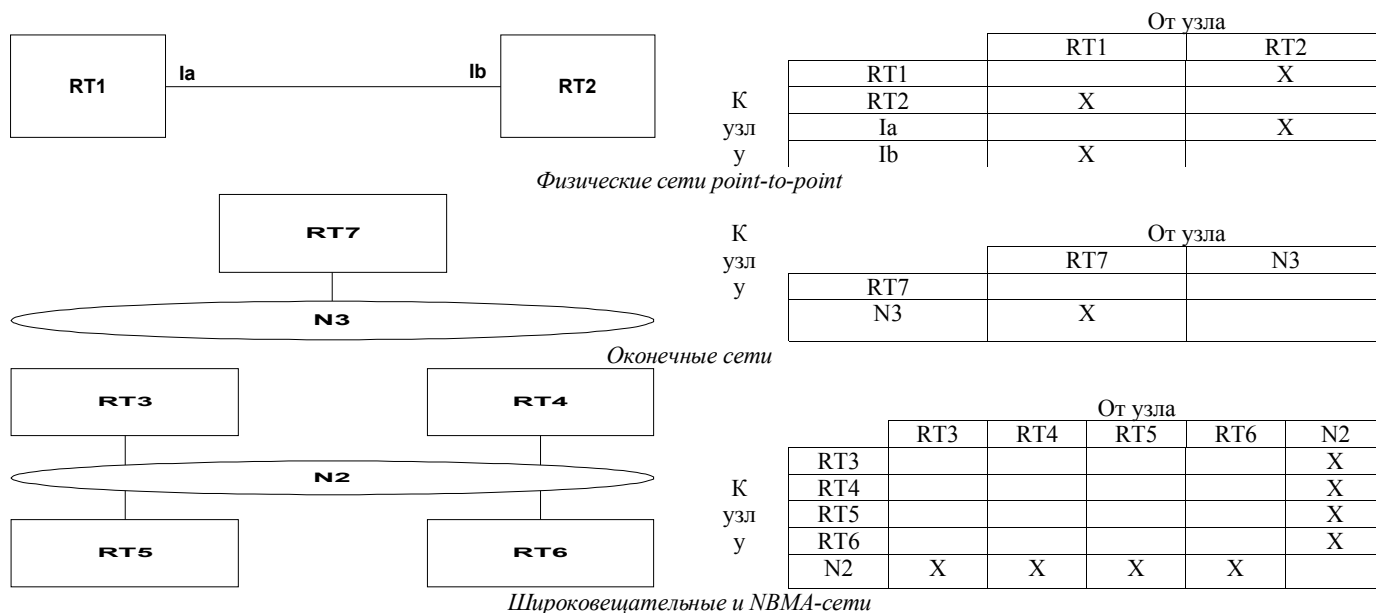


Рисунок 1а. Компоненты представления сетей.

Сети и маршрутизаторы представлены вершинами графа. Ребро графа соединяет вершину A с вершиной B, если на пересечении колонки A и строки B поставлен знак X.

В верхней части рисунка 1а показаны два маршрутизатора, соединенные каналом «точка-точка». В результирующем графе базы данных вершины двух маршрутизаторов напрямую соединены парой ребер (по одному ребру в каждом направлении). Интерфейсы сетей point-to-point могут работать без IP-адресов. Когда интерфейсу присваивается адрес, он представляется как оконечное соединение (stub link) и каждый маршрутизатор анонсирует такое соединение по адресам других интерфейсов маршрутизатора. В дополнение к адресу с парным соединением может быть связана подсеть IP. В таких случаях оба маршрутизатора анонсируют stub-соединение с подсетью IP взамен анонса IP-адреса интерфейса.

В средней части рисунка 1а показана сеть с единственным подключенным маршрутизатором (т. е., оконечная сеть - stub network). В этом случае сеть появляется как окончание stub-соединения в графе базы каналов.

Когда множество маршрутизаторов подключено к широковещательной сети, граф базы каналов показывает двунаправленные соединения каждого маршрутизатора к сетевой вершине. Пример такой ситуации показан в нижней части рисунка 1а.

Каждая сеть (оконечная или транзитная) на графе имеет IP-адрес и связанную с ним маску подсети. Маска показывает число узлов в сети. Хосты, напрямую подключенные к маршрутизатору (в этом случае говорят о маршруте к хосту - host route) появляются на графе как оконечные сети. Маска подсети для маршрутов к хостам всегда имеет значение 0xffffffff, показывающее, что в подсети присутствует единственный узел.

### 2.1.1. Представление сетей без широковещания

Как было отмечено выше, протокол OSPF может работать в сетях без широковещания в одном из двух режимов - NBMA или Point-to-MultiPoint. Выбор режима определяет работу протокола Hello и лавинной маршрутизации в сетях без широковещания, а также способ представления сети в базе данных о каналах.

В режиме NBMA протокол OSPF эмулирует работу в широковещательной сети – для сети NBMA задается выделенный маршрутизатор DR, который генерирует LSA для сети. Графы для широковещательных сетей и сетей NBMA идентичны (см. рисунок 1а).

Режим NBMA является наиболее эффективным вариантом использования OSPF в сетях без широковещания как с точки зрения размера базы данных о каналах, так и по объему служебного трафика. Однако этому режиму присущи серьезные ограничения – он требует подключения всех маршрутизаторов к сети NBMA для того, чтобы стал возможен прямой обмен информацией между маршрутизаторами. Это ограничение можно обойти для некоторых сетей без широковещания (например, для подсетей ATM, использующих SVC). Однако в других случаях, такое ограничение может оказаться очень существенным (например, для сетей Frame Relay, в которых поддерживаются только постоянные соединения PVC). В сетях без широковещания, где не все маршрутизаторы могут быть связаны напрямую, можно разбить сеть на логические подсети, в каждой из которых маршрутизаторы смогут взаимодействовать напрямую и тогда каждая отдельная подсеть может функционировать как сеть NBMA (см. [Ref15]). Однако такое решение требует от администратора значительных усилий и, к тому же, на этом пути легко допустить ошибки в настройке. Возможно, что для таких сетей будет лучше использовать режим Point-to-Multipoint.

В режиме Point-to-MultiPoint протокол OSPF трактует соединения между маршрутизаторами через сеть без широковещания просто как каналы «точка-точка». В сети не задается маршрутизатор DR и для сети не генерируются LSA. Фактически, вершина для сетей Point-to-MultiPoint просто не появляется на графе базы каналов.

На рисунке 1b показано представление базы данных для сети Point-to-MultiPoint. Слева приведена схема сети Point-to-MultiPoint. Предполагается, что все маршрутизаторы могут взаимодействовать напрямую, за исключением маршрутизаторов RT4 и RT5. Интерфейсы I3 - I6 идентифицируются по их IP-адресам в сети Point-to-MultiPoint. В графическом представлении базы данных маршрутизаторы, способные взаимодействовать напрямую через сеть Point-to-MultiPoint, соединены двусторонними ребрами и каждый маршрутизатор также имеет stub-соединение со своим интерфейсом по IP-адресу (в отличие от сети point-to-point, показанной на рисунке 1a).

В некоторых сетях без широковещания использование режима Point-to-MultiPoint и протоколов канального уровня типа Inverse ARP (см. [Ref14]) позволяет автоматически детектировать соседей OSPF, несмотря на отсутствие широковещания.

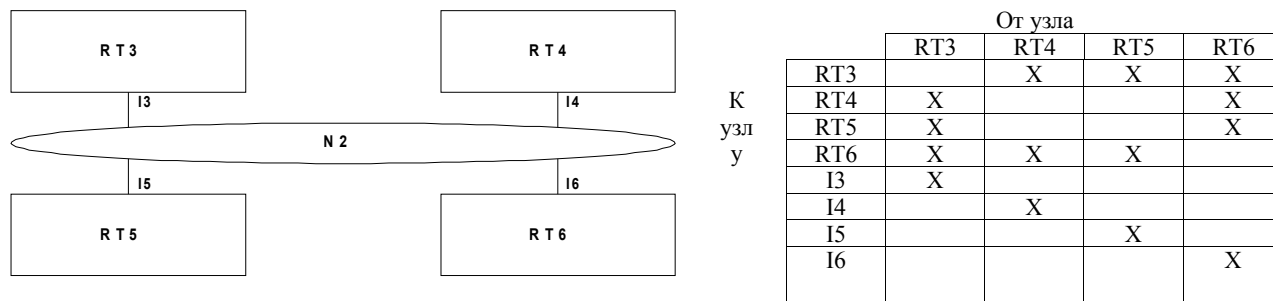


Рисунок 1b: Компоненты представления сетей Point-to-MultiPoint

Все маршрутизаторы (за исключением RT4 и RT5) могут взаимодействовать напрямую через сеть N2. Интерфейсы I3- I6 используют IP-адреса.

### 2.1.2. Пример базы данных о каналах

На рисунке 2 показан пример автономной системы. Прямоугольник с меткой H1 показывает хост, имеющий SLIP-соединение с маршрутизатором RT12. Маршрутизатор RT12, следовательно, анонсирует маршрут к хосту. Линии между маршрутизаторами показывают физические сети point-to-point. Единственная сеть point-to-point, в которой используются адреса для интерфейсов, соединяет маршрутизаторы RT6 и RT10. Маршрутизаторы RT5 и RT7 имеют BGP-соединение с другими AS. Набор полученных по BGP маршрутов показан для обоих маршрутизаторов.

Стоимость связывается с выходом каждого интерфейса в маршрутизаторе. Значение стоимости задает администратор сети. Чем меньше стоимость для интерфейса, тем с большей вероятностью этот интерфейс будет использоваться для пересылки трафика. Значение стоимости связывается также с полученной извне маршрутной информацией (например, данными BGP).

Направленный граф для сети, показанной на рисунке 2, приведен в таблице 1. На рисунке цифры показывают стоимость, связанную с интерфейсами маршрутизаторов. Если стоимость не указана, предполагается нулевое значение. Отметим, что соединения, выходящие из сетей, всегда имеют нулевую стоимость. Также подчеркнем, что полученные извне маршрутные данные показаны на графе как окончные (stub).

База данных о каналах собирается по частям из LSA, генерируемых маршрутизаторами. В связанном графическом представлении соседство каждого маршрутизатора или транзитной сети представлено в одной, отдельной записи LSA. В таблице 2 эти LSA представлены графически. Маршрутизатор RT12 имеет интерфейсы в две широковещательные сети и SLIP-соединение с хостом. Сеть N6 является широковещательной и с ней соединены три маршрутизатора. Стоимость всех соединений из сети N6 с подключенными к этой сети маршрутизаторами равна 0. Отметим, что LSA для сети N6 на самом деле порождается одним из подключенных к этой сети маршрутизаторов, который для этой сети как DR.

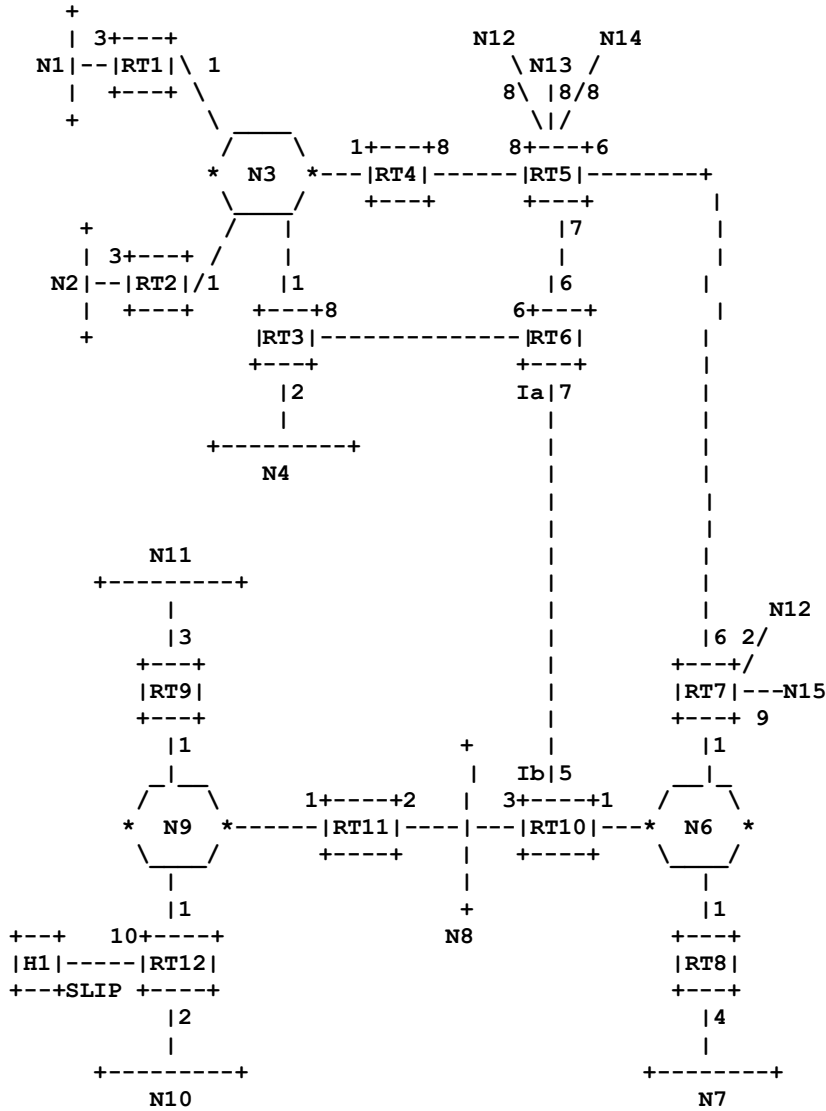


Рисунок 2. Пример автономной системы

Таблица 1 Графическое представление базы данных.

		От узла															
		RT1	RT2	RT3	RT4	RT5	RT6	RT7	RT8	RT9	RT10	RT11	RT12	N3	N6	N8	N9
К у з л у	RT1													0			
	RT2						6							0			
	RT3													0			
	RT4					8								0			
	RT5				8		6	6									
	RT6			8		7					5						
	RT7				6												
	RT8														0		
	RT9														0		0
	RT10						7										
	RT11															0	0
	RT12																0
	N1	3															
	N2		3														
	N3	1	1	1	1												
	N4		2														
	N6						1	1		1							
	N7							4									
	N8									3	2						
	N9								1		1	1					
	N10											2					
	N11							3									
	N12				8		2										
	N13				8												
	N14				8												
	N15						9										
	H1										10						



Таблица 2 Отдельные компоненты базы данных.

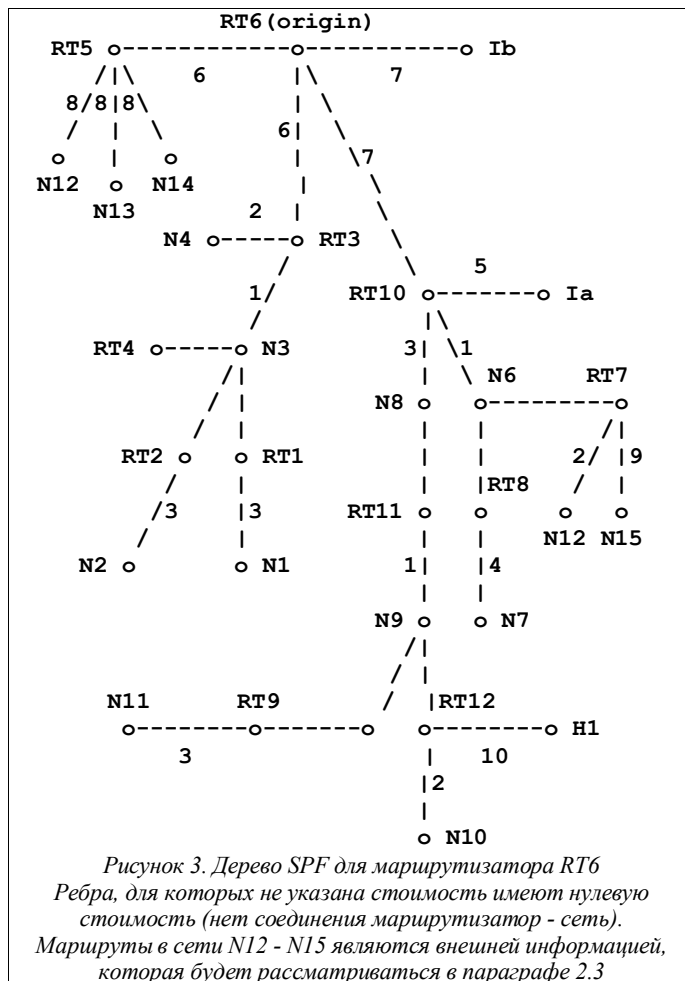
		От узла			
		RT12	N9	N10	H1
К узел у	RT12				
	N9	1			
	N10	2			
	H1	10			

LSA для маршрутизатора RT12

Сети и маршрутизаторы представлены вершинами. Ребро стоимостью  $X$  соединяет вершину  $A$  с вершиной  $B$ , если на пересечении колонки  $A$  и строки  $B$  показан знак  $X$ .

К узл у	От узла			
	RT9	RT11	RT12	N9
	RT9			0
	RT11			0
	RT12			0
	N9			

LSA для сети N9



## 2.2. Дерево кратчайших путей

При отсутствии областей OSPF все маршрутизаторы в AS имеют идентичные базы данных о состоянии каналов; графическое представление этих баз также будет идентичным. Маршрутизатор генерирует свою таблицу маршрутов из графа базы данных, рассчитывая дерево кратчайших путей, корнем которого является сам маршрутизатор. Обычно дерево зависит от маршрутизатора, для которого оно рассчитано. Дерево для маршрутизатора RT6 из приведенного примера показано на рисунке 3.

Дерево включает путь к любой сети или хосту. Однако при пересылке пакетов адресату используется только следующий маршрутизатор (next hop). Отметим также, что рассчитывается лучший путь к каждому маршрутизатору. Для обработки внешних данных отметим next hop и дистанцию для всех маршрутизаторов, анонсирующих внешние маршруты. Полученная в результате таблица маршрутизации для RT6 показана в таблице 3. Отметим, что существует отдельный маршрут для каждого конца адресуемой сети point-to-point (в нашем случае это последовательный канал между RT6 и RT10).

Маршрутизаторы в сети, относящиеся к другим AS (например, N12) показаны пунктирными линиями на дереве кратчайших путей (рисунок 3). Использование этой полученной извне маршрутной информации рассмотрено в следующем параграфе.

### 2.3. Внешние данные

После создания дерева проверяется маршрутная информация, полученная извне. Эта информация может приходиться от других протоколов маршрутизации (например, BGP) или задаваться статически. Принятые по умолчанию маршруты также могут включаться как часть внешних маршрутных данных AS.

Внешняя маршрутная информация рассылается через AS без изменений с использованием лавинной маршрутизации. В нашем примере все маршрутизаторы в AS знают, что RT7 имеет два внешних маршрута с метрикой 2 и 9.

Протокол OSPF поддерживает два типа внешней метрики. Тип 1 использует для метрики такие же единицы, которые служат для обозначения стоимости интерфейсов в OSPF (т. е., метрика определяется состоянием канала). Внешняя метрика типа 2 использует на порядок большие значения и предполагается, что любая метрика типа 2 заведомо превосходит стоимость любого пути внутреннего по отношению к AS.

Использование метрики типа 2 предполагает, что маршрутизация между AS составляет основную часть стоимости пересылки пакетов – это избавляет от необходимости преобразования внешней метрики во внутренние единицы.

В качестве примера обработки внешней метрики типа 1 предположим, что маршрутизаторы RT7 и RT5 на рисунке 2 анонсируют внешнюю метрику этого типа. Для каждого из анонсируемых внешних маршрутов стоимость от маршрутизатора RT6 рассчитывается как сумма анонсированной стоимости внешнего маршрута и дистанции от RT6 до анонсирующего маршрутизатора. Когда два маршрутизатора анонсируют путь к одному получателю, RT6 выбирает тот маршрутизатор, через который общая стоимость доставки будет меньше. После этого RT6 устанавливает в качестве значения next hop на пути к внешнему адресату следующий маршрутизатор, который использовался для выбора анонсированного пути.

На рисунке 2 оба маршрутизатора RT5 и RT7 анонсируют внешний маршрут в сеть N12. Маршрутизатор RT7 является предпочтительным, поскольку он анонсирует для N12 дистанцию 10 (8+2), а маршрутизатор RT5 - 14 (6+8). В таблице 4 показаны записи, добавляемые в таблицу маршрутизации при проверке внешних маршрутов.

Обработка внешней метрики типа 2 проще. Выбирается граничный маршрутизатор AS, анонсирующий наименьшую метрику, независимо от внутренней дистанции до граничного маршрутизатора AS. Предположим, что в нашем примере маршрутизаторы RT5 и RT7 анонсируют внешние маршруты типа 2. Тогда весь трафик для сети N12 будет пересылаться маршрутизатору RT7.

Таблица 3 Часть таблицы маршрутизации RT6 с локальными адресатами.

Получатель	Next Hop	Дистанция
N1	RT3	10
N2	RT3	10
N3	RT3	7
N4	RT3	8
Ib	*	7
Ia	RT10	12
N6	RT10	8
N7	RT10	12
N8	RT10	10
N9	RT10	11
N10	RT10	13
N11	RT10	14
H1	RT10	21
RT5	RT5	6
RT7	RT10	8

Таблица 4 Часть таблицы маршрутизации RT6 с внешними маршрутами.

Получатель	Next Hop	Дистанция
N12	RT10	10
N13	RT5	14
N14	RT5	14
N15	RT10	17

на рисунке 2 присутствует дополнительный маршрутизатор RTX, подключенный к сети N6. Предположим также, что RTX не участвует в маршрутизации OSPF, но обменивается информацией BGP с граничным маршрутизатором AS RT7. В этом случае RT7 будет анонсировать все внешние маршруты OSPF для адресатов, которые должны маршрутизироваться через RTX. Если пакеты для таких адресатов всегда будут передаваться через RT7 (анонсирующий маршрутизатор), в некоторых случаях будет появляться ненужная петля. В таких ситуациях протокол OSPF позволяет граничному маршрутизатору AS задать «адрес пересылки» (forwarding address) в его внешнем по отношению к AS анонсе LSA. В приведенном выше примере маршрутизатор RT7 будет указывать IP-адрес RTX как адрес пересылки для всех получателей, пакеты к которым должны маршрутизироваться напрямую в RTX.

Адрес пересылки имеет еще одно применение – он позволяет маршрутизаторам внутри AS работать в качестве маршрутных серверов (route server). Например, маршрутизатор RT6 на рисунке 2 может стать маршрутным сервером, получающим внешние маршрутные данные из статических конфигурационных параметров и внешних протоколов маршрутизации. RT6 будет в этом случае анонсировать себя как граничный маршрутизатор AS и может генерировать анонсы OSPF LSA, внешние по отношению к AS. В каждом таком анонсе LSA маршрутизатор RT6 будет корректно указывать выходную точку AS для рассылки соответствующим получателям путем установки в LSA поля forwarding address.

## 2.4. Множество равноценных путей

В предыдущих параграфах для простоты предполагался единственный маршрут к каждому получателю. На практике к каждому адресату может вести множество равноценных маршрутов и все такие маршруты определяются и используются. Такое расширение не требует концептуального изменения алгоритма и мы отложим рассмотрение вопроса обработки множества равноценных путей до момента более детального описания процесса построения дерева маршрутов.

При наличии множества равноценных путей маршрутизатор может иметь несколько вариантов next hop (следующий маршрутизатор) для каждого адресата.

## 3. Деление AS на области

Протокол OSPF позволяет объединять смежные сети и хосты в группы. Такая группа вместе с маршрутизаторами, имеющими интерфейс в одну (любую) из включенных в группу сетей, называется областью - area. В каждой области работает своя независимая копия алгоритма маршрутизации link-state. Это означает, что каждая область имеет свою базу каналов и соответствующий граф, как было описано в предыдущих параграфах.

Топология области невидима за ее пределами и наоборот, внутренние маршрутизаторы области не знают ничего о топологии за пределами данной области. Такая локализация сведений о топологии позволяет сильно снизить объем служебного трафика протокола по сравнению с ситуацией, когда вся AS является единым доменом link-state.

При использовании областей допущение идентичности баз данных о каналах на всех маршрутизаторах AS перестает быть верным. Маршрутизаторы теперь имеют отдельные базы данных для каждой области, с которой данный маршрутизатор соединен<sup>4</sup>. Два маршрутизатора, относящиеся к одной области имеют для этой области идентичные базы каналов.

Маршрутизация в AS осуществляется на двух уровнях в зависимости от местоположения отправителя и получателя. Если обе стороны находятся в одной области используется внутridoменная маршрутизация (intra-area routing), при расположении отправителя и получателя в разных областях применяется междоменная маршрутизация (inter-area routing). При внутridoменной маршрутизации используются только сведения, полученные из данной области. Такое решение предохраняет область от использования некорректных маршрутных данных. Междоменная маршрутизация рассматривается в параграфе 3.2.

### 3.1. Магистраль AS

Магистраль OSPF (backbone) представляет собой специальную область OSPF Area 0 (ее часто обозначают как Area 0.0.0.0, поскольку в OSPF идентификаторы областей обычно записывают в формате IP-адресов). Магистраль OSPF всегда включает все граничные маршрутизаторы областей. Магистраль отвечает за распространение маршрутной информации между остальными (non-backbone) областями. Магистраль должна обеспечивать смежность (contiguous), однако это не обязательно физическое соседство – магистральные соединения могут организовываться и поддерживаться с помощью настройки виртуальных соединений.

Виртуальное соединение можно настроить между любой парой маршрутизаторов магистрали, имеющих интерфейсы в одну область за пределами магистрали. Виртуальные соединения относятся к магистральной области. Протокол трактует пару маршрутизаторов, соединенных виртуальным каналом, как маршрутизаторы, связанные через безадресную магистральную сеть point-to-point. На графе магистрали два таких маршрутизатора соединяются дугами (arc), для которых стоимость определяется внутridoменным расстоянием между маршрутизаторами. Для служебного трафика, передаваемого через виртуальное соединение, используется только внутridoменная маршрутизация.

### 3.2. Маршрутизация между областями

При маршрутизации пакетов между двумя областями, лежащими за пределами магистрали, используется магистраль AS. Маршрут передачи пакетов можно разбить на три смежных фрагмента – путь внутри области до граничного маршрутизатора области, путь через магистраль между областями отправителя и получателя, путь внутри области от граничного маршрутизатора до адресата. Алгоритм маршрутизации обеспечивает нахождение набора сегментов пути с минимальной стоимостью суммарного маршрута.

Такой подход позволяет представить AS как систему с топологией «звезда», имеющую в центре концентратор-магистраль (backbone hub), к которому подключены все остальные области автономной системы.

<sup>4</sup> Маршрутизаторы, имеющие интерфейсы в несколько областей, называют граничными маршрутизаторами областей - area border router.

Топология магистральной области определяет пути между областями. Для более эффективной организации магистральных путей могут использоваться виртуальные соединения. Это дает администратору некоторые средства контроля для маршрутов передачи междоменного трафика.

Подходящий граничный маршрутизатор для вывода пакетов из области отправителя определяется так же, как выбираются маршрутизаторы, анонсирующие внешние пути. Каждый граничный маршрутизатор в области резюмирует для области стоимость доставки во все внешние по отношению к данной области сети. После расчета для области дерева SPF маршруты ко всем внешним получателям рассчитываются путем проверки резюме от граничных маршрутизаторов области.

### 3.3. Классификация маршрутизаторов

До введения областей единственными маршрутизаторами OSPF, выполняющими специальные функции, были те маршрутизаторы, которые анонсировали внешние маршрутные данные (например, RT5 на рисунке 2). После раздела AS на области OSPF, маршрутизаторы приобретают дополнительную специализацию в соответствии с выполняемыми функциями:

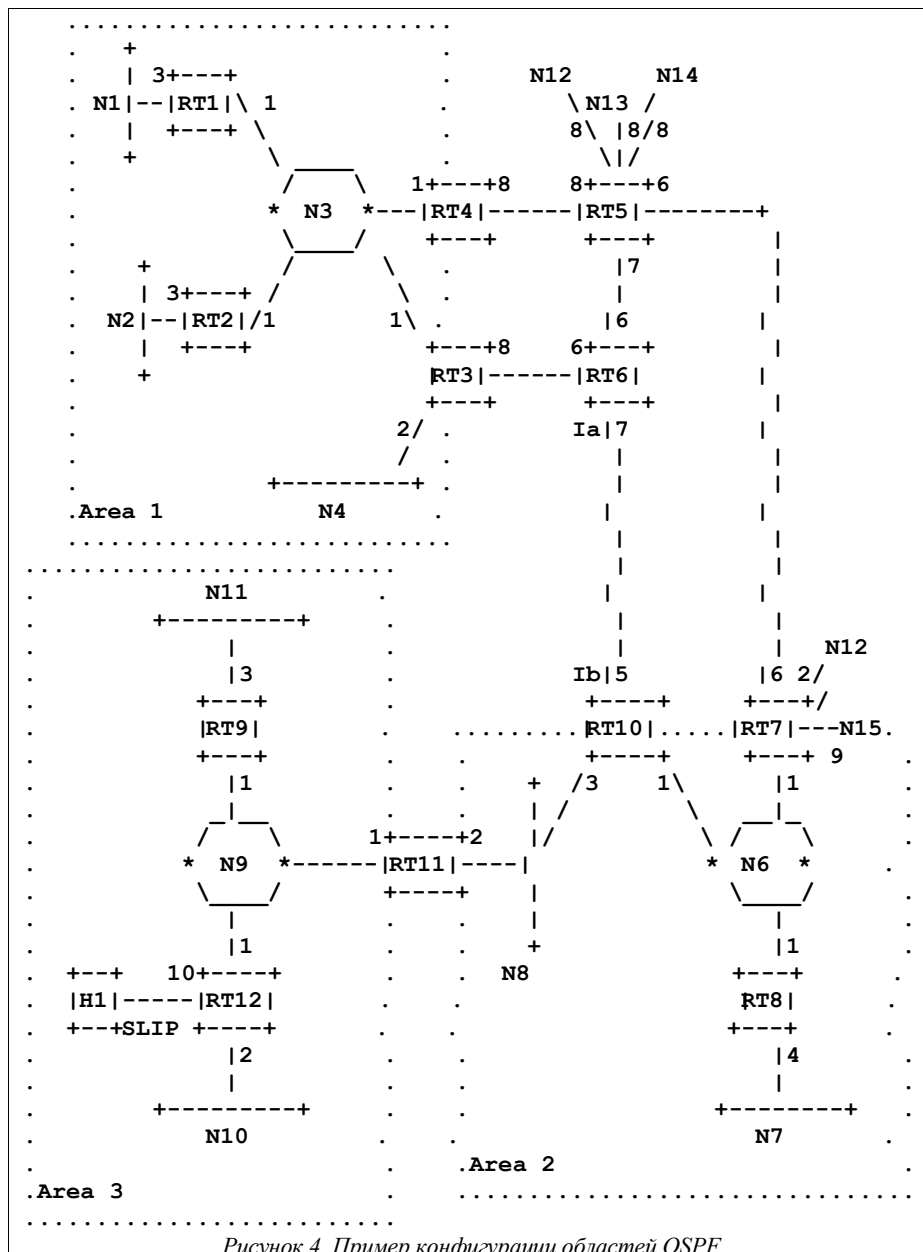


Рисунок 4. Пример конфигурации областей OSPF

данные в автономную систему. Пути ко всем граничным маршрутизаторам AS известны каждому маршрутизатору в автономной системе. Эта классификация никак не связана с предыдущими типами – граничными маршрутизаторами AS могут быть внутренние маршрутизаторы или граничные маршрутизаторы областей, а также магистральные маршрутизаторы.

### 3.4. Пример конфигурации областей

На рисунке 4 показан пример конфигурации областей. Первая область включает сети N1 - N4 и подключенные к ним маршрутизаторы RT1 - RT4. Во вторую область входят сети N6 - N8 и маршрутизаторы RT7, RT8, RT10, RT11. Третья область состоит из сетей N9 - N11 и хоста H1 вместе с подключенными к ним маршрутизаторами RT9, RT11 и RT12. Третья область настроена так, что сети N9 - N11 и хост H1 группируются в один маршрут при анонсировании внешних по отношению к данной области путей (см. параграф 3.5).

На рисунке 4 маршрутизаторы RT1, RT2, RT5, RT6, RT8, RT9 и RT12 являются внутренними, RT3, RT4, RT7, RT10 и RT11 – граничными маршрутизаторами областей, а RT5 и RT7 – граничными маршрутизаторами AS.

#### Внутренние маршрутизаторы (Internal)

Маршрутизатор, непосредственно подключенный к сетям единственной области. Такие маршрутизаторы используют единственную копию базового алгоритма маршрутизации.

#### Граничные маршрутизаторы областей (Area border router)

Маршрутизаторы, подключенные к нескольким областям. На граничных маршрутизаторах областей используется несколько копий базового алгоритма маршрутизации – по одной копии на каждую подключенную к маршрутизатору область. Граничные маршрутизаторы областей собирают сведения о топологии подключенных областей для передачи этих сведений в магистральную область. Магистраль распространяет полученную информацию между всеми областями.

#### Магистральные маршрутизаторы (Backbone)

Маршрутизаторы, имеющие интерфейс в магистральную область. К этому типу относятся все маршрутизаторы, подключенные к нескольким областям (т. е., граничные маршрутизаторы областей). Однако магистральные маршрутизаторы не обязаны быть граничными маршрутизаторами областей, поскольку некоторые маршрутизаторы могут быть подключены только к магистральной области.

#### Граничные маршрутизаторы AS (AS boundary router)

Маршрутизаторы, обменивающимися маршрутными данными с маршрутизаторами других AS. Такие маршрутизаторы анонсируют внешние маршрутные

Таблица 5 Сети, анонсируемые в магистраль, маршрутизаторами RT3 и RT4.

Сеть	Анонс RT3	Анонс RT4
N1	4	4
N2	4	4
N3	1	1
N4	2	3

В таблице 6 показана база данных о каналах для области 1. Эта таблица полностью описывает внутрисетевую маршрутизацию. В таблице также приведено полное представление внешних сетей (internet) для двух внутренних маршрутизаторов RT1 и RT2. Ответственность за анонсирование в области 1 дистанций до всех внешних адресатов ложится на граничные маршрутизаторы области - RT3 и RT4. Маршрутизаторы RT3 и RT4 должны также анонсировать в область 1 местоположение граничных маршрутизаторов AS RT5 и RT7. И, наконец, внешние по отношению к AS анонсы LSA от маршрутизаторов RT5 и RT7 рассылаются с использованием лавинной маршрутизации через всю AS и, в частности, через область 1. Эти LSA включаются в базу данных области 1 и дают маршруты в сети N12 - N15.

Таблица 6 База данных области 1

К	От узла						
	RT1	RT2	RT3	RT4	RT5	RT7	N3
у з л у	RT1						0
	RT2						0
	RT3						0
	RT4						0
	RT5			14	8		
	RT7			20	14		
	N1	3					
	N2		3				
	N3	1	1	1	1		
	N4			2			
	Ia, Ib			20	27		
	N6			16	15		
	N7			20	19		
	N8			18	18		
	N9-11, H1			29	36		
	N12				8	2	
	N13				8		
	N14				8		
	N15					9	

Маршрутизаторы RT3 и RT4 также будут резюмировать топологию области 1 для передачи в магистраль. Анонсы LSA от этих маршрутизаторов показаны в таблице 4. Эти анонсы показывают какие сети включены в область 1 (N1 - N4) и указывают дистанцию до этих сетей от маршрутизаторов RT3 и RT4, соответственно.

База данных о состоянии каналов для магистральной области показана в таблице 7. В таблицу включены маршрутизаторы, являющиеся магистральными. Маршрутизатор RT11 является магистральным, поскольку он подключен к двум областям. Для организации магистрали между маршрутизаторами R10 и R11 организовано виртуальное соединение.

Граничные маршрутизаторы областей RT3, RT4, RT7, RT10 и RT11 собирают маршрутные сведения от своих областей, не относящихся к магистрали для распространения этих данных через магистраль AS. Напомним, что для третьей области настроена группировка сетей N9 - N11 и хоста H1 в один маршрут. Этим обусловлена общая запись для сетей N9 - N11 и хоста H1 в таблице 7. Маршрутизаторы RT5 и RT7 являются пограничными для AS и полученная от них

внешняя маршрутная информация также показана в таблице 7.

Магистраль обеспечивает возможность обмена маршрутной информацией между граничными маршрутизаторами областей. Каждый граничный маршрутизатор области получает суммарную информацию для других областей от всех остальных граничных маршрутизаторов областей. После этого маршрутизатор может сформировать таблицу дистанций до всех сетей за пределами своей области, проверяя собранные анонсы LSA и добавляя дистанцию через магистраль до каждого анонсирующего маршрутизатора.

Если снова использовать в качестве примера маршрутизаторы RT3 и RT4, расчет будет включать следующие этапы:

- ◆ Сначала рассчитывается дерево SPF для магистрали. Этот расчет дает дистанции до всех остальных граничных маршрутизаторов областей. Определяются также дистанции до сетей (Ia и Ib) и граничных маршрутизаторов AS (RT5 и RT7), входящих в магистраль. Эти расчеты показаны в таблице 8.
- ◆ После этого просматриваются резюме областей от граничных маршрутизаторов RT3 и RT4 для определения дистанции до всех маршрутизаторов за пределами данной области. Эти данные анонсируются внутри области маршрутизаторами RT3 и RT4. Анонсы, которые маршрутизаторы RT3 и RT4 будут передавать в область 1, показаны в таблице 9. Отметим, что в таблице 9 предполагается группировка интерфейсов Ia и Ib в один анонс LSA.

Таблица 8 Магистральные дистанции, рассчитанные RT3 и RT4			
Д и с т а н ц и я  д о	Дистанция	от RT3	от RT
	RT3	*	21
	RT4	22	*
	RT7	20	14
	RT10	15	22
	RT11	18	25
	Ia	20	27
	Ib	15	22
	RT5	14	8
	RT7	20	14

Информация, импортируемая в область 1 маршрутизаторами RT3 и RT4, позволяет внутренним маршрутизаторам (например, RT1) выбирать граничный маршрутизатор области более эффективно. Маршрутизатор RT1 будет использовать RT4 для трафика в сеть N6, RT3 – для сети N10, а для трафика в сеть N8 будут использоваться оба маршрутизатора с распределением нагрузки.

Маршрутизатор RT1 таким же способом может определить кратчайший путь до граничных маршрутизаторов AS (RT5 и RT7). Просматривая в маршрутизаторах RT5 и RT7 записи AS-external-LSA, маршрутизатор RT1 может выбирать между RT5 и RT7 при пересылке пакетов адресатам из других AS (одна из сетей N12 - N15).

Отметим, что разрыв соединения между маршрутизаторами RT6 и RT10 будет приводить к разрыву магистрали. Организация виртуального соединения между маршрутизаторами

RT7 и RT10 позволит обеспечить устойчивость магистрали к такого рода сбоям.

Таблица 7 База данных магистральной области

К	От узла						
	RT3	RT4	RT 5	RT6	RT7	RT10	RT11
у з л у	RT3			6			
	RT4		8				
	RT5		8	6	6		
	RT6	8	7			5	
	RT7		6				
	RT10			7			2
	RT11	3				3	
	N1	4	4				
	N2	4	4				
	N3	1	1				
	N4	2	3				
	Ia					5	
	Ib			7			
	N6				1	1	3
	N7				5	5	7
	N8				4	3	2
	N9-11, H1						11
	N12		8		2		
	N13		8				
	N14		8				
	N15				9		

(например, RT1) выбирать граничный маршрутизатор области более эффективно. Маршрутизатор RT1 будет использовать RT4 для трафика в сеть N6, RT3 – для сети N10, а для трафика в сеть N8 будут использоваться оба маршрутизатора с распределением нагрузки.

3.5. Поддержка подсетей IP

OSPF связывает адресную маску IP с каждым анонсируемым маршрутом. Маска показывает диапазон адресов, описываемых данным маршрутом. Например, summary-LSA для адресата 128.185.0.0 с маской 0xffff0000 на самом деле описывает маршрут к



адресатам в диапазоне 185.0.0 - 128.185.255.255. Аналогичным способом указываются маршруты к хостам – маска 0xfffffff показывает наличие в подсети единственного адреса.

Включение масок в каждый анонсируемый маршрут позволяет использовать при распределении адресов маски переменной длины (variable-length subnetting). Это значит, что в одной IP-сети класса А, В или С может существовать множество подсетей различных размеров. Например, сеть 128.185.0.0 можно разбить на 62 подсети переменного размера: 15 подсетей по 4К, 15 подсетей по 256, и 32 подсети по 8 адресов. В таблице 10 приведены примеры адресов таких подсетей вместе с масками.

Таблица 10 Примеры масок подсетей

Адрес сети	Маска IP	Размер подсети
128.185.16.0	0xfffff000	4К
128.185.1.0	0xfffff00	256
128.185.0.8	0xfffff8	8

Существует множество способов деления сетей класса А, В и С на подсети переменного размера.

Описание процедур такого деления выходит за пределы данной спецификации, однако мы рекомендуем пользоваться следующим правилом – при пересылке

пакетов IP их следует адресовать в ту сеть, которая точнее всего совпадает с адресом получателя пакета. В контексте подсетей наилучшее соответствие эквивалентно самой длинной маске<sup>5</sup>. Например, используемый по умолчанию маршрут с адресом 0.0.0.0 и маской 0x00000000 соответствует любому из возможных адресов IP, но этот адрес является наименее соответствующим из всех возможных. Маски подсетей следует задавать таким образом, чтобы можно было однозначно определить наилучшее соответствие для любого адреса IP.

Добавление адресных масок в каждый маршрут позволяет поддерживать «сверхсети» (IP supernetting). Например, физическому сегменту может соответствовать пара адрес-маска [192.9.4.0, 0xfffffc00]. Такой сегмент, являющийся единой сетью IP, содержит адреса из четырех последовательных сетей класса С – от 192.9.4.0 до 192.9.7.0. Такая адресация стала общепринятой после введения CIDR (см. [Ref10]).

Для более эффективного объединения маршрутов на границе области можно указать диапазон адресов области (см. Приложение С.2). Каждый диапазон адресов задается парой [адрес, маска]. Это позволяет объединить множество разных сетей в общий диапазон адресов (как сеть делится на множество разных подсетей). Граничные маршрутизаторы областей будут резюмировать содержимое области (для передачи в магистраль) анонсируя один маршрут для каждого диапазона адресов. Стоимость для такого маршрута устанавливается равной максимальному значению стоимости сетей из данного диапазона адресов.

Например, сеть, разделенная на подсети IP, может быть настроена как единая область OSPF. В таких случаях можно обойтись единственным диапазоном адресов – номером сети класса А, В или С с соответствующей классу маской адреса. Внутри области может быть организовано любое число подсетей, однако за пределы области будет распространяться единственный маршрут, скрывающий факт разбиения на подсети. Стоимость такого маршрута будет равна максимальной из стоимостей входящих в область подсетей.

### 3.6. Поддержка «тупиковых» областей

В некоторых автономных системах основная часть базы данных о каналах может состоять из внешней информации AS-external-LSA. Записи OSPF AS-external-LSA обычно рассылаются с использованием лавинной маршрутизации через всю AS. Однако протокол OSPF позволяет указать некоторые области как тупиковые (stub area) и анонсы AS-external-LSA не будут рассылаться в такие области или передаваться через них. Маршрутизация внешним по отношению к AS адресатам в таких системах базируется полностью на принятых по умолчанию для таких областей маршрутах. Выделение «тупиковых» областей позволяет уменьшить размер базы данных link-state и связанный с этим расход памяти для внутренних маршрутизаторов stub-областей.

Чтобы воспользоваться преимуществами поддержки тупиковых областей OSPF, для таких областей следует обязательно задавать принятые по умолчанию маршруты. Для этого один или несколько граничных маршрутизаторов stub-области должны анонсировать принятый по умолчанию маршрут с помощью summary-LSA. Такие анонсы рассылаются с использованием лавинной маршрутизации внутри тупиковой области, не покидая ее пределов (принятый по умолчанию маршрут относится только к данной области). Заданный для использования по умолчанию маршрут будет применяться во всех случаях, когда адресат недоступен явно с использованием пути внутри области или между областями AS (т. е., адресат относится к другой автономной системе).

Область можно назначить тупиковой, если имеется единственный выход из данной области или когда точек выхода несколько, но для выбора не требуется принимать решение с учетом адреса внешнего (по отношению к области) получателя. Например, область 3 на рисунке 4 можно указать как тупиковую, поскольку весь внешний трафик этой области должен проходить через один граничный маршрутизатор RT11. Если область 3 указана как тупиковая, маршрутизатор RT11 будет анонсировать принятый по умолчанию маршрут для его распространения внутри области 3 (с анонсах summary-LSA) взамен лавинной рассылки анонсов AS-external-LSA для сетей N12 - N15 в область 3.

Протокол OSPF обеспечивает оповещение всех маршрутизаторов области о тупиковом характере данной области – такое решение предотвращает ненужные лавинные рассылки анонсов AS-external-LSA.

Существуют некоторые ограничения на использование stub-областей. Через такие области невозможно организовать виртуальные соединения и, кроме того, внутрь тупиковых областей не должны попадать граничные маршрутизаторы AS.

### 3.7. Разделение областей

OSPF не предпринимает активных попыток восстановления распавшихся на части областей (area partition). В таких случаях просто каждый фрагмент области рассматривается как новая область и магистраль автономной системы обеспечивает маршрутизацию между новыми областями. Некоторые адресаты, для которых использовалась внутридоменная маршрутизация, могут после раздела потребовать междоменной маршрутизации.

Однако для сохранения полной маршрутизации после разделения области диапазон адресов не должен разделяться на множество компонент, связанных с фрагментами области. Не должна делиться на части и магистраль. Если это произойдет, некоторые части AS перестанут быть доступными. Разорванную магистральную область можно собрать заново путем организации виртуальных соединений (см. параграф 15).

<sup>5</sup> Самой длинной маске будет соответствовать самая мелкая из возможных подсетей. Прим. перев.



Другой способ представления раздела областей основан на графах AS, описанных в параграфе 2. Идентификаторы областей (Area ID) можно представить цветом ребер графа<sup>6</sup>. Каждое ребро графа подключено к сети или является сетью point-to-point. В любом случае цвета графов отображают идентификаторы областей. Группа ребер одного цвета, соединенных вершинами, представляет область. Если топология AS не изменяется, граф будет иметь несколько цветных областей для различных идентификаторов Area ID.

Когда топология AS изменяется, одна из областей может быть разделена на части. В этом случае граф AS будет включать несколько областей одного цвета (Area ID). Маршрутизация в AS будет сохраняться до тех пор, пока такие области одного цвета могут быть соединены через единую магистральную область.

## 4. Основные функции

В каждой области работает отдельная копия базового алгоритма маршрутизации OSPF. Маршрутизаторы с интерфейсами в разные области поддерживают соответствующее число копий алгоритма маршрутизации. В последующих параграфах приведено краткое описание алгоритма.

При активизации маршрутизатора он сначала инициализирует структуры данных протокола. После инициализации маршрутизатор ждет от протоколов нижележащих уровней индикации активности интерфейсов. Далее маршрутизатор отыскивает соседей с помощью протокола OSPF Hello – для этого соседям передаются пакеты Hello и в ответ должны также прийти пакеты Hello. В ширококестельных сетях и сетях point-to-point маршрутизаторы динамически детектируют соседей, передавая пакеты Hello по групповому адресу AllSPFRouters. В сетях без поддержки ширококестельного обнаружения соседей может потребоваться некоторой настройкой маршрутизатора. В ширококестельных сетях и сетях NBMA протокол Hello также указывает выделенный маршрутизатор (Designated Router или DR).

Маршрутизатор будет пытаться установить отношения смежности (adjacency) с некоторыми из новообнаруженных соседей. Пары смежных маршрутизаторов синхронизируют между собой базы данных о каналах. В ширококестельных сетях и сетях NBMA выделенный маршрутизатор DR определяет, какие маршрутизаторы должны быть смежными. Отношения смежности определяют распространение маршрутной информации – обновления маршрутных данных передаются только смежным маршрутизаторам.

Маршрутизатор периодически анонсирует свое состояние, называемое также состоянием канала - link state. При изменении состояния маршрутизатора изменяется и состояние канала. Отношения смежности маршрутизаторов отражаются в содержимом анонсов LSA. Связь между отношениями смежности и состояниями каналов позволяет протоколу обнаруживать неработающие («мертвые») маршрутизаторы достаточно быстро.

Анонсы LSA рассылаются в области с использованием лавинной маршрутизации. Алгоритм лавинной рассылки надежен и обеспечивает распространение по всей области единой копии базы данных о каналах. База данных содержит набор LSA от каждого маршрутизатора, входящего в область. На основании этих данных каждый маршрутизатор рассчитывает дерево кратчайших путей, используя себя в качестве корня. Это дерево дает таблицу маршрутизации для протокола.

### 4.1. Междоменная маршрутизация

В предыдущем параграфе рассматривалась маршрутизация в пределах одной области. В таких случаях никакой дополнительной маршрутной информации не требуется. Для обеспечения маршрутизации между областями граничные маршрутизаторы областей вводят в область дополнительные сведения о маршрутах. Эта информация представляет собой выжимку из сведений о топологии остальной части AS.

Распространение сведений о топологии автономной системы происходит следующим образом. Каждый граничный маршрутизатор области по определению связан с магистралью области. Все граничные маршрутизаторы областей анонсируют топологию подключенных к ним областей, не являющихся магистральными, для передачи в магистраль и, следовательно, всем остальным граничным маршрутизаторам областей. В результате каждый граничный маршрутизатор области имеет полные сведения о магистрали и суммарные данные по каждой области от граничных маршрутизаторов остальных областей. На основании этих данных маршрутизатор рассчитывает пути для междоменной маршрутизации и анонсирует эти пути в подключенные к нему области. Получив такие сведения, внутренние маршрутизаторы могут выбрать оптимальные точки выхода из области для рассылки междоменного трафика.

### 4.2. Внешние маршруты AS

Маршрутизаторы, имеющие сведения о других AS, могут организовать лавинную рассылку этой информации в своей AS. Внешняя маршрутная информация передается без изменения каждому заинтересованному в ней маршрутизатору. Единственным исключением является то, что внешняя маршрутная информация не передается в «тупиковые» области (см. параграф 3.6).

Для использования внешних маршрутных данных, путь ко всем маршрутизаторам, анонсирующим такие данные, должен быть известен во всей AS (за исключением тупиковых областей). По этой причине местоположение таких граничных маршрутизаторов резюмируется и рассылается граничными маршрутизаторами областей (нетупиковых).

### 4.3. Пакеты протокола маршрутизации

Протокол OSPF работает непосредственно на базе IP, используя идентификатор 89. OSPF не требует какой-либо дополнительной фрагментации/сборки пакетов – при возникновении такой необходимости используется обычная фрагментация и сборка IP. Пакеты протокола OSPF имеют такой формат, что большие блоки протокольной информации в общем случае можно легко разделить на более мелкие пакеты. Рекомендуется использовать именно такой вариант, избегая по возможности фрагментирования IP.

Пакеты протокола маршрутизации всегда должны передаваться с нулевым значением поля IP TOS. Если это возможно, пакеты протоколов маршрутизации должны иметь преимущество по сравнению с обычным трафиком данных IP как для приема, так и при передаче. Чтобы облегчить эту задачу, протокол OSPF должен использовать в поле IP precedence значение Internetwork Control (см. [Ref5]).

<sup>6</sup> Вершины графов представляют маршрутизаторы, транзитные или оконечные сети. Поскольку маршрутизаторы могут принадлежать нескольким областям, невозможно использовать для маркировки цвет вершин графа.

Все пакеты протокола OSPF используют однотипные заголовки, описанные в Приложении А. Типы пакетов OSPF перечислены в таблице 11. Форматы пакетов также описаны в Приложении А.

Протокол OSPF Hello использует пакеты Hello для организации и поддержки соседских отношений. Пакеты Database Description (описание базы данных) и Link State Request (запрос состояния канала) служат для поддержки отношений смежности. Гарантированный обмен обновлениями OSPF основан на обмене пакетами Link State Update (обновление состояния канала) Link State Acknowledgment (подтверждение приема обновления).

Таблица 11 Типы пакетов OSPF

Тип	Название	Назначение
1	Hello	Обнаружение и поддержка соседства
2	Database Description	Резюмирование содержимого БД
3	Link State Request	Загрузка БД
4	Link State Update	Обновление БД
5	Link State Ack	Подтверждение лавинной рассылки

Таблица 12 Типы анонсов OSPF (LSA).

Тип	Имя LSA	Описание LSA
1	Router-LSA	Генерируются всеми маршрутизаторами. Этот тип LSA описывает состояния интерфейсов маршрутизатора в области. Анонс рассылается в лавинном режиме внутри области.
2	Network-LSA	Генерируется выделенным маршрутизатором DR для широковещательных и NBMA-сетей. Этот тип LSA включает список маршрутизаторов, подключенных к сети. Рассылается в лавинном режиме внутри области.
3, 4	Summary-LSA	Генерируется граничными маршрутизаторами областей и рассылается в лавинном режиме в пределах связанной с LSA области. Каждый анонс summary-LSA описывает маршрут к адресату за пределами данной области, но внутри данной AS (междоменный маршрут). Тип 3 summary-LSA описывает маршруты в сети, а тип 4 – к граничным маршрутизаторам AS.
5	AS-external-LSA	Генерируется граничными маршрутизаторами AS и рассылается по всей автономной системе. Каждый анонс AS-external-LSA описывает маршрут к адресатам в другой AS. Принятые по умолчанию маршрутизаторы AS также могут описываться в AS-external-LSA.

Каждый пакет Link State Update содержит набор новых анонсов состояния каналов (LSA) на один интервал (hop) удаленных от пункта генерации анонса. Один пакет Link State Update может содержать анонсы LSA от нескольких маршрутизаторов. Каждая запись LSA помечается идентификатором создавшего анонс маршрутизатора и сопровождается контрольной суммой содержимого. В каждой записи LSA имеется также поле типа; возможные варианты этого поля описаны в таблице 12.

Пакеты протокола OSPF (за исключением пакетов Hello) передаются только между смежными маршрутизаторами. Это означает, что все пакеты протокола OSPF проходят только один интервал между маршрутизаторами (IP hop), за исключением тех ситуаций, когда смежность поддерживается через виртуальные соединения (virtual adjacency). IP-адрес отправителя пакета OSPF является адресом одного из смежных маршрутизаторов, а IP-адрес получателя является адресом второго из смежных маршрутизаторов или групповым IP-адресом.

## 4.4. Основные требования к реализации

Каждая реализация OSPF должна обеспечивать перечисленные ниже возможности.

### Таймеры

Протоколы требуются таймеры двух типов. Первый тип - single shot timers (одноразовый таймер) служит для запуска обработки событий по истечении заданного времени. Второй тип таймеров называют интервальными (interval timer) и они служат для непрерывного отсчета интервалов. Такие таймеры используются для периодической отправки пакетов. Хорошим примером может служить регулярная широковещательная рассылка пакетов Hello. Единица отсчета для таймеров обоих типов составляет 1 сек.

Реализация периодических таймеров не должна иметь дрейфа. В некоторых маршрутизаторах обработка пакетов может влиять на скорость работы таймера. При подключении множества маршрутизаторов к одной сети они все рассылают широковещательные пакеты и может возникнуть синхронная рассылка служебных пакетов, которой следует избегать. Если таймеры не удастся реализовать без дрейфа, следует менять интервалы периодической рассылки на небольшие случайные значения для каждого периода.

### Групповая адресация - IP multicast

Некоторые пакеты OSPF передаются в виде дейтаграмм IP multicast. Поэтому требуется поддержка приема и передачи дейтаграмм с групповыми адресами IP и (при необходимости) соответствующих протоколов нижележащих уровней. Групповые дейтаграммы IP, используемые протоколом OSPF никогда не передаются дальше следующего маршрутизатора. По этой причине поддержка пересылки дейтаграмм IP multicast не требуется. Групповая адресация IP описана в работе [Ref7].

### Подсети разных размеров - Variable-length subnet support

Реализация протокола IP в маршрутизаторах должна поддерживать возможность разделения сетей класса А, В или С на множество подсетей различных размеров. Обычно такое деление называют variable-length subnetting (см. параграф 3.5).

### Поддержка «сверхсетей» - IP supernetting support

Реализация протокола IP в маршрутизаторах должна поддерживать возможность объединения непрерывного набора сетей классов А, В и С в более крупные объекты, называемые сверхсетями (supernet). Поддержка сверхсетей предложена в качестве одного из способов масштабирования маршрутизации IP в рамках сети Internet в целом. Дополнительную информацию о сверхсетях можно найти в работе [Ref10].

### Поддержка протоколов нижележащих уровней - Lower-level protocol support

К упоминаемым в этой спецификации протоколам нижних уровней относятся протоколы доступа в сеть (например, канальный протокол Ethernet). От этих протоколов к OSPF должна передаваться индикация состояний интерфейса (up и down). В случае Ethernet представляется разумным получение информации об отключении сетевого кабеля Ethernet.

### Поддержка протоколов нижних уровней без широковещания - Non-broadcast lower-level protocol support

В сетях без широковещания протокол OSPF Hello можно использовать, обеспечивая индикацию попыток отправки пакетов несуществующему или неработающему маршрутизатору. Например, в сетях X.25 PDN «умерший» соседний маршрутизатор можно указать с помощью X.25 clear (сведения о причинах «умирания» и диагностика), передавая эти данные протоколу OSPF.

### Действия со списками - List manipulation primitives

Большая часть функций OSPF описывается как операции над списками LSA. Например, набор LSA, передаваемых смежному маршрутизатору, пока не будет получено подтверждение, может быть представлен в виде списка. Любая отдельно взятая запись LSA может включаться во множество таких списков. Реализация OSPF должна поддерживать работу с такими списками, удаляя или добавляя в них при необходимости соответствующие LSA.

**Поддержка задач - Tasking support**

Некоторые процедуры, описанные в этой спецификации, включают в себя выполнение других процедур. Иногда такие процедуры должны выполняться в режиме in-line, т. е., до того, как будет завершена текущая процедура. В тексте такие ситуации указаны инструкциями на выполнение процедуры. В других случаях, новая процедура выполняется только после завершения текущей. Такие ситуации показаны инструкциями на планирование задач.

**4.5. Дополнительные возможности OSPF**

Протокол OSPF включает несколько дополнительных возможностей, реализация которых осуществляется по усмотрению разработчиков. Маршрутизатор указывает поддерживаемые им дополнения в своих пакетах OSPF Hello, Database Description LSA. Это позволяет использовать в одной автономной системе маршрутизаторы с разными наборами дополнительных возможностей.

Некоторые возможности должны поддерживаться всеми маршрутизаторами, подключенными к данной области. В таких случаях маршрутизатор не будет воспринимать пакеты Hello от соседей, пока не убедится в наличии требуемого соответствия (т. е., несоответствие возможностей предотвращает организацию отношений соседства). Примером такой дополнительной функции может служить ExternalRoutingCapability (см. ниже).

Другие возможности могут быть согласованы при обмене базами данных (Database Exchange). Это согласование реализуется указанием дополнительных возможностей в пакетах Database Description. Несовпадение возможностей с соседями в данном случае будет приводить лишь к сужению числа используемых дополнений и соответствующей корректировке содержимого баз данных при обмене между соседями.

Наличие дополнительных возможностей может влиять и на процесс построения таблицы маршрутов. Поскольку дополнительные возможности указываются в LSA, маршрутизаторы, неспособные поддерживать те или иные дополнения, могут быть опущены при построении дерева кратчайших путей.

Дополнительные возможности протокола OSPF, определенные в данной спецификации, перечислены ниже. Более подробные сведения о таких возможностях приведены в параграфе A.2.

**ExternalRoutingCapability**

Как было сказано выше (параграф 3.6) целые области OSPF могут быть настроены как тупиковые (stub). Анонсы AS-external-LSA не будут рассылаться в такие области. Эта возможность указывается битом E в поле опций OSPF (см. параграф A.2). Для того чтобы обеспечить согласованную настройку тупиковых областей, все маршрутизаторы, имеющие интерфейсы в такие области, должны устанавливать нулевое значение бита E в своих пакетах Hello (см. параграфы 9.5 и 10.5).

**5. Структуры данных протокола**

Протокол OSPF описывается в данной спецификации в терминах обработки различных структур данных этого протокола. Ниже приведен список структур данных OSPF верхнего уровня. Если структура данных требует инициализации, нужные действия указаны явно. В протоколе OSPF области, интерфейсы и соседи также используют структуры данных, которые описаны ниже вместе с соответствующими объектами.

**Router ID (идентификатор маршрутизатора)**

32-битовое целое число, позволяющее уникально пронумеровать каждый маршрутизатор в AS. Одним из вариантов идентификатора может служить наименьший из IP-адресов маршрутизаторов интерфейса. При смене значения Router ID требуется перезагрузка программ OSPF в маршрутизаторе для работы с новым идентификатором Router ID. В таких случаях маршрутизатор должен удалить свои (self-originated) LSA из области маршрутизации (см. параграф 14.1) до перезагрузки, поскольку в противном случае эти анонсы будут сохраняться в течение времени MaxAge.

**Area structures (структуры области)**

Каждая из областей, к которой подключен маршрутизатор, имеет свою структуру данных, описывающую работу базового алгоритма OSPF. Напомним, что в каждой области работает отдельная копия базового алгоритма OSPF.

**Backbone (area) structure (структура магистральной области)**

Магистральная область OSPF отвечает за распространение данных междоменной (inter-area) маршрутизации.

**Virtual links configured (настроенное виртуальное соединение)**

Виртуальные соединения настраиваются между маршрутизаторами, служащими конечными точками таких соединений. Для организации виртуального соединения маршрутизатор должен быть граничным в области. Виртуальные соединения обозначаются идентификаторами Router ID удаленного маршрутизатора (он также является граничным в своей области). Для организации виртуального соединения оба участвующих в нем маршрутизатора должны быть подключены к общей области, называемой транзитной для виртуального канала (virtual link's Transit area). Виртуальные соединения включаются в магистраль AS, как будто они связаны между собой через сеть «точка-точка» без адресации интерфейсов (unnumbered). Виртуальное соединение использует внутримежную маршрутизацию своей транзитной области для пересылки пакетов. Виртуальные соединения организуются (up) и удаляются (down) при построении дерева кратчайших путей для транзитной области.

**List of external routes (список внешних маршрутов)**

В этот список включаются маршруты ко внешним по отношению к AS адресатам, которые были явно получены от других протоколов маршрутизации (таких, как BGP), заданы администратором или получены иным способом (например, динамическая маршрутная информация, анонсируемая OSPF с заданным значением метрики). Все маршрутизаторы, имеющие такие внешние маршруты, называются граничными маршрутизаторами AS. Внешние маршруты анонсируются маршрутизаторами в область маршрутизации OSPF с помощью AS-external-LSA.

**List of AS-external-LSAs (список AS-external-LSA)**

Часть базы данных о каналах, полученная от граничных маршрутизаторов AS. Этот список содержит маршруты ко внешним по отношению к данной AS адресатам. Отметим, что для граничных маршрутизаторов AS часть записей AS-external-LSA порождена данным маршрутизатором (self-originated).

**The routing table (таблица маршрутизации)**

Эта таблица строится на основе базы данных о каналах (link-state database). Каждая запись в таблице маршрутизации индексируется по получателям и включает набор путей для пересылки пакетов адресатам. Для описания путей используется их тип и адрес следующего маршрутизатора (см. параграф 11).

На рисунке 5 показана структура данных типичного маршрутизатора (RT10 из примера на рисунке 4). Отметим, что маршрутизатор RT10 имеет виртуальное соединение с RT11, а область 2 является для этого соединения транзитной. Это соединение показано звездочками (\*) на рисунке 5. Когда виртуальное соединение активизируется при построении дерева кратчайших путей для области 2, оно становится интерфейсом в магистральную область (см. два магистральных интерфейса, отмеченных на рисунке 3).

## 6. Структура данных области

Структура данных области содержит всю информацию, используемую при работе базового алгоритма маршрутизации OSPF. Каждая область поддерживает свою базу данных link-state. Сеть целиком принадлежит к одной области и каждый интерфейс маршрутизатора тоже подключен к одной области. Отношения смежности организуются между маршрутизаторами одной области. Магистраль OSPF является специальной областью OSPF, отвечающей за распространение данных междоменной маршрутизации.

База данных о каналах состоит из набора router-LSA, network-LSA и summary-LSA, получаемых от маршрутизаторов области. Эта информация рассылается в лавинном режиме в пределах данной области. Список AS-external-LSA (см. параграф 5) также является частью каждой базы каналов.

### Area ID – идентификатор области

32-битовый идентификатор области<sup>7</sup>. Значение ID = 0.0.0.0 зарезервировано для магистральной области.

### List of area address ranges – список адресных диапазонов

Для агрегирования маршрутной информации на границах областей могут использоваться диапазоны адресов. Каждый такой диапазон задается парой [адрес, маска] с указанием информации о состоянии (Advertise или DoNotAdvertise, см. параграф 12.4.3).

### Associated router interfaces – интерфейсы подключенных маршрутизаторов

Интерфейс маршрутизатора, подключенный к области. Интерфейс маршрутизатора может относиться к одной и только к одной области (или магистральной). Для магистральной области этот список включает также все виртуальные соединения. Виртуальное соединение обозначается идентификатором Router ID маршрутизатора на другой стороне соединения; стоимость виртуального соединения совпадает со стоимостью кратчайшего пути через транзитную область между маршрутизаторами.

### List of router-LSAs – список router-LSA

Анонсы router-LSA генерируются каждым маршрутизатором области и описывают состояние интерфейсов маршрутизатора в данную область.

### List of network-LSAs – список network-LSA

Один анонс network-LSA генерируется для каждой транзитной ширококестельной или NBMA-сети в данной области. Анонсы network-LSA описывают набор маршрутизаторов, подключенных в данный момент к области.

### List of summary-LSAs – список summary-LSA

Анонсы Summary-LSA исходят от граничных маршрутизаторов областей и описывают пути к получателям в пределах данной AS, но внешних по отношению к этой области (междоменный маршрут).

### Shortest-path tree – дерево кратчайших путей

Дерево кратчайших путей для данной области с маршрутизатором в качестве корня. Дерево строится на базе анонсов router-LSA и network-LSA с помощью алгоритма Dijkstra (см. параграф 16.1).

### TransitCapability – возможность транзита

Этот параметр показывает возможность передачи через область транзитного трафика, т. е., пакетов, отправители и получатели которых находятся за пределами области. Значение этого параметра определяется при построении для области дерева кратчайших путей (см. параграф 16.1 – поле TransitCapability имеет значение TRUE тогда и только тогда, когда существует один или несколько виртуальных соединений, для которых данная область является транзитной) и используется при построении таблицы маршрутов для области (см. параграф 16.3). Если TransitCapability = TRUE, область называется транзитной (transit area).

### ExternalRoutingCapability

Этот задаваемый административно параметр определяет лавинную рассылку анонсов AS-external-LSA в область. Если рассылка AS-external-LSA не проводится для данной области, область считается тупиковой (stub). В тупиковых областях маршрутизация внешним адресатам осуществляется исключительно на базе принятого по умолчанию summary-LSA для данной области. Магистральная область не может быть тупиковой. Кроме того, через тупиковые области невозможно организовать виртуальные соединения. Дополнительная информация о тупиковых областях содержится в параграфе 3.6.

### StubDefaultCost

Если область указана как тупиковая и данный маршрутизатор является граничным для такой области, параметр StubDefaultCost показывает стоимость принятого по умолчанию summary-LSA, который маршрутизатор должен анонсировать в область (см. параграф 12.4.3).

Если явно не указано иное, в остальной части данной спецификации работа протокола OSPF рассматривается в масштабе одной области.

## 7. Организация отношений смежности

Протокол OSPF организует отношения смежности (adjacency) между соседними маршрутизаторами для обмена маршрутной информацией. Не каждая пара соседних маршрутизаторов является смежной. В этой главе рассмотрены общие вопросы организации отношений смежности, а более детальное рассмотрение приводится в главе 10.

<sup>7</sup> Для записи Area ID часто используют общепринятый формат записи IP-адресов в виде десятичных чисел, разделенных точками.  
Прим. перев.

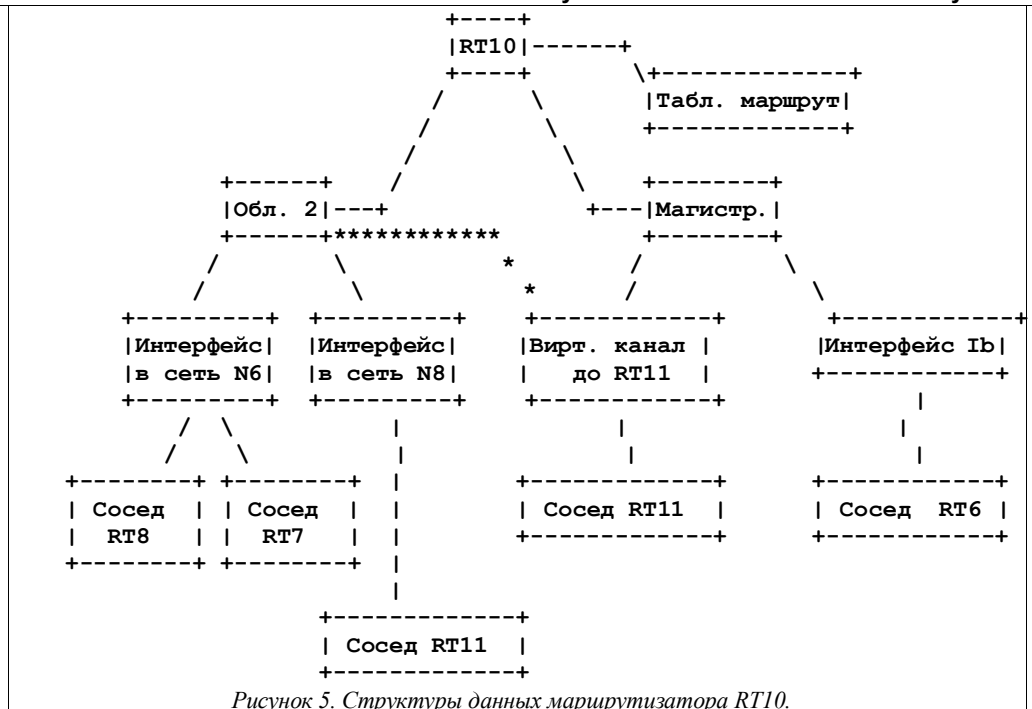


Рисунок 5. Структуры данных маршрутизатора RT10.



## 7.1. Протокол Hello

Протокол Hello отвечает за организацию и поддержку отношений смежности между соседними маршрутизаторами, а также обеспечивает двухстороннюю связь между соседями. Пакеты Hello периодически передаются через все интерфейсы маршрутизатора. Наличие двухсторонней связи устанавливается, когда маршрутизатор видит себя в пакете Hello от соседа. В широковещательных и NBMA-сетях протокол Hello используется также для выбора маршрутизатора DR для данной сети.

Работа протокола Hello различается в широковещательных сетях, сетях NBMA и Point-to-MultiPoint. В широковещательных сетях каждый маршрутизатор анонсирует себя, периодически передавая пакеты Hello с использованием групповой адресации. Это позволяет динамически обнаруживать присутствие соседей. Такие пакеты Hello содержат представление маршрутизатора о тождественности DR и список маршрутизаторов, чьи пакеты Hello были приняты недавно. В сетях NBMA требуется некоторая настройка для обеспечения работы протокола Hello. Каждый маршрутизатор, который потенциально может быть назначен в качестве DR, имеет список всех маршрутизаторов, подключенных к сети. Маршрутизатор, имеющий возможность стать DR, передает пакеты Hello всем остальным потенциальным DR, когда его интерфейс в сеть NBMA начинает работать. Это является попыткой обнаружения маршрутизатора DR для данной сети. Если данный маршрутизатор назначен в качестве DR, он начинает передачу пакетов Hello всем остальным маршрутизаторам, подключенным к сети.

В сетях Point-to-MultiPoint маршрутизатор передает пакеты Hello всем соседям, с которыми он может взаимодействовать напрямую. Поиск таких соседей может осуществляться динамически с использованием протоколов типа Inverse ARP [Ref14] или соседи могут быть заданы статически.

После того, как найдены соседи, организована двухсторонняя связь и (в широковещательных и NBMA-сетях) назначен выделенный маршрутизатор DR, принимается решение о возможности и целесообразности поддержки отношений смежности с соседом (см. параграф 10.4). Если отношения смежности организованы, первым шагом является синхронизация баз данных между смежными маршрутизаторами, подробно описанная в следующем параграфе.

## 7.2. Синхронизация баз данных

Для алгоритмов link-state большое значение имеет синхронизация баз данных о состоянии каналов во всех маршрутизаторах. Протокол OSPF упрощает ситуацию, требуя синхронизировать лишь базы данных смежных маршрутизаторов. Процесс синхронизации начинается сразу после попытки маршрутизаторов организовать отношения смежности. Каждый маршрутизатор описывает свою базу каналов, посылая соседу последовательность пакетов Database Description, каждый из которых описывает набор LSA, включенных в базу данных маршрутизатора. Когда сосед видит анонсы LSA, более новые по сравнению с имеющейся у него копией, он делает отметку о необходимости запроса новейших LSA.

Процесс обмена пакетами Database Description называется обменом базами данных (Database Exchange Process). Во время обмена между маршрутизаторами организуются отношения ведущий – ведомый (master/slave). Каждый пакет Database Description имеет порядковый номер. Переданные ведущим маршрутизатором пакеты Database Description подтверждаются ведомой стороной с возвратом порядкового номера подтверждаемого пакета. Передаваемые в обоих направлениях пакеты содержат резюме данных о состоянии соединений (link state data). Ведущий маршрутизатор может повторно передавать пакеты Database Description с использованием фиксированного интервала, задаваемого для каждого интерфейса административно с помощью параметра RxmInterval.

Каждый пакет Database Description содержит флаг индикации наличия последующих пакетов – бит M. Процесс Database Exchange завершается, когда маршрутизатор принял и передал пакет Database Description с нулевым значением бита M. В процессе Database Exchange и после его завершения каждый маршрутизатор имеет список LSA, для которых сосед имеет более свежие данные. Для запроса этой информации используются пакеты Link State Request. Если запрос не выполнен пакет Link State Request передается повторно по истечении интервала RxmInterval. После завершения процесса Database Exchange<sup>8</sup> и выполнения всех запросов Link State базы данных засинхронизированы и маршрутизаторы маркируются как смежные (fully adjacent). С этого момента отношения смежности являются полными и анонсируются в router-LSA обоих маршрутизаторов.

Смежность используется процедурой лавинной рассылки с самого начала процесса Database Exchange. Это упрощает синхронизацию баз данных и гарантирует ее завершение в разумные сроки.

## 7.3. Выделенный маршрутизатор DR

В каждой широковещательной или NBMA-сети имеется выделенный маршрутизатор DR, выполняющий две основных задачи:

- ♦ Маршрутизатор DR генерирует анонсы network-LSA от имени сети. Эти LSA содержат список маршрутизаторов (включая DR), подключенных в данный момент к сети. Идентификатором Link State ID для таких LSA (см. параграф 12.1.4) является IP-адрес интерфейса в маршрутизаторе DR. Номер IP-сети можно найти по этому адресу, используя адресную маску.
- ♦ Маршрутизатор DR является смежным для всех остальных маршрутизаторов сети. Поскольку базы данных смежных маршрутизаторов синхронизируются (в процессе организации отношений смежности), маршрутизатор DR играет центральную роль в процессе синхронизации данных.

Маршрутизатор DR задается с помощью протокола Hello. Пакет Hello от маршрутизатора содержит значение Router Priority, настраиваемое для каждого интерфейса. В общем случае перед тем как интерфейс маршрутизатора в сеть начнет работать, он проверяет наличие в сети маршрутизатора DR. Если такой маршрутизатор уже задан, он принимается, независимо от значения Router Priority. Такой подход осложняет прогноз тождественности DR, но избавляет от частой смены маршрутизатора DR (см. ниже). Если маршрутизатор DR еще не назначен, им становится данный маршрутизатор, если он имеет наивысшее в сети значение Router Priority. Более детальное и точное описание процедуры выбора DR приведено в параграфе 9.4.

Маршрутизатор DR является конечной точкой для множества смежных пар. Для оптимизации процедуры лавинной рассылки в широковещательных сетях DR использует для передачи своих пакетов Link State Update Packets групповой адрес AllSPFRouters, вместо передачи пакетов каждому из смежных маршрутизаторов по отдельности.

В главе 2 было рассмотрено представление областей в виде графа. Маршрутизаторы помечаются их идентификаторами (Router ID), для обозначения транзитных сетей используются IP-адреса их маршрутизатора DR. Из сказанного следует, что при смене маршрутизатора DR на графе это отражается как полная замена сетевого узла. При такой смене сеть и все подключенные к ней маршрутизаторы будут генерировать новые анонсы LSA. В результате может теряться связность сети, пока не будут засинхронизированы все базы данных. В результате потери связности может быть сгенерировано множество сообщений ICMP unreachable. По этим причинам замены DR-маршрутизатора не должны быть частыми. Параметр Router Priority следует устанавливать таким, чтобы наиболее надежный маршрутизатор сети в конце концов принимал на себя функции DR.

<sup>8</sup> В оригинале ошибочно указано Database Description. *Прим. перев.*



## 7.4. Резервный маршрутизатор DR

Для облегчения процесса перехода к новому маршрутизатору DR вводится понятие резервного DR-маршрутизатора (Backup Designated Router) для каждой широковещательной и NBMA-сети. Маршрутизатор Backup DR также является смежным со всеми маршрутизаторами сети и берет на себя функции выделенного маршрутизатора при сбое прежнего DR. Если в сети нет Backup DR, для назначения нового DR от этого маршрутизатора потребуется организовать отношения смежности со всеми остальными маршрутизаторами в сети. Частью этого процесса является синхронизация баз данных, которая может занимать достаточно много времени, в течение которого сеть будет недоступна для передачи данных. Наличие маршрутизатора Backup DR избавляет от необходимости формирования отношений смежности, поскольку они уже существуют. Это уменьшает время простоя для транзитного трафика до периода, требуемого для лавинной рассылки новых LSA, которые анонсируют новый маршрутизатор DR.

Резервный маршрутизатор DR не генерирует анонсов network-LSA для сети (если это сделать, переход к новому DR дополнительно ускорится, но приходится учитывать соотношение между ростом базы данных и скоростью перехода к новому DR).

Маршрутизатор Backup DR также задается с помощью протокола Hello. Каждый пакет Hello имеет поле для указания Backup DR для сети.

На некоторых этапах процедуры лавинной рассылки Backup DR играет пассивную роль, позволяя маршрутизатору DR выполнять больше работы. В результате снижается уровень локального служебного трафика (см. параграф 13.3).

## 7.5. Граф смежности

Смежность привязывается к сети, в которую оба маршрутизатора имеют интерфейс. Если смежные маршрутизаторы имеют несколько общих сетей, между этими маршрутизаторами организуется соответствующее число отношений смежности.

Набор отношений смежности сети можно представить в форме ненаправленного графа. Вершины графа представляют маршрутизаторы, а ребра соединяют смежные маршрутизаторы между собой. Граф смежности описывает поток пакетов протокола маршрутизации и, в частности, пакетов Link State Update через AS.

В зависимости от наличия в сети выделенного маршрутизатора DR возможны два варианта графа. В физических сетях «точка-точка», сетях Point-to-MultiPoint и виртуальных каналах соседние маршрутизаторы становятся смежными, если они могут обмениваться данными напрямую. В широковещательных и NBMA-сетях маршрутизаторы DR и Backup DR являются смежными для всех остальных маршрутизаторов сети.

Оба варианта графов показаны на рисунке 6. Предполагается, что маршрутизатор RT7 является DR, а RT3 - Backup DR для сети N2. Маршрутизатор Backup DR выполняет меньше работы в процессе лавинной рассылки, нежели маршрутизатор DR (см. параграф 13.3). По этой причине соединение с маршрутизатором Backup DR (RT3) выделено на графе звездочками (\*).

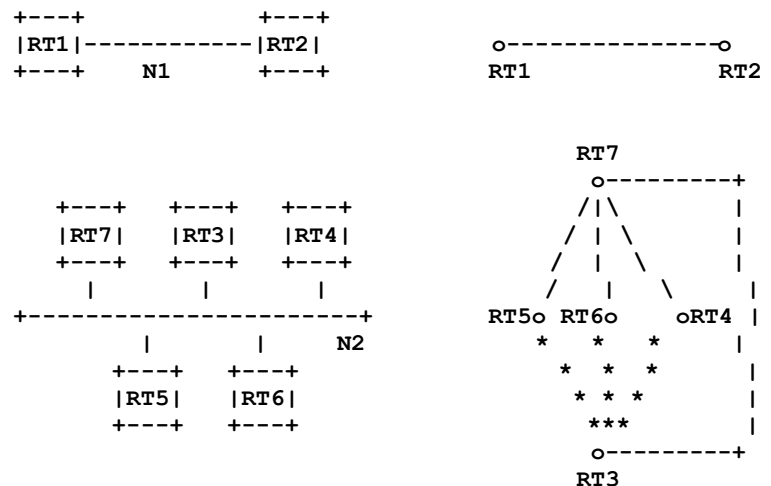


Рисунок 6. Граф смежности

## 8. Обработка пакетов протокола

В этой главе рассмотрены общие вопросы обработки сообщений протокола маршрутизации OSPF. Важно обеспечение синхронного состояния баз данных о каналах в маршрутизаторах. По этой причине пакеты протокола маршрутизации должны иметь преимущества перед обычными пакетами данных как для приема, так и для передачи.

Пакеты протокола маршрутизации передаются только между смежными узлами (исключением являются пакеты Hello, используемые для организации отношений смежности). Это означает, что все пакеты протокола маршрутизации передаются только на один интервал (hop), за исключением пакетов, передаваемых через виртуальные соединения.

Все пакеты протокола маршрутизации используют стандартный заголовок. В следующих параграфах детально описаны правила заполнения и проверки полей заголовка. Далее приведены более детальные описания для всех типов пакетов и сведения об обработке пакетов.

### 8.1. Передача пакетов

Когда маршрутизатор передает пакет протокола маршрутизации, он заполняет поля стандартного заголовка OSPF, перечисленные ниже. Детальное описание формата заголовка приведено в параграфе A.3.1:

#### Version #

Это поле имеет значение 2 – текущий номер версии протокола в соответствии с данной спецификацией.

#### Packet type

Тип пакета OSPF (например, Link State Update или Hello).

#### Packet length

Размер всего пакета OSPF в байтах и учетом стандартного заголовка OSPF.

#### Router ID

Идентификатор маршрутизатора, породившего пакет.

#### Area ID

Идентификатор области OSPF, в которую передается пакет.

#### Checksum

Стандартная контрольная сумма IP для пакета OSPF в целом без учета 64-битового поля аутентификации. Эта контрольная сумма рассчитывается в процессе аутентификации и для некоторых типов аутентификации OSPF контрольная сумма не используется (см. параграф D.4).

**AuType и Authentication**

Весь обмен пакетами OSPF осуществляется с использованием аутентификации. Типы аутентификации задаются протоколом и описаны в Приложении D. Для каждой сети или подсети IP могут использоваться различные процедуры аутентификации. Значение AuType показывает тип используемой процедуры аутентификации. 64-битовое поле аутентификации используется выбранной процедурой проверки полномочий, которая должна вызываться при передаче сформированного пакета (см. параграф D.4).

При установке IP-адреса получателя используются описанные здесь правила. В физических сетях point-to-point в качестве IP-адреса получателя всегда устанавливается значение AllSPFRouters. Для других типов сетей (включая виртуальные соединения), большинство пакетов OSPF передается с указанием точного адреса (unicast) смежного маршрутизатора - Neighbor IP (см. главу 10). В широковещательных сетях для передачи пакетов Hello используется multicast-адрес AllSPFRouters, маршрутизаторы DR и Backup DR передают пакеты Link State Update и Link State Acknowledgment также по групповому адресу AllSPFRouters, а все прочие маршрутизаторы передают пакеты Link State Update и Link State Acknowledgment по групповому адресу AllDRouters.

Retransmissions of Link State Update packets are ALWAYS sent directly to the neighbor. On multi-access networks, this means that retransmissions should be sent to the neighbor's IP address.

В качестве адреса отправителя должен устанавливаться IP-адрес передающего интерфейса маршрутизатора. Интерфейсы в безадресные сети point-to-point не имеют IP-адресов, поэтому для таких интерфейсов в качестве адреса отправителя может устанавливаться IP-адрес любого из интерфейсов данного маршрутизатора. По этой причине каждый маршрутизатор должен иметь по крайней мере один интерфейс с IP-адресом<sup>9</sup>. Отметим, что в большинстве случаев виртуальные соединения работают подобно безадресным сетям point-to-point. Однако с каждым виртуальным каналом связан IP-адрес интерфейса (этот адрес определяется при построении таблицы маршрутизации), который используется в качестве адрес отправителя при передаче пакетов в виртуальный канал.

Более подробные сведения о форматах различных пакетов OSPF приведены в параграфах, указанных в таблице 13.

Таблица 13 Описания способов распространения пакетов OSPF.		
Тип	Название пакета	Параграф
1	Hello	9.5
2	Database description	10.8
3	Link state request	10.9
4	Link state update	13.3
5	Link state ack	13.5

**8.2. Прием пакетов протокола**

Когда пакет протокола принимается маршрутизатором, он маркируется принявшим пакет интерфейсом. Для маршрутизаторов, использующих виртуальные соединения, не всегда возможно сразу определить с каким интерфейсом связан пакет. Рассмотрим в качестве примера маршрутизатор RT11 (рисунок 4). Если RT11 принимает пакет протокола OSPF на своем интерфейсе в сеть N8, этот пакет можно связать с интерфейсом в область 2 или виртуальным каналом к маршрутизатору RT10, являющемуся частью магистрали. При дальнейшем рассмотрении предполагается, что пакет изначально связан не с виртуальным соединением<sup>10</sup>.

Для того, чтобы пакет был воспринят на уровне IP, он должен пройти ряд проверок еще до того, как будет передан протоколу OSPF для обработки:

- ♦ Контрольная сумма IP должна совпадать с одноименным полем заголовка пакета.
- ♦ В качестве адреса получателя должен быть задан IP-адрес принявшего пакет интерфейса или один из групповых адресов IP (AllSPFRouters или AllDRouters).
- ♦ Поле протокола IP должно указывать на OSPF (89).
- ♦ Пакеты локального происхождения не должны передаваться протоколу OSPF – IP-адрес отправителя должен проверяться на предмет совпадения с одним из адресов принявшего пакет маршрутизатора или одним из групповых адресов.

После этого проверяется заголовок OSPF – поля, указанные в заголовке, должны совпадать со значениями, настроенными для принявшего пакет интерфейса. При отсутствии требуемого пакет должен отбрасываться:

- ♦ Поле версии должно содержать значение 2 (текущая версия протокола).
- ♦ Проверяется идентификатор области Area ID из заголовка OSPF – если не выполняется ни одно из двух перечисленных ниже условий, пакет должен быть отброшен.
  - (1) **Идентификатор Area ID относится к области принявшего пакет интерфейса.** В этом случае пакет был принят через один интервал и, следовательно, IP-адрес отправителя должен относиться к той же сети, что и адрес принявшего пакет интерфейса. Это можно проверить, сравнив IP-адрес отправителя из заголовка пакета с IP-адресом принявшего пакет интерфейса после использования для обоих адресов маски подсети, заданной для интерфейса. Такое сравнение не должно проводиться для сетей point-to-point, в таких сетях адреса на каждой стороне соединения выделяются совершенно или могут отсутствовать вообще.
  - (2) **Пакет принят из магистрали.** В таких случаях пакеты передаются через виртуальное соединение. Принявший пакет маршрутизатор должен быть граничным в своей области и значение Router ID, указанное в пакете (маршрутизатор-отправитель) должно относиться к другой стороне виртуального соединения. Принимающий интерфейс должен быть подключен к настроенной для виртуального соединения транзитной области. После выполнения всех перечисленных проверок пакет принимается и ассоциируется с виртуальным соединением (и магистральной областью).
- ♦ Пакеты, отправленные по адресу AllDRouters должны приниматься только маршрутизаторами DR и Backup (см. параграф 9.1).
- ♦ Указанное в пакете значение AuType (тип аутентификации) должно совпадать со значением AuType для связанной с пакетом области.
- ♦ Выполняется процедура аутентификации пакета, указанная полем AuType (см. Приложение D). В процедуре аутентификации может использоваться один или несколько ключей Authentication, которые задаются независимо для каждого интерфейса. Процедура аутентификации может также проверять контрольную сумму полей заголовка OSPF (если эта контрольная сумма используется она представляет собой стандартную контрольную сумму IP для содержимого пакета OSPF без учета 64-битового поля аутентификации). Пакеты, не прошедшие процедуру аутентификации, должны отбрасываться.

Таблица 14 Параграфы, описывающие прием пакетов OSPF.

Тип	Название	Описание
1	Hello	10.5
2	Описание базы данных (Database description)	10.6
3	Запрос состояния канала (Link state request)	10.7
4	Обновление состояния канала (Link state update)	13
5	Подтверждение состояния канала (Link state ack)	13.7

В случае приема пакетов Hello их дальнейшая обработка осуществляется протоколом Hello, описанным в параграфе 10.5. Все остальные типы пакетов передаются только между смежными маршрутизаторами. Это означает, что пакет должен быть передан одним из активных соседей принявшего маршрутизатора. Если принимающий интерфейс подключен к широковещательной сети, сети Point-to-

<sup>9</sup> Все интерфейсы маршрутизатора могут быть подключены к безадресным каналам point-to-point. В таких случаях следует связать IP-адрес с самим маршрутизатором. Этот адрес будет анонсироваться в router-LSA как маршрут к хосту.

<sup>10</sup> Отметим, что в таких случаях оба интерфейса (виртуальный и реальный) имеют одинаковый адрес IP.

MultiPoint или NBMA, отправитель идентифицируется адресом IP в заголовке IP-пакета. Если принимающий интерфейс подключен к сети point-to-или виртуальному соединению, отправитель идентифицируется полем Router ID в заголовке пакета OSPF. Структура данных принимающего интерфейса содержит список активных соседей. Пакеты, не соответствующие любому из перечисленных выше условий, должны отбрасываться.

После проверки пакет ассоциируется с передавшим его активным соседом. Дальнейшие операции по обработке принимаемых пакетов описаны ниже в отдельных параграфах (см. таблицу 14).

## 9. Структура данных интерфейса

Интерфейс OSPF представляет собой соединение между маршрутизатором и сетью. Мы предполагаем один интерфейс OSPF в каждую сеть/подсеть, хотя в приложении F рассматриваются варианты с множеством интерфейсов в одну сеть. С каждой структурой данных интерфейса связан по крайней мере один адрес IP.

Интерфейс OSPF можно рассматривать как часть области, к которой относится соединенная с интерфейсом сеть. Все пакеты протокола маршрутизации, передаваемые маршрутизатором через интерфейс, помечаются идентификатором Area ID соответствующей области. Через один интерфейс может быть организовано несколько отношений смежности. Записи LSA в маршрутизаторе отражают состояние его интерфейсов и отношения смежности.

Ниже приведен список данных, связанных с каждым интерфейсом. Отметим, что многие параметры задают конфигурацию подключенной к интерфейсу сети и должны быть одинаковы для всех подключенных к этой сети маршрутизаторов.

### **Type**

Интерфейсы OSPF могут быть нескольких типов - point-to-point, broadcast, NBMA, Point-to-MultiPoint или virtual link.

### **State**

Функциональное состояние интерфейса, определяющее возможность организации отношений смежности через данный интерфейс. Состояние интерфейса отражается также в LSA маршрутизатора.

### **IP interface address**

IP-адрес, связанный с интерфейсом. Этот адрес появляется в качестве адреса отправителя для пакетов, исходящих от данного интерфейса. Интерфейсы в безадресные сети point-to-point не имеют своего IP-адреса.

### **IP interface mask**

Маска сети/подсети, определяющую часть IP-адреса интерфейса, обозначающую номер сети. Маскирование IP-адресов позволяет определять IP-номер подключенной к интерфейсу сети. В сетях point-to-point и виртуальных каналах маска для интерфейса не задается, поскольку в таких сетях соединение не имеет связанного с ним адреса IP, а адреса для разных сторон соединения задаются независимо или не используются совсем.

### **Area ID**

Идентификатор области, в которую входит подключенная к интерфейсу сеть. Все пакеты протокола маршрутизации, отправляемые с данного интерфейса, помечаются значением ID.

### **HelloInterval**

Промежуток времени (в секундах) между передачей интерфейсом двух последовательных пакетов Hello. Это значение анонсируется в передаваемых интерфейсом пакетах Hello.

### **RouterDeadInterval**

Время (в секундах), по истечении которого соседний маршрутизатор признается неработающим. Время исчисляется от момента приема последнего пакета Hello от соседнего маршрутизатора. Значение этого параметра анонсируется в пакетах Hello.

### **InfTransDelay**

Оценка времени (в секундах), затрачиваемого на передачу пакета Link State Update Packet через данный интерфейс. Анонсы LSA, содержащиеся в пакете Link State Update будут увеличивать свой возраст на величину этого параметра до передачи пакета в интерфейс. Это значение должно учитывать задержки на прохождение и передачу сигнала. Параметр должен иметь положительное значение.

### **Router Priority**

8-битовое беззнаковое целое, определяющее приоритет маршрутизатора. Когда два подключенных к сети маршрутизатора пытаются занять место DR, выбирается маршрутизатор с большим значением Router Priority. Для маршрутизаторов, которые не должны играть роль DR устанавливается Router Priority = 0. Значение этого параметра анонсируется в пакетах Hello.

### **Hello Timer**

Интервальный таймер, используемый для передачи пакетов Hello. Этот таймер запускается каждые HelloInterval секунд. Отметим, что в сетях без широковещания пакеты Hello передаются по отдельности каждому соседу.

### **Wait Timer**

Однократный таймер, вынуждающий интерфейс выходить из состояния Waiting (ожидание) и, как следствие, выбирать маршрутизатор DR в сети. Время отсчета этого таймера составляет RouterDeadInterval.

### **List of neighboring routers**

Список всех прочих маршрутизаторов, подключенных к данной сети, который формируется протоколом Hello. С некоторыми из соседних маршрутизаторов организуются отношения смежности. Набор смежных соседей определяется путем перебора состояний всех соседних маршрутизаторов.

### **Designated Router**

Выделенный маршрутизатор (Designated Router или DR) для подключенной к интерфейсу сети. Маршрутизаторы DR выбираются в широковещательных и NBMA-сетях протоколом Hello. Для маршрутизатора DR сохраняются два параметра - Router ID и IP-адрес интерфейса в данную сеть. Маршрутизатор DR анонсирует состояние канала для сети; анонс network-LSA помечается IP-адресом маршрутизатора DR. При инициализации поле Designated Router принимает значение 0.0.0.0, говорящее об отсутствии в сети маршрутизатора DR.

### **Backup Designated Router**

Маршрутизатор Backup DR также назначается для всех широковещательных и NBMA-сетей с помощью протокола Hello. Все маршрутизаторы сети являются смежными как с DR, так и с Backup DR. Маршрутизатор Backup Designated Router переходит в состояние DR, если текущий маршрутизатор DR «падает». При инициализации Backup DR принимает значение 0.0.0.0, говорящее об отсутствии в сети маршрутизатора Backup DR.

### **Interface output cost(s)**

Стоимость передачи пакета данных через интерфейс, выраженная в единицах метрики link state. Это значение анонсируется как стоимость для данного интерфейса в пакетах router-LSA. Стоимость должна иметь положительное значение.

### **RxmtInterval**

Период (в секундах) повтора передачи LSA для смежных узлов данного интерфейса. Этот же период используется для повтора передачи пакетов Database Description и Link State Request.

### **AuType**

Тип аутентификации, используемой в подключенной к интерфейсу сети/подсети. Описание типов аутентификации приведено в Приложении D. Весь обмен пакетами OSPF осуществляется с использованием аутентификации. В разных сетях/подсетях могут использоваться различные варианты аутентификации.

#### **Authentication key**

Эти задаваемые административно данные позволяют процедуре аутентификации генерировать и/или проверять пакеты протокола OSPF. Ключи аутентификации можно независимо задавать для каждого интерфейса. Например, если поле AuType указывает на использование простого пароля, Authentication key будет содержать 64-байтовое значение пароля, помещаемое в заголовок пакета OSPF без шифрования. Если для AuType указана криптографическая аутентификация (Cryptographic) поле Authentication key будет содержать открытый ключ (shared secret), позволяющий генерировать/проверять цифровые подписи (message digests), добавляемые к пакетам протокола OSPF. При использовании криптографической аутентификации поддерживается множество ключей, что позволяет обеспечить эффективную работу системы аутентификации (см. параграф D.3).

## **9.1. Состояния интерфейса**

В этом параграфе описаны различные состояния, в которых может находиться интерфейс маршрутизатора. Состояния перечислены в порядке расширения их функциональности. Например, нерабочее состояние указано первым, после чего следует набор промежуточных состояний и заканчивается список полнофункциональным состоянием интерфейса. В данной спецификации будет сохраняться такой порядок состояний и могут встречаться фразы типа: «интерфейс находится в состоянии, более высоком по сравнению с X». На рисунке 7 показан граф состояний интерфейса. Дуги графа отмечены событиями, вызывающими смену состояния интерфейса. Эти события описаны в параграфе 9.2. Машина состояний интерфейса описана в параграфе 9.3.

#### **Down**

Изначальное состояние интерфейса, в котором протоколы нижних уровней показывают, что интерфейс неработоспособен. Никакой трафик не может быть передан или принят через неактивный интерфейс. В этом состоянии для параметров интерфейса должны быть установлены начальные значения. Все таймеры должны быть отключены и не должно существовать отношений смежности, связанных с неактивным интерфейсом.

#### **Loopback**

В этом состоянии интерфейс маршрутизатора в сеть закольцован (петля, шлейф, loopback) на программном или аппаратном уровне. Интерфейс не может использоваться для реальной передачи данных. Однако в таком состоянии желательно получать информацию о работе интерфейса с помощью пакетов ICMP (ping) или за счет использования тестового оборудования (скажем, BER-тестера). По этой причине могут появляться пакеты IP, адресованные интерфейсу, который находится в состоянии Loopback. Для упрощения процедур шлейфовой проверки интерфейсов они анонсируются в router-LSA как маршруты к хосту, адрес которого совпадает с IP-адресом проверяемого интерфейса<sup>11</sup>.

#### **Waiting**

В этом состоянии маршрутизатор пытается определить наличие в сети маршрутизатора (Backup) DR. Для решения этой задачи маршрутизатор использует мониторинг принимаемых пакетов Hello. Маршрутизатор не может занимать место Backup DR или DR, пока он не выйдет из состояния Waiting. Это позволяет избавиться от ненужных замен маршрутизатора (Backup) DR.

#### **Point-to-point**

Интерфейс работает и соединен с физической сетью point-to-point или виртуальным каналом. При переходе в это состояние маршрутизатор пытается оформить отношения смежности с соседним маршрутизатором. Соседу передаются пакеты Hello с периодом HelloInterval.

#### **DR Other**

Интерфейс соединен с широковещательной или NBMA-сетью, в которой уже присутствует выделенный маршрутизатор DR. В этом состоянии данный маршрутизатор не может быть назначен в качестве Backup DR. Маршрутизатор формирует отношения смежности с DR и Backup DR (если он назначен).

#### **Backup**

Маршрутизатор используется в качестве Backup DR для подключенной сети. При сбоях основного маршрутизатора DR его место займет данный маршрутизатор. Маршрутизатор формирует отношения смежности со всеми остальными маршрутизаторами подключенной сети. Функции Backup DR несколько отличаются от функций DR в процессе лавинной рассылки Flooding Procedure, (см. параграф 13.3). Детальное описание функций маршрутизатора Backup DR приведено в параграфе 7.4.

#### **DR**

Маршрутизатор выполняет функции DR для подключенной к нему сети и формирует отношения смежности со всеми остальными маршрутизаторами сети. Выделенный маршрутизатор генерирует для сетевого узла анонсы network-LSA, содержащие ссылки на все маршрутизаторы (включая DR), подключенные к сети. Функции выделенного маршрутизатора DR подробно описаны в параграфе 7.3.

## **9.2. События, изменяющие состояние интерфейса**

Состояния интерфейса могут меняться в результате различных событий, показанных на рисунке 7. Приведенные на рисунке метки событий описаны ниже, а детальное рассмотрение этих событий на работу OSPF содержится в параграфе 9.3.

#### **InterfaceUp**

Протоколы нижележащих уровней показали работоспособность сетевого интерфейса. Это позволяет интерфейсу выйти из состояния Down. На виртуальных каналах активизация интерфейса ведет к расчету кратчайшего пути (см. параграф 16.7).

#### **WaitTimer**

Включен таймер Wait, показывающий окончание периода ожидания перед назначением (Backup) DR.

#### **BackupSeen**

Маршрутизатор обнаружил наличие или отсутствие в сети Backup DR. Такое детектирование может выполняться двумя способами – по приему от соседа пакета Hello, содержащего анонс соседнего маршрутизатора как Backup DR или по приему от соседа пакета Hello, содержащего анонс DR и сведения об отсутствии Backup DR. В любом случае с соседом должна быть двухсторонняя связь, т. е., данный маршрутизатор должен также появиться в пакете от соседа. Это событие говорит о завершении состояния Waiting.

#### **NeighborChange**

<sup>11</sup> Что для интерфейсов в безадресные сети point-to-point не генерируется анонсов маршрута к хосту или тестовых IP-пакетов независимо от состояния такого интерфейса.



Это событие связано с изменениями набора соседей данного интерфейса, с которыми осуществляется двухсторонняя связь. В результате требуется перерасчет выделенного маршрутизатора (Backup) DR. Событие NeighborChange может быть вызвано одной из перечисленных ниже причин (описания состояний соседей приведены в параграфе 10.1):

- ◆ Организована двухсторонняя связь с соседом (состояние соседа перешло на уровень 2-Way или выше).
- ◆ Прервана двухсторонняя связь с соседом (состояние соседа перешло на уровень Init или ниже).
- ◆ Один из соседей с двухсторонней связью недавно обозначил себя в качестве DR или Backup DR (определяется из пакетов Hello от этого соседа).
- ◆ Один из соседей с двухсторонней связью перестал играть роль DR или Backup DR (определяется из пакетов Hello от этого соседа).
- ◆ Изменилось анонсируемое значение Router Priority для соседа с двунаправленной связью (определяется из пакетов Hello от этого соседа).

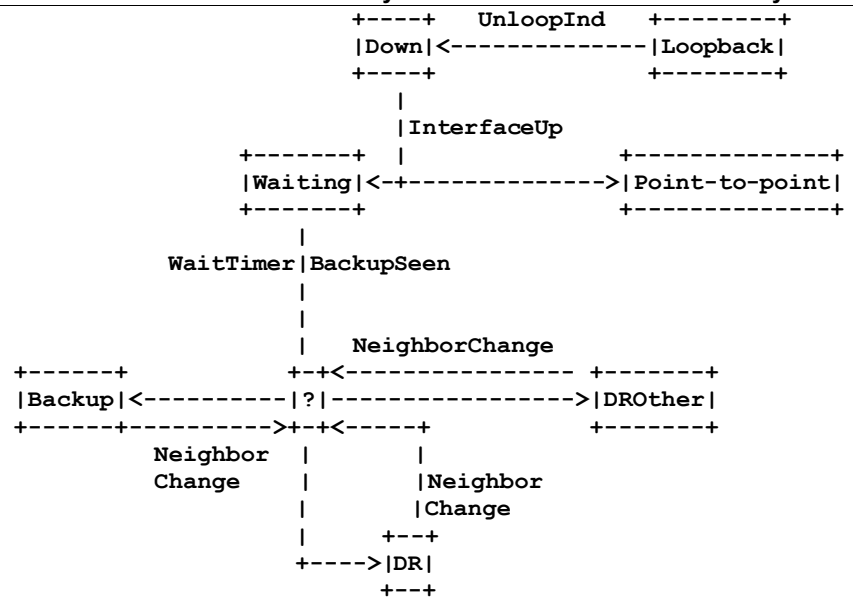


Рисунок 7. Смена состояний интерфейса

В дополнение к показанным на графе переходам событие InterfaceDown всегда приводит к состоянию Down, а LoopInd – к состоянию Loopback.

#### LoopInd

Принята индикация перевода интерфейса в состояние loopback (эта информация может быть получена от программ сетевого управления или протоколов нижележащих уровней).

#### UnloopInd

Принята индикация выхода интерфейса из состояния loopback. Как и для события LoopInd, эта информация может быть получена от программ сетевого управления или протоколов нижележащих уровней.

#### InterfaceDown

Протокол нижележащего уровня сообщил об утрате интерфейсом работоспособности. Независимо от текущего состояния интерфейса он должен быть переведен в состояние Down.

## 9.3. Машина состояний интерфейса

Ниже приводится детальное описание изменений в состояниях интерфейсов. Каждое такое изменение вызывается событиями, перечисленными в параграфе 9.2. Результат событий зависит от текущего состояния интерфейса, поэтому приведенные ниже описания используют в качестве отправной точки текущее состояние интерфейса. Каждая запись в машине состояний интерфейса описывает переход в новое состояние и требуемые действия.

При изменении состояния интерфейса может потребоваться генерация нового анонса router-LSA (см. параграф 12.4).

Некоторые из перечисленных ниже действий включают генерацию событий для машины состояний соседа. Например, при прекращении работы интерфейса должны быть уничтожены все записи о соседстве этого интерфейса. Дополнительная информация о машине состояний соседа приведена в параграфе 10.3.

**Состояние:** Down

**Событие:** InterfaceUp

**Новое состояние:** В зависимости от предпринятых действий

**Действия:** Начало нового отсчета таймера Hello, задающего периодическую рассылку пакетов Hello данным интерфейсом. Если интерфейс подключен к физической сети point-to-point, сети Point-to-MultiPoint или виртуальному соединению, он переводится в состояние Point-to-Point. Для широковещательных сетей и сетей NBMA интерфейс переводится в состояние DR Other, если для маршрутизатора нежелательна работа в качестве DR. Если же маршрутизатор может играть роль DR для подключенной сети, интерфейс переводится в состояние Waiting и запускается таймер Wait Timer. Кроме того, в сетях NBMA проверяется список соседей интерфейса и генерируется событие Start для каждого из соседей, который тоже может стать DR.

**Состояние:** Waiting

**Событие:** BackupSeen

**Новое состояние:** В зависимости от предпринятых действий.

**Действия:** Определяются маршрутизаторы Backup DR и DR (см. параграф 9.4). В результате завершения расчетов интерфейс будет переведен в состояние DR Other, Backup или DR.

**Состояние:** Waiting

**Событие:** WaitTimer

**Новое состояние:** В зависимости от предпринятых действий.

**Действия:** Определяются маршрутизаторы Backup DR и DR (см. параграф 9.4). В результате завершения расчетов интерфейс будет переведен в состояние DR Other, Backup или DR.

**Состояние:** DR Other, Backup или DR

**Событие:** NeighborChange

**Новое состояние:** В зависимости от предпринятых действий.

**Действия:** Заново определяются маршрутизаторы Backup DR и DR (см. параграф 9.4). В результате завершения расчетов интерфейс будет переведен в состояние DR Other, Backup или DR.

**Состояние:** Любое

**Событие:** InterfaceDown

**Новое состояние:** Down

**Действия:** Сбрасываются все связанные с интерфейсом переменные, отключаются таймеры интерфейса и удаляются все отношения соседства (путем генерации события KillNbr для всех соседей – см. параграф 10.2).

**Состояние:** Любое

**Событие:** LoopInd



**Новое состояние:** Loopback

**Действия:** Поскольку интерфейс в таком состоянии больше не связан с подключенной сетью, выполняются операции, связанные с описанным выше событием InterfaceDown.

**Состояние:** Loopback

**Событие:** UnloopInd

**Новое состояние:** Down

**Действия:** Не требуется выполнения каких-либо действий (например, все связанные с интерфейсом переменные уже были сброшены при переходе в состояние Loopback). Отметим, что для перехода интерфейса в рабочее состояние он должен получить сигнал InterfaceUp.

## 9.4. Назначение DR

В этом параграфе описан механизм назначения выделенных маршрутизаторов DR и Backup DR, используемых машиной состояний Interface. Сначала маршрутизатор запускает механизм выбора выделенного маршрутизатора для сети. Значения DR и Backup DR равны 0.0.0.0 – это говорит об отсутствии в сети как DR, так и Backup DR.

Рассмотрим механизм назначения DR более подробно, обозначив маршрутизатор, делающий расчет, как Router X. Проверяется список соседей, подключенных к сети и организовавших двухстороннюю связь с Router X. Этот список в точности совпадает с набором соседей Router X (в той же сети), чьи состояния не ниже 2-Way (см. параграф 10.1). Маршрутизатор Router X также присутствует в этом списке. Сначала из списка исключаются все маршрутизаторы, которые нежелательно использовать в качестве DR (маршрутизаторы с Router Priority = 0 не могут быть DR). После этого для оставшихся в списке маршрутизаторов выполняются следующие операции:

- (1) Фиксируются текущие значения для DR и Backup DR, которые будут потом использоваться для сравнения.
- (2) Определяется новый маршрутизатор Backup DR с учетом только тех маршрутизаторов, которые не заявили о своем нежелании выступать в качестве Backup DR. Если на роль Backup DR претендует (т. е., указали себя в качестве Backup DR, но не DR в своих пакетах Hello) один или несколько маршрутизаторов, среди них выбирается тот, для которого задан высший приоритет Router Priority. Если высший приоритет заявлен для нескольких маршрутизаторов, выбирается тот, у которого больше значение Router ID. Если ни один маршрутизатор не заявлен как Backup DR, выбирается маршрутизатор с наибольшим Router Priority среди тех маршрутизаторов, которые не заявили себя в качестве DR (если таких маршрутизаторов окажется несколько, снова используется Router ID).
- (3) Определяется новый маршрутизатор DR для сети. Если на роль DR претендует (т. е., указали себя в качестве DR в своих пакетах Hello) один или несколько маршрутизаторов, среди них выбирается тот, для которого задан высший приоритет Router Priority. Если высший приоритет заявлен для нескольких маршрутизаторов, выбирается тот, у которого больше значение Router ID. Если ни один маршрутизатор не заявлен как DR, в качестве выделенного маршрутизатора выбирается назначенный в последний раз Backup DR.
- (4) Если маршрутизатор Router X стал (Backup) DR или наоборот, утратил это значение, повторяются п.п. 2 и 3, а затем выполняется пункт 5. Пусть Router X играет роль DR, тогда при повторении п. 2 для X уже невозможно назначение на роль Backup DR. Наряду с другими мерами это предотвращает захват функций DR и Backup DR одним маршрутизатором<sup>12</sup>.
- (5) В результате проведения расчетов маршрутизатор сам может занять место DR или Backup DR (см. параграфы 7.3 и 7.4, содержащие детальное описание). Состояние интерфейса маршрутизатора соответствующим образом изменяется. Если маршрутизатор занял место DR, интерфейс перейдет в состояние DR. Интерфейс маршрутизатора, ставшего Backup DR, перейдет в состояние Backup. В остальных случаях интерфейс переходит в состояние DR Other.
- (6) Если подключенная сеть относится к типу NBMA, а маршрутизатор является DR или Backup DR, он должен начать передачу пакетов Hello тем из соседей, которые не могут стать DR (см. параграф 9.5.1). Это осуществляется созданием события Start для каждого из соседей с Router Priority = 0.
- (7) Если описанные расчеты привели к смене DR или Backup DR, требуется обновить набор отношений смежности, связанных с данным интерфейсом – для этого может потребоваться организация новых отношений смежности или удаление существовавших ранее. Для обновления отношений смежности вводится событие AdjOK? для всех соседей, состояние которых не ниже 2-Way. Это событие приводит к пересмотру отношений смежности (см. параграфы 10.3 и 10.4).

Усложнение механизма назначения выделенного маршрутизатора обусловлено желанием обеспечить упорядоченный переход от Backup DR к DR при сбоях в работе текущего маршрутизатора DR. Упорядоченность перехода обеспечивается за счет введения гистерезиса (запаздывания, *прим. перев.*) – новый маршрутизатор Backup DR не может быть назначен, пока прежний маршрутизатор не примет на себя функции DR.

Описанная выше процедура может привести к выбору одного маршрутизатора в качестве DR и Backup DR, хотя этого никогда не может случиться с маршрутизатором, проводящим вычисления для выбора маршрутизатора (в нашем примере Router X). Выбранный маршрутизатор DR не обязательно имеет наибольшее значение Router Priority, а Backup DR не обязан иметь второе по порядку значение Router Priority. Если Router X не может стать DR, возможно, что приведенные выше процедуры не позволят назначить ни Backup DR, ни DR. Отметим также, что в тех случаях, когда Router X является единственным маршрутизатором, способным стать DR, он может назначить себя на роль DR, а маршрутизатор Backup DR просто не будет выбран для сети.

## 9.5. Передача пакетов Hello

Пакеты Hello передает каждый работающий интерфейс маршрутизатора. Эти пакеты служат для организации и поддержки соседских отношений<sup>13</sup>. В широкополосных и NBMA-сетях пакеты Hello используются также при выборе DR и Backup DR.

Формат пакетов Hello рассмотрен в параграфе A.3.2. Пакеты Hello содержат значения Router Priority (для выбора DR) и HelloInterval (интервала между пакетами Hello передаваемыми интерфейсом). Пакет Hello также показывает, как часто должно быть слышно соседа для осознания его работоспособности (RouterDeadInterval). Значения HelloInterval и RouterDeadInterval должны быть одинаковы для всех маршрутизаторов, подключенных к одной сети. Пакеты Hello также содержат маску IP для подключенной сети (Network Mask); для безадресных сетей point-to-point и виртуальных каналов это поле должно иметь значение 0.0.0.0.

<sup>12</sup> Будет поучительно рассмотреть ситуацию, когда маршрутизатор DR перестает работать. Пусть функции DR выполняет маршрутизатор RT1, а Backup DR – RT2. Если RT1 «падает» (или прекращает работать интерфейс в данную сеть), другие маршрутизаторы смогут обнаружить отсутствие RT1 в течение RouterDeadInterval. Все маршрутизаторы не смогут обнаружить отсутствие выделенного маршрутизатора одновременно. В результате маршрутизатор, обнаруживший отсутствие RT1 до того, как это удалось сделать RT2 на некоторое время выберет RT2 в качестве DR и Backup DR одновременно. Когда RT2 обнаружит отсутствие RT1, он объявит себя DR. Одновременно с этим произойдет назначение маршрутизатора с высшим значением Router Priority на роль Backup DR.

<sup>13</sup> В сетях point-to-point наличие и работоспособность соседа проверяется протоколами нижних уровней с соответствующей индикацией. Для виртуальных каналов существование соседей определяется при расчете таблицы маршрутизации. Однако в том и другом случае по-прежнему используется протокол Hello. Это обеспечивает между соседями двухстороннюю связь и дает гарантию работоспособности протокола маршрутизации на соседнем маршрутизаторе.

Поле Options в пакетах Hello описывает дополнительные возможности OSPF, определенные в данной спецификации (см. параграфы 4.5 и A.2). Бит E в поле Options устанавливается тогда и только тогда, когда подключенная область может обрабатывать анонсы AS-external-LSA (т. е., не является тупиковой). Если бит E установлен некорректно, соседние маршрутизаторы не будут воспринимать пакеты Hello (см. параграф 10.5). Непонятные биты поля Options должны иметь нулевые значения.

Для обеспечения двухсторонней связи между маршрутизаторами, пакеты Hello содержат список всех маршрутизаторов сети, от которых недавно были приняты пакеты Hello. Пакет Hello также содержит текущий выбор маршрутизатора для DR и Backup DR (0.0.0.0 в этих полях говорит о том, что эти маршрутизаторы еще не выбраны).

В широковещательных сетях и физических сетях point-to-point пакеты Hello передаются с интервалом HelloInterval по групповому адресу AllSPFRouters. В виртуальных соединениях пакеты Hello передаются по конкретному адресу на другую сторону виртуального канала каждые HelloInterval секунд. В сетях Point-to-MultiPoint пакеты Hello раздельно передаются каждому подключенному соседу с интервалом HelloInterval. Передача пакетов Hello в сетях NBMA описана ниже.

### 9.5.1. Передача пакетов Hello в сети NBMA

Для работы протокола Hello в сетях без широковещания может потребоваться специальная настройка (см. параграфы C.5 и C.6). В сетях NBMA каждый подключенный маршрутизатор, который может быть выбран DR, знает всех своих соседей по сети (для этого может потребоваться специальная настройка или дополнительный механизм). Для каждого соседа создается метка возможности использования в качестве DR.

Для передачи пакетов Hello в сетях NBMA интерфейс должен иметь состояние не ниже Waiting. Пакеты Hello передаются напрямую (unicast) некоторому подмножеству соседей маршрутизатора. В некоторых случаях пакеты Hello передаются периодически, а в иных ситуациях – в ответ на принятый пакет Hello. Поведение маршрутизатора в части передачи этих пакетов зависит от возможности данного маршрутизатора выполнять функции DR.

Если маршрутизатор может стать DR, он должен периодически посылать пакеты Hello тем соседям, которые также могут быть DR. В дополнение к этому маршрутизаторы, играющие роль DR или Backup DR, должны передавать пакеты Hello всем своим соседям. Это означает, что любая пара маршрутизаторов, способных быть DR, постоянно обменивается пакетами Hello, которые нужны для корректной работы механизма выбора DR. Для минимизации числа передаваемых пакетов Hello число претендентов на роль DR в сетях NBMA должно быть небольшим.

Если маршрутизатор не может стать DR, он должен периодически передавать пакеты Hello маршрутизаторам DR и Backup DR (если он задан). Кроме того, пакеты Hello должны передаваться в ответ на прием пакетов Hello от соседей, способных быть DR, но не являющихся DR или Backup DR. Это требуется для организации начальной двухсторонней связи с любым потенциальным DR.

Когда пакеты Hello периодически отправляются всем соседям, интервал между пакетами Hello определяется состоянием соседа. Если сосед находится в состоянии Down, пакеты Hello передаются каждые PollInterval секунд, в остальных случаях интервал передачи Hello составляет HelloInterval секунд.

## 10. Структура данных Neighbor

Маршрутизаторы OSPF обмениваются данными со своими соседями. В каждом случае обмена используются структуры данных neighbor (сосед). Факт обмена данными ограничен конкретным интерфейсом маршрутизатора OSPF, а для идентификации сеанса используется Router ID соседнего маршрутизатора адрес Neighbor IP (см. ниже). Если маршрутизатор OSPF и второй маршрутизатор подключены к нескольким сетям, возникает множество сеансов обмена данными, каждое из которых описывается уникальной структурой данных о соседе. Каждый отдельный «разговор» между маршрутизаторами будем рассматривать как обмен данными с отдельным соседом.

Структура данных о соседе содержит всю информацию, относящуюся к формируемым или существующим отношениям смежности между двумя соседями (напомним, что не все соседи являются смежными). Смежность можно рассматривать как расширенную форму «разговора» между двумя маршрутизаторами.

### State

Функциональный уровень обмена данными между маршрутизаторами, описанный в параграфе 10.1.

### Inactivity Timer

Однократный таймер, активизируемый при отсутствии пакета Hello от соседа (отсчет RouterDeadInterval секунд).

### Master/Slave

При обмене базами данных между соседями используются отношения «ведущий-ведомый» (master/slave). Ведущий маршрутизатор передает пакет Database Description (это единственная часть обмена, которую при необходимости можно повторять). Ведомый маршрутизатор может только отвечать на пакеты Database Description. Отношения ведущий-ведомый согласуются в состоянии ExStart.

### DD Sequence Number

Порядковый номер пакета Database Description (далее DD, *прим. перев.*), передаваемого в настоящий момент соседу.

### Last received Database Description packet

Биты I (инициализация), M (дополнение) и MS (ведущий), полк Options и порядковый номер DD, содержащиеся в последнем пакете DD, полученном от соседа. Эти данные используются для обнаружения дубликатов пакетов DD.

### Neighbor ID

Идентификатор OSPF Router ID соседнего маршрутизатора. Значение Neighbor ID определяется из пакета Hello, полученного от соседа, или задается административно для случаев виртуальной смежности (см. параграф C.4).

### Neighbor Priority

Значение Router Priority для соседнего маршрутизатора, получаемое из пакета Hello и используемое при выборе DR для подключенной сети.

### Neighbor IP address

IP-адрес интерфейса соседнего маршрутизатора в подключенную сеть. Этот адрес используется как Destination IP при передаче пакетов в режиме unicast. Кроме того, это значение используется как параметр Link ID для подключенной сети в анонсах router-LSA, если соседний маршрутизатор выбран в качестве DR (см. параграф 12.4.1). Адрес Neighbor IP определяется из пакетов Hello, принятых от соседа. Для виртуальных каналов адрес Neighbor IP определяется в процессе построения таблицы маршрутизации (см. главу 15).

### Neighbor Options

Дополнительные возможности OSPF, поддерживаемые соседом. Это значение определяется в процессе Database Exchange (см. параграф 10.6). Дополнительные возможности OSPF сосед также указывает в своих пакетах Hello. Это позволяет отбрасывать принятые пакеты Hello (даже не начав формирование соседских отношений), при рассогласовании некоторых важных опций OSPF (см. параграф 10.5). Дополнительные возможности OSPF описаны в параграфе 4.5.

### Neighbor's Designated Router

Соображения соседа по поводу выделенного маршрутизатора DR. Если сосед предлагает себя как DR, это важно принимать во внимание при локальном расчете DR. Этот параметр определен только для широковещательных и NBMA-сетей.

**Neighbor's Backup Designated Router**

Соображения соседа по поводу Backup DR. Если сосед предлагает себя в этом качестве, это важно принимать во внимание при локальном расчете Backup DR. Этот параметр определен только для широковещательных и NBMA-сетей.

Остальные переменные являются списками LSA. Три списка описывают подмножество базы данных link-state. В этом документе определяется пять различных типов LSA, каждый из которых может присутствовать в базе данных области: router-LSA, network-LSA, summary-LSA типов 3 и 4 (эти 4 типа хранятся в структуре данных области), а также AS-external-LSA (хранится в отдельной глобальной структуре данных).

**Link state retransmission list**

Список LSA, которые были разосланы в лавинном режиме, но не подтверждены для данной смежности. Эти анонсы будут повторяться до тех пор, пока не будет получено подтверждение или разорваны отношения смежности.

**Database summary list**

Полный список LSA, собранных в базу данных области к моменту перехода соседа в состояние Database Exchange. Этот список передается соседу в пакетах DD.

**Link state request list**

Список LSA, которые требуется получить от соседа для синхронизации с ним базы данных. Этот список создается когда принят пакет DD и передается соседу в пакетах Link State Request. Этот список очищается по мере получения соответствующих пакетов Link State Update.

**10.1. Состояния соседей**

Состояние соседа (реально, состояние «разговора» с соседним маршрутизатором) описано в последующих параграфах. Состояния прослушиваются для выполнения требуемых функций. Например, сначала прослушивается нерабочее (inoperative) состояние, после которого проходит несколько промежуточных состояний пока не будет достигнуто финальное. В спецификации используется перечисленный ниже порядок состояний как некая иерархия и в тексте встречаются фразы типа "состояние не ниже X". На рисунках 8 и 9 показаны графы изменения состояний соседа. Дуги графа помечены именами событий, вызывающих смену состояния. Описание этих событий приводится в параграфе 10.2.

Граф на рисунке 8 показывает изменения состояний, связанные с протоколом Hello. Этот протокол отвечает за организацию и поддержку соседских отношений, а также обеспечивает двухстороннюю связь между соседями.

Граф на рисунке 9 показывает формирование отношений смежности (не каждая пара соседей поддерживает отношения смежности – см. параграф 10.4). Организация этих отношений начинается при состоянии соседа ExStart. После того, как пара маршрутизаторов согласует отношения ведущий-ведомый, состояние меняется на Exchange. С этого момента сосед начинает процедуру лавинной рассылки (flooding procedure) и начинается процесс синхронизации баз данных двух маршрутизаторов. По завершении синхронизации сосед переходит в состояние Full и маршрутизаторы становятся полностью смежными. С этого момента смежность указывается в анонсах LSA.

Более детальное описание переходов состояний соседа и связанных с этим дополнительных действий приведено в параграфе 10.3.

**Down**

Начальное состояние разговора с соседом, показывающее, что от соседа в последнее время не приходило информации. В сетях NBMA сообщения Hello могут передаваться соседям в состоянии Down (частота передачи снижается, см. параграф 9.5.1).

**Attempt**

Это состояние корректно только для соседей, подключенных к сетям NBMA, и показывает, что в последнее время от соседа не поступало информации, но попытки ее получения должны продолжаться путем передачи пакетов Hello с интервалом HelloInterval (см. параграф 9.5.1).

**Init**

Это состояние говорит о недавнем приеме пакета Hello от соседа, с которым еще не организована двухсторонняя связь (т. е., данный маршрутизатор еще не включен в пакет Hello от соседа). Все соседи в этом (и более высоких) состоянии указываются в пакетах Hello, передаваемых связанным с ними интерфейсом.

**2-Way**

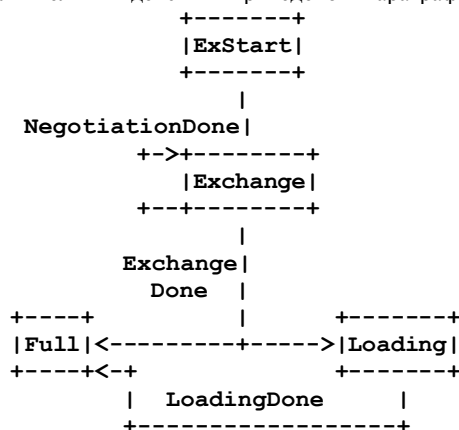
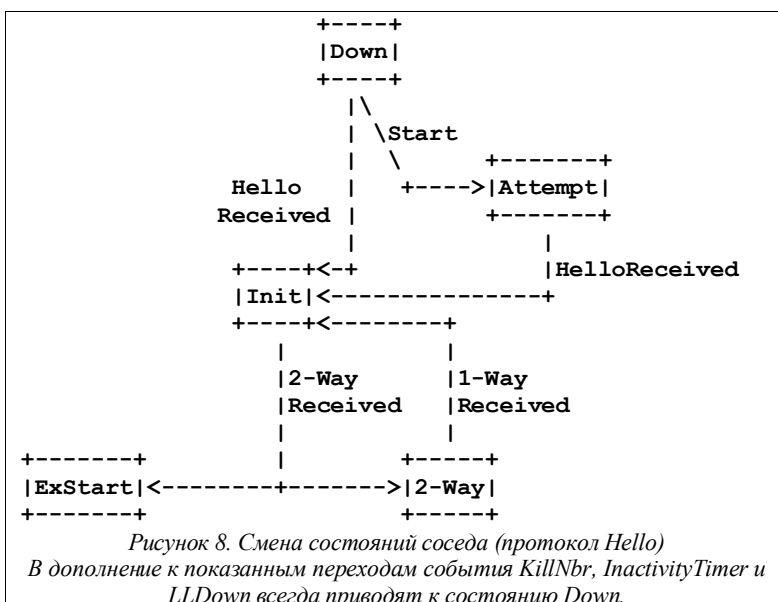
Между соседями организована двухсторонняя связь, обеспечивающая работу протокола Hello. С этого состояния начинается организация отношений смежности. Маршрутизаторы (Backup) DR выбираются из числа соседей в состоянии 2-Way или выше.

**ExStart**

Первый этап организации отношений смежности, обеспечивающий согласование режимов ведущий/ведомый и выбор начального порядкового номера DD. Обмен данными между соседями в данном и более высоких состояниях называют отношениями смежности (adjacency).

**Exchange**

В этом состоянии маршрутизатор полностью описывает свою базу данных link state, посылая пакеты DD своему соседу. Каждый пакет DD имеет порядковый номер и прием пакета должен явно подтверждаться. Только пакеты DD допускается передавать в любой момент. В этом состоянии могут также передаваться пакеты Link State Request для получения от соседа



свежих LSA. Смежные узлы в состоянии Exchange и выше используют лавинную рассылку. Фактически отношения смежности позволяют принимать и передавать все типы пакетов протокола OSPF.

**Loading**

В этом состоянии соседу передаются пакеты Link State Request для получения свежих LSA, которые были обнаружены (но еще не получены) в состоянии Exchange.

**Full**

В этом состоянии обеспечивается полная смежность соседних маршрутизаторов. Отношения смежности отражаются в анонсах router-LSA и network-LSA.

**10.2. События, меняющие состояние соседа**

Изменения состояний могут быть связаны с различными событиями, описанными ниже (названия событий служат метками дуг графа на рисунках 8 и 9):

**HelloReceived**

Получен пакет Hello от соседа.

**Start**

Соседу нужно отправить пакет Hello в течение HelloInterval секунд (только для соседей, связанных с сетями NBMA).

**2-Way Received**

Между двумя соседними маршрутизаторами организована двухсторонняя связь. Такая связь отражается появлением маршрутизатора в пакетах Hello от соседа.

**NegotiationDone**

Согласованы отношения Master/Slave и порядковые номера DD. Это событие служит сигналом для начала приема/передачи пакетов DD. Дополнительные сведения об этом событии приведены в параграфе 10.8.

**ExchangeDone**

Оба маршрутизатора успешно завершили передачу всех пакетов DD. Каждый маршрутизатор сейчас знает, что часть его базы данных устарела. Дополнительные сведения об этом событии приведены в параграфе 10.8.

**BadLSReq**

Получен пакет Link State Request для записи LSA, отсутствующей в базе данных. Это говорит об ошибке в процессе DE.

**Loading Done**

Получены пакеты Link State Update для всех устаревших записей базы данных. Это событие указывается пустым списком в запросе Link state после завершения процесса обмена базами данных (DE).

**AdjOK?**

Требуется принятие решения об организации/поддержке отношений смежности с соседом. Это событие ведет к началу формирования отношений смежности и разрыву существующих отношений.

Перечисленные далее события заставляют снижать уровень соседских отношений. В отличие от перечисленных выше эти события могут происходить при любом состоянии соседских отношений.

**SeqNumberMismatch**

Принят пакет DD, который содержит а) некорректный порядковый номер DD, б) неожиданно установленный бит Init или с) значение поля Options, отличающегося от значения Options в последнем пакете DD. Любое из перечисленных условий говорит о наличии ошибки в процессе организации смежности.

**1-Way**

От соседа получен пакет Hello, в котором не упомянут данный маршрутизатор – отсутствует двухсторонняя связь.

**KillNbr**

Невозможна какая-либо связь с соседом и требуется перейти в состояние Down.

**InactivityTimer**

Активизирован таймер бездействия в результате отсутствия пакетов Hello от соседа. Сосед переводится в состояние Down.

**LLDown**

Информация от нижележащих протоколов о недоступности соседа. Например, сеть X.25 может показывать такие ситуации с явно с указанием причин и данных диагностики. Это событие переводит соседа в состояние Down.

**10.3. Машина состояний Neighbor**

Ниже приводится детальное описание смены состояний с указанием вызвавшего смену события (см. параграф 10.2). События могут иметь различные эффекты в зависимости от текущего состояния соседа. Поэтому машина состояний описывается с учетом текущего состояния и принятого события. Каждая запись содержит сведения о новом состоянии и требуемых действиях.

При смене состояния соседа может потребоваться повтор процедуры выбора маршрутизатора DR. Это определяется генерацией события NeighborChange (см. параграф 9.2). Если интерфейс находится в состоянии DR (данный маршрутизатор является DR), изменения состояния соседа будут приводить к генерации нового анонса network-LSA (см. параграф 12.4).

Когда машине состояний соседа требуется привлечение машины состояний интерфейса, это должно выполняться как запланированная задача (см. параграф 4.4). Это упрощает работу протокола и предотвращает рекурсивные вызовы машины состояний.

**Состояние:** Down

**Событие:** Start

**Новое состояние:** Attempt

**Действия:** Передается пакет Hello соседу (это сосед всегда связан с сетью NBMA) и запускается таймер Inactivity для этого соседа. Большое значение этого таймера говорит о невозможности связи с соседом.

**Состояние:** Attempt

**Событие:** HelloReceived

**Новое состояние:** Init

**Действия:** Заново запускается таймер Inactivity для соседа, поскольку от него была получена информация.

**Состояние:** Down

**Событие:** HelloReceived

**Новое состояние:** Init

**Действия:** Запускается таймер Inactivity для соседа. Большое значение этого таймера говорит о неработоспособности соседа.

**Состояние:** Init or greater

**Событие:** HelloReceived

**Новое состояние:** Состояние не меняется.

**Действия:** Заново запускается таймер Inactivity для соседа, поскольку от него была получена информация.

**Состояние:** Init

**Событие:** 2-WayReceived



**Новое состояние:** Зависит от предпринятых действий.

**Действия:** Определяется необходимость организации отношений смежности с соседом (см. параграф 10.4). Если смежность не требуется, сохраняется состояние 2-Way, в противном случае для соседа вводится состояние ExStart. После смены состояния маршрутизатор увеличивает значение счетчика порядковых номеров DD в базе данных соседа. Если это происходит впервые с момента организации смежности, для порядкового номера DD должно быть выбрано уникальное значение (например, на основе текущего времени). После этого маршрутизатор объявляет себя ведущим (устанавливается значение master для бита master/slave) и начинает передачу пакета DD с установленными битами I, M и MS. Остальные поля этого пакета DD должны быть пустыми. Передача пакета DD должна повторяться с интервалом RxmtInterval до перехода в следующее состояние (см. параграф 10.8).

**Состояние:** ExStart

**Событие:** NegotiationDone

**Новое состояние:** Exchange

**Действия:** Маршрутизатор должен перечислить содержимое своей базы данных LS (link state) для всей области в списке Database summary для соседа. База данных для области включает router-LSA, network-LSA и summary-LSA, содержащиеся в структуре данных области, вместе с AS-external-LSA, содержащимися в глобальной структуре данных. Записи AS-external-LSA опускаются из списка баз данных виртуальных соседей, а также для случая тупиковых областей (см. параграф 3.6). Взамен их в список добавляются LSA, возраст которых равен MaxAge. Резюме списка Database summary будут передаваться соседу в пакетах DD. Каждый пакет DD имеет порядковый номер и прием пакета должен явно подтверждаться. Только для пакетов DD допустима передача в любой момент времени. Дополнительные сведения о работе с пакетами DD содержатся в параграфах 10.8 и 10.6.

**Состояние:** Exchange

**Событие:** ExchangeDone

**Новое состояние:** Зависит от предпринятых действий.

**Действия:** Если список запроса LS от соседа пуст, новым состоянием будет Full и никаких действий не требуется – это финальное состояние отношений смежности. В остальных случаях новым состоянием будет Loading с последующим началом (продолжением) передачи пакетов Link State Request соседу (см. параграф 10.9). Это запросы на получение от соседа свежих записей LSA (обнаруженных, но еще не полученных в состоянии Exchange). LSA перечисляются в списках Link State Request, связанных с соседом.

**Состояние:** Loading

**Событие:** Loading Done

**Новое состояние:** Full

**Действия:** Дополнительных действий не требуется – это финальное состояние отношений смежности.

**Состояние:** 2-Way

**Событие:** AdjOK?

**Новое состояние:** Зависит от предпринятых действий.

**Действия:** Определяет необходимость организации смежности с соседним маршрутизатором (см. параграф 10.4). Если смежность не требуется, сохраняется состояние 2-Way, в противном случае осуществляется переход в состояние ExStart и выполнение действий, описанных выше для состояний Init и 2-WayReceived.

**Состояние:** ExStart или выше

**Событие:** AdjOK?

**Новое состояние:** Зависит от предпринятых действий.

**Действия:** Определяет необходимость сохранения смежности с соседним маршрутизатором. При положительном ответе состояние сохраняется и дополнительных действий не требуется. Если смежность не нужно сохранять, состояние меняется на 2-Way. Списки Link state, Database summary и Link state request удаляются из LSA.

**Состояние:** Exchange или выше

**Событие:** SeqNumberMismatch

**Новое состояние:** ExStart

**Действия:** Отношения смежности (возможно частично организованные) были разорваны и предпринимается попытка восстановить их. Состояние соседа переводится сначала в ExStart. Списки Link state retransmission, Database summary и Link state request удаляются из LSA. После этого маршрутизатор увеличивает значение счетчика порядковых номеров DD в структуре данных о соседе, объявляет себя ведущим (установив для бита master/slave значение master) и начинает передавать пакеты с установленными битами I, M и MS. Остальные поля пакета DD должны быть пустыми (см. параграф 10.8).

**Состояние:** Exchange или выше

**Событие:** BadLSReq

**Новое состояние:** ExStart

**Действия:** Действия для событий BadLSReq в точности совпадают с действиями для SeqNumberMismatch. Отношения смежности (возможно частично организованные) были разорваны и предпринимается попытка восстановить их. Дополнительная информация приведена выше в описании действий по событию SeqNumberMismatch для Exchange и более высоких состояний.

**Состояние:** Любое

**Событие:** KillNbr

**Новое состояние:** Down

**Действия:** Списки Link state retransmission, Database summary и Link state request удаляются из LSA. Отключается таймер Inactivity.

**Состояние:** Любое

**Событие:** LLDwn

**Новое состояние:** Down

**Действия:** Списки Link state retransmission, Database summary и Link state request удаляются из LSA. Отключается таймер Inactivity.

**Состояние:** Любое

**Событие:** InactivityTimer

**Новое состояние:** Down

**Действия:** Списки Link state retransmission, Database summary и Link state request удаляются из LSA.

**Состояние:** 2-Way или выше

**Событие:** 1-WayReceived

**Новое состояние:** Init

**Действия:** Списки Link state retransmission, Database summary и Link state request удаляются из LSA.

**Состояние:** 2-Way or greater

**Событие:** 2-WayReceived

**Новое состояние:** Состояние не меняется.



**Действия:** Дополнительные действий не требуется.

**Состояние:** Init

**Событие:** 1-WayReceived

**Новое состояние:** Состояние не меняется.

**Действия:** Дополнительные действий не требуется.

## 10.4. Когда организуется смежность

Отношения смежности организуются с частью соседей маршрутизатора. Маршрутизаторы, связанные сетями point-to-point, Point-to-MultiPoint или виртуальными каналами, всегда являются смежными. В широковещательных и NBMA-сетях все маршрутизаторы являются смежными с DR и Backup DR.

Решение вопроса об организации отношений смежности происходит в двух точках машины состояний соседа. Первой точкой является начальная организация двухсторонней связи между соседями, а второй – обнаружение смены маршрутизаторов (Backup) DR. Если принято решение не пытаться организовать отношения смежности, смена состояний соседей останавливается на уровне 2-Way.

Отношения смежности следует организовывать при наличии с соседом двухсторонней связи и выполнении, по крайней мере, одного из перечисленных ниже условий:

- ◆ маршрутизаторы связаны сетью point-to-point
- ◆ маршрутизаторы связаны сетью Point-to-MultiPoint
- ◆ маршрутизаторы связаны виртуальным каналом
- ◆ маршрутизатор играет роль DR
- ◆ маршрутизатор играет роль Backup DR
- ◆ соседний маршрутизатор является DR
- ◆ соседний маршрутизатор является Backup DR

## 10.5. Прием пакетов Hello

В этом параграфе рассматривается процесс обработки пакетов Hello (описание формата пакетов приведено в параграфе A.3.2). На приемной стороне базовые операции обработки пакетов OSPF включают проверку заголовка IP и заголовка OSPF. После этого значения полей Network Mask, HelloInterval и RouterDeadInterval в принятом пакете Hello сравниваются с параметрами принимающего интерфейса. При любом несовпадении обработка завершается с отбрасыванием пакета. Иными словами, перечисленные поля реально описывают конфигурацию подключенной сети. Однако это правило имеет одно исключение – в сетях point-to-point и виртуальных каналах значение Network Mask принятого пакета Hello следует игнорировать.

Принимающий интерфейс подключен одной области OSPF (это может быть магистраль). Установка бита E в поле Options пакета Hello должна соответствовать параметру ExternalRoutingCapability для этой области. Если анонсы AS-external-LSA не рассылаются в эту область (область является тупиковой), бит E в пакетах Hello должен быть сброшен, в остальных случаях бит E=1. Несоответствие значений прерывает обработку пакета и пакет отбрасывается. Установка остальных опций Hello должна игнорироваться.

После этого проверяется совпадение адреса отправителя пакета Hello с адресами соседей принимающего интерфейса. Если принимающий интерфейс подключен к широковещательной, Point-to-MultiPoint или NBMA-сети, отправитель идентифицируется по IP-адресу отправителя в заголовке IP пакета Hello. Если принимающий интерфейс подключен к сети point-to-point или виртуальному каналу, отправитель идентифицируется значением Router ID в заголовке OSPF пакета Hello. Текущий список соседних интерфейсов содержится в структуре данных интерфейса. Если отправитель не найден в списке соседей (т. е., получен первый пакет от него), в структуру данных вносятся дополнения. Для нового соседа состояние устанавливается в Down.

При получении пакета Hello от соседа в широковещательной, Point-to-MultiPoint или NBMA-сети, устанавливается значение поля Neighbor ID, равное значению Router ID в заголовке OSPF принятого пакета. Для этих типов сетей поля структуры данных о соседе Router Priority, Designated Router и Backup Designated Router также устанавливаются равными значениям соответствующих полей пакета Hello; изменения этих полей должны быть зафиксированы для последующей обработки. При получении пакета Hello из сети point-to-point (но не из виртуального канала) значение Neighbor IP устанавливается в соответствии с IP-адресом отправителя.

Далее проверяется оставшая часть пакета Hello и генерируются события для машин состояний соседа и интерфейса. Эти машины состояний выполняются немедленно или помещаются в очередь (см. параграф 4.4). В приведенном ниже примере указано, что машина состояний соседа исполняется сразу же (in line) и несколько смен состояния могут быть инициированы одним пакетом Hello:

- ◆ Каждый принятый пакет Hello активизирует машину состояний соседа событием HelloReceived.
- ◆ После этого проверяется список соседей в пакете Hello. Если принявший маршрутизатор присутствует в списке, машине состояний передается событие 2-WayReceived. В противном случае используется событие 1-WayReceived и обработка пакета заканчивается.
- ◆ Далее, если было зафиксировано изменение поля Router Priority, планируется активизация машины состояний интерфейса событием NeighborChange.
- ◆ Если сосед декларирует себя как DR (в пакете Hello поле Designated Router = Neighbor IP), поле Backup DR в пакете имеет значение 0.0.0.0 и принимающий интерфейс находится в состоянии Waiting, планируется активизация машины состояний интерфейса событием BackupSeen. В остальных случаях, если сосед декларирует себя как DR, но не был им до этого или не декларирует себя в этом качестве, хотя ранее играл роль DR, планируется активизация машины состояний принимающего интерфейса событием NeighborChange.
- ◆ Если сосед декларирует себя как Backup DR (в пакете Hello поле Backup Designated Router = Neighbor IP) и приемный интерфейс находится в состоянии Waiting, планируется активизация машины состояний интерфейса событием BackupSeen. В остальных случаях, когда сосед объявил себя Backup DR, не будучи таковым до этого, или снял с себя полномочия Backup DR, планируется активизация машины состояний принимающего интерфейса событием NeighborChange. В сетях NBMA прием пакета Hello может также вызывать передачу ответного пакета Hello (см. параграф 9.5.1).

## 10.6. Прием пакетов DD

В этом параграфе описан процесс обработки принимаемых пакетов DD. Входящие пакеты DD уже связаны с соседом и принявшим интерфейсом в процессе базовой обработки (см. параграф 8.2). Отказ или восприятие и дальнейшая обработка воспринятых пакетов DD зависит от состояния соседа.

Для воспринимаемых пакетов DD в соответствующей структуре данных о соседе после last received Database Description packet (последний принятый пакет DD) должны быть сохранены значения битов I, M и MS, а также поля Options и порядкового номера

DD. Если эти поля совпадают для двух последовательных пакетов DD от одного соседа, второй пакет DD рассматривается в процессе последующей обработки как дубликат.

Если поле Interface MTU в пакете DD показывает, что размер дейтаграммы IP превышает возможности маршрутизатора (требует фрагментации), пакет DD отбрасывается. Дальнейшая обработка зависит от состояния соседа:

#### **Down**

Пакет должен отбрасываться.

#### **Attempt**

Пакет должен быть отброшен.

#### **Init**

Машина состояний соседа должна активизироваться событием 2-WayReceived, что ведет к немедленному переходу в состояние 2-Way или ExStart. Если новым состоянием является ExStart, обработка текущего пакета должна продолжаться для случая ExStart, описанного ниже.

#### **2-Way**

Пакет следует игнорировать. Пакеты DD используются только для организации мостов и отношений смежности<sup>14</sup>.

#### **ExStart**

Если принятый пакет соответствует одному из приведенных ниже условий, машина состояний соседа должна активизироваться событием NegotiationDone (переход в транзитное состояние Exchange), поле Options должно быть сохранено в структуре данных о соседе (Neighbor Options), а пакет должен быть воспринят для последующей обработки (см. ниже).

- Биты I, M и MS установлены, остальные поля пакеты пусты и значение Router ID у соседа больше, чем у принявшего пакет маршрутизатора. Маршрутизатор в данном случае является ведомым, бит master/slave устанавливается в slave и в структуре данных о соседе устанавливается порядковый номер DD, заданный ведущим маршрутизатором.
- Биты I и MS не установлены, порядковый номер DD для пакета равен порядковому номеру DD в структуре данных о соседе (подтверждение) и значение Router ID у соседа меньше, чем у принявшего пакет маршрутизатора (данный маршрутизатор является ведущим).

В остальных случаях пакет следует игнорировать.

#### **Exchange**

Дубликаты пакетов DD отбрасываются ведущим маршрутизатором и заставляют ведомый заново передавать последний пакет DD. Для пакетов, не являющихся дубликатом:

- Если состояние бита MS не соответствует состоянию master/slave для соединения, генерируется событие SeqNumberMismatch для соседа и обработка пакета прекращается.
- Если бит I установлен, для соседа генерируется событие SeqNumberMismatch и обработка пакета прекращается.
- Если поле Options показывает другой набор дополнительных возможностей OSPF по сравнению с полученным от соседа ранее (записан в поле Neighbor Options структуры данных о соседе), для соседа генерируется событие SeqNumberMismatch и обработка пакета прекращается.
- Пакеты DD должны обрабатываться в соответствии с их порядковыми номерами. Если маршрутизатор является ведущим, следующий порядковый номер принятого пакета должен совпадать с номером в структуре данных о соседе. Для ведомого маршрутизатора принятый пакет должен иметь порядковый номер на 1 больше номера, записанного в структуру данных о соседе. В любом случае, следующий по порядку пакет должен быть принят и обработан, как описано ниже.
- В противном случае для соседа генерируется событие SeqNumberMismatch и обработка пакета прекращается.

#### **Loading или Full**

В этих состояниях маршрутизатор передает или принимает всю последовательность пакетов DD. Дубликаты возможны только для принимаемых пакетов (см. выше). Поле Options в пакетах должно соответствовать дополнительным возможностям OSPF, ранее показанным соседом (этот параметр хранится в поле опций структуры данных Neighbor). Прием любых других пакетов, включая пакеты с установленным битом I, должен сопровождаться генерацией события SeqNumberMismatch<sup>15</sup>. Ведущий маршрутизатор должен отбрасывать дубликаты, а ведомый – повторять в ответ передачу последнего пакета DD.

Когда маршрутизатор принимает пакет DD со следующим порядковым номером, этот пакет воспринимается и обрабатывается. Для каждой перечисленной записи LSA проверяется корректность типа LS. Если тип LS неизвестен (т. е., не относится к типам 1 – 5, определенным в данной спецификации) или запись является AS-external-LSA (тип 5), а сосед связан с тупиковой областью, для соседа генерируется событие SeqNumberMismatch и обработка пакета прекращается. В остальных случаях маршрутизатор ищет LSA в своей базе данных. Если такой записи нет или имеется более старая версия (см. параграф 13.1), запись LSA помещается в список Link state request для ее последующего запроса (немедленно или спустя некоторое время) с помощью Link State Request.

Для пакетов DD со следующим порядковым номером, кроме перечисленных выше операций, маршрутизатор выполняет еще ряд действий в зависимости от своей роли – ведущий или ведомый:

#### **Master**

Увеличивается порядковый номер DD в базе данных о соседе. Если маршрутизатор уже передал все пакеты DD и принятый пакет не содержит флага M, для соседа генерируется событие ExchangeDone. В остальных случаях ведомому маршрутизатору передается новый пакет DD.

#### **Slave**

Порядковый номер DD в структуре данных о соседе устанавливается в соответствии с порядковым номером DD в принятом запросе. Ведомый маршрутизатор должен передать в ответ пакет DD. Если в принятом пакете не установлен бит M и переданный ведомым маршрутизатором пакет также не содержит флага M, для соседа генерируется событие ExchangeDone. Отметим, ведомый маршрутизатор всегда генерирует это событие раньше ведущего.

## 10.7. Прием пакетов Link State Request

В этом параграфе описывается процедура обработки принятых запросов Link State Request (LSR). Принятые пакеты LSR содержат списки LSA, которые сосед хочет получить. Пакеты LSR должны восприниматься, если сосед находится в состоянии Exchange, Loading или Full. Для всех остальных состояний соседа пакеты LSR должны игнорироваться.

Каждая запись LSA, указанная в пакете LSR, должна отыскиваться в базе данных маршрутизатора и копироваться в пакет Link State Update (LSU) для передачи соседу. Эти записи **не следует** помещать в список Link state retransmission для соседа. Если LSA

<sup>14</sup> При смене DR достаточно типичным случаем для соседа в этом состоянии будет передача пакета DD для маршрутизатора; это означает наличие кратковременного рассогласования тождественности DR.

<sup>15</sup> Для маршрутизатора возможна рассинхронизация полных отношений смежности путем перехода назад в состояние ExStart. Это заставит смежный маршрутизатор обрабатывать сообщение SeqNumberMismatch и вернуться в состояние ExStart.

не удастся найти в базе данных, это говорит о наличии ошибки в процессе Database Exchange и ведет к генерации события BadLSReq.

## 10.8. Передача пакетов DD

В этом параграфе описан процесс передачи соседу пакетов Database Description (DD). Для пакетов DD значение поля Interface MTU устанавливается равным размеру максимальной дейтаграммы IP, которая может быть передана интерфейсом без фрагментации. Общепринятые в Internet значения MTU можно найти в таблице 7-1 работы [Ref22]. Для пакетов DD, передаваемых через виртуальные соединения устанавливается Interface MTU=0.

Дополнительные возможности OSPF (см. параграф 4.5) сообщаются соседу через поле Options пакета DD. Маршрутизатор должен поддерживать такой же набор дополнительных возможностей в процедурах Database Exchange и лавинной рассылки. Если по каким-то причинам дополнительные возможности маршрутизатора изменяются, следует заново выполнить процедуру Database Exchange путем возврата для соседа состояния ExStart. В данной спецификации определяется только одна дополнительная возможность (см. параграфы 4.5 и A.2). Флаг E- устанавливается тогда и только тогда, когда подключенная сеть не относится к тупиковой области. Нераспознанные биты поля Options должны иметь нулевое значение.

Передача пакетов DD зависит от состояния соседа. В состоянии ExStart маршрутизатор передает только пустые пакеты DD с установленными битами I, M и MS, повторяя с интервалом RxmtInterval. В состоянии Exchange пакеты DD реально содержат резюме информации о состоянии каналов, содержащейся в базе данных маршрутизатора. Каждая запись LSA в базе данных области (во время перехода соседа в состояние Exchange) указывается в списке Database summary соседа. Каждый новый пакет DD копирует свой порядковый номер из структуры данных о соседе и после этого описывает начальную часть списка Database summary. Описанные элементы удаляются из списка после подтверждения приема содержащего их пакета.

В состоянии Exchange момент передачи пакетов DD определяется ролью маршрутизатора (ведущий или ведомый):

### **Master**

Пакеты DD передаются, если а) ведомый маршрутизатор подтвердил прием предыдущего пакета DD, указав его номер в отклике, или б) прошло RxmtInterval секунд без подтверждения приема предыдущего пакета (заново передается прежний пакет).

### **Slave**

Пакеты DD передаются только в ответ на получение пакета DD от ведущего маршрутизатора. Если получен новый пакет DD, в ответ передается новый отклик DD, в противном случае снова передается предыдущий пакет DD.

В состояниях Loading и Full ведомый маршрутизатор должен повторять передачу последнего пакета DD в ответ на прием дубликата пакета DD от ведущего маршрутизатора. По этой причине ведомый маршрутизатор должен ждать RouterDeadInterval секунд перед освобождением пакета DD. Получение пакета DD от ведущего маршрутизатора по истечении этого времени ведет к генерации для соседа события SeqNumberMismatch.

## 10.9. Передача пакетов Link State Request

При состояниях соседа Exchange и Loading список запросов состояний каналов содержит те LSA, которые нужно получить от соседа. Для запроса LSA маршрутизатор сначала отправляет соседу список запросов, помещенный в пакет LSR.

Когда сосед отвечает на эти запросы пакетами Link State Update, соответствующие элементы удаляются из списка запросов и передается новый пакет LSR. Этот процесс продолжается, пока список запросов не будет исчерпан. Записи из списка LSA, которые уже были запрошены, но еще не получены, помещаются в пакет LSR для повторной передачи по истечении интервала RxmtInterval. В любой момент времени должен сохраняться по крайней мере один необработанный пакет LSR.

Когда список запросов исчерпан и сосед находится в состоянии Loading (передан и принят полный набор пакетов DD от соседа) для соседа генерируется событие Loading Done.

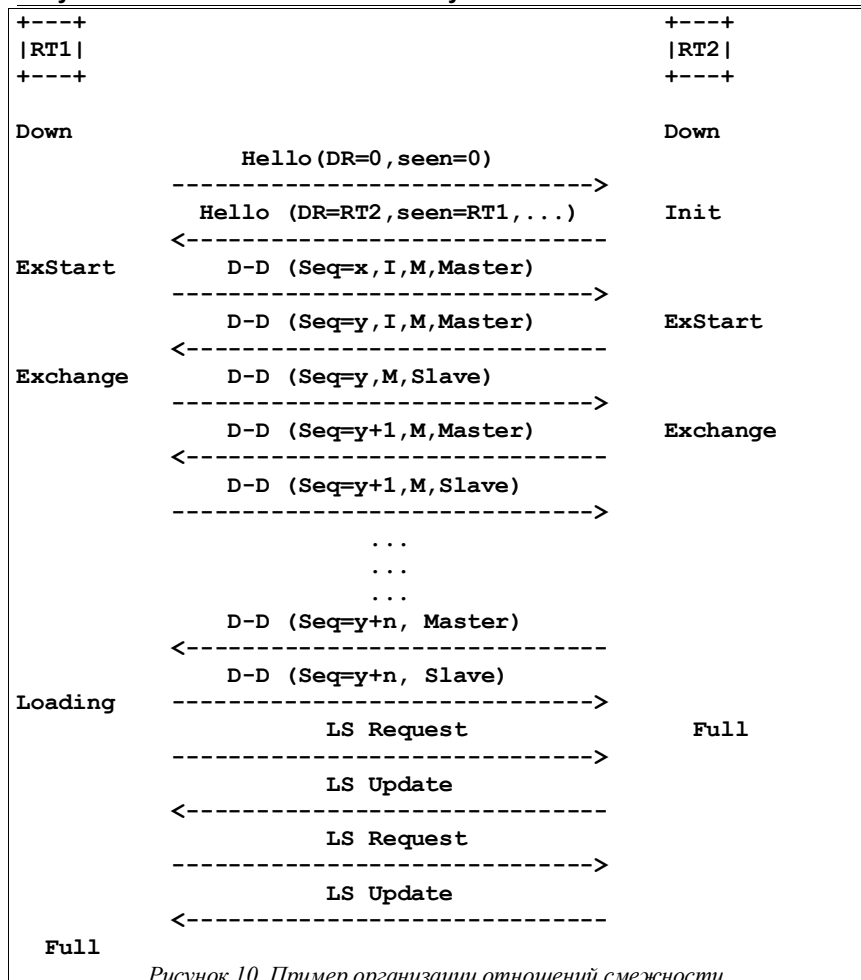


Рисунок 10. Пример организации отношений смежности.

## 10.10. Пример

На рисунке 10 показан пример организации смежности. Маршрутизаторы RT1 и RT2 подключены к широковещательной сети. Предполагается, что RT2 играет роль DR и его значение Router ID больше, нежели Router ID для RT1.

Изменения состояния соседа отслеживаются каждым и показаны по краям рисунка.

На начальном этапе примера маршрутизатор RT1 только начинает работу в сети. Маршрутизатор начинает передавать пакеты Hello, хотя он еще не знает о DR и других соседних маршрутизаторах. RT2 слышит эти пакеты (и вводит состояние Init для соседа) и в своем следующем пакете Hello указывает себя как DR и показывает прием пакетов Hello от RT1. Это заставляет RT1 перейти в состояние ExStart, начиная формирование смежности.

RT1 начинает заявлять себя как ведущий маршрутизатор. Когда RT1 видит, что RT2 должен быть ведущим (большее значение Router ID), он переходит в режим ведомого и адаптирует свой порядковый номер DD для соседа. Происходит обмен пакетами DD с запросами от ведущего (RT2) и откликами от ведомого (RT1) маршрутизатора. Эта последовательность пакетов DD завершается когда обе стороны сбрасывают флаг M.

В приведенном примере предполагается, что база данных RT2 полностью актуальна. В этом случае RT2 незамедлительно переходит в состояние Full. RT1 будет переходить в состояние Full после обновления нужных

частей своей базы данных. Это осуществляется путем передачи пакетов LSR и приема откликов LSU. Отметим, что ожидание маршрутизатором RT1 завершения приема всего набора пакетов DD от RT2 до передачи пакетов LSR не является обязательным. RT1 может чередовать передачу пакетов LSR и прием пакетов DD.

## 11. Структура таблицы маршрутов

Таблица маршрутов содержит все данные, требуемые для пересылки пакетов IP в направлении адресата. Каждая запись таблицы описывает набор лучших маршрутов к отдельному получателю. При пересылке пакетов данных IP определяется запись таблицы с максимальным соответствием IP-адресу получателя пакета. После этого из выбранной записи берется адрес следующего маршрутизатора (next hop) в направлении адресата. Протокол OSPF также обеспечивает поддержку принятого по умолчанию маршрута (Destination ID = DefaultDestination, Address Mask = 0x00000000). Если существует принятый по умолчанию маршрут, он соответствует всем IP-адресам получателей (хотя любые другие маршруты обеспечивают лучшее соответствие). Поиск маршрута с лучшим соответствием IP-адресу получателя описан в параграфе 11.1.

В каждом маршрутизаторе существует одна таблица маршрутов. Два примера таблиц маршрутизации рассмотрены в параграфах 11.2 и 11.3. Построение таблицы маршрутов описано в главе 16.

В остальной части параграфа описаны поля записей в таблице маршрутов. Первый набор полей описывает адресатов для записи.

### Destination Type

Адресатом может быть сеть (network) или маршрутизатор (router). При пересылке пакетов данных IP реально используются только записи для сетей. Связанные с маршрутизаторами записи таблицы применяются только на промежуточных этапах построения таблицы маршрутов.

Сеть представляет собой диапазон адресов IP, по которым могут пересылаться пакеты данных. Сюда включаются сети классов A, B и C, подсети IP, свержсети IP и отдельные хосты IP. Принятый по умолчанию маршрут также входит в эту категорию.

Записи, связанные с маршрутизаторами, сохраняются для граничных маршрутизаторов областей и AS. Записи в таблице маршрутов для граничных маршрутизаторов областей используются при расчете междоменных маршрутов (см. параграф 16.2) и для поддержки виртуальных соединений (см. главу 15), а записи для граничных маршрутизаторов AS служат для расчета внешних по отношению к AS маршрутов (см. параграф 16.4).

### Destination ID

Имя или идентификатор адресата в зависимости от значения Destination Type. Для сетей используется связанный с сетью IP-адрес, для маршрутизаторов - OSPF Router ID<sup>16</sup>.

### Address Mask

Маска определена только для сетей и при использовании вместе с адресом IP позволяет определить сетевой диапазон IP-адресов. Для подсетей IP адресную маску называют маской подсети. Маршруты к хостам имеют маску 0xffffffff.

### Optional Capabilities

Когда получателем является маршрутизатор, это поле показывает поддерживаемые им дополнительные возможности OSPF. В данной спецификации определена единственная дополнительная возможность – обработка AS-external-LSA. Обсуждение дополнительных возможностей OSPF приведено в параграфе 4.5.

<sup>16</sup> Пространство адресов IP-сетей и значений OSPF Router ID могут перекрываться, т. е., IP-адрес сети в 32-битовом представлении может совпадать с Router ID какого-либо из маршрутизаторов.



Набор путей к адресату может меняться в зависимости от областей OSPF, через которые проходит путь. Это означает, что для одного получателя в таблице может содержаться несколько записей в зависимости от значения следующего поля.

#### *Area*

Это поле указывает области, для которых информация о каналах приводит к включению в таблицу множества путей (эти записи называют связанными или ассоциированными). Для наборов внешних по отношению к AS путей это поле не определено. Для маршрутизаторов может существовать несколько наборов путей (и отдельных записей в таблице), связанных с каждой из нескольких областей. Например, это может произойти, когда два граничных маршрутизатора областей относятся к одной области. Для сетей сохраняется единственный набор путей, связанный с лучшей областью (предпочтительный маршрут).

Остальная часть маршрутной записи описывает набор путей к адресату. Перечисленные ниже поля относятся ко всему набору путей для одного адресата, т. е., все пути в одной записи таблицы маршрутов имеют одинаковый тип и стоимость (см. ниже).

#### *Path-type*

Существует 4 типа путей для передачи пакетов адресату: внутридоменные, междоменные, внешние типа 1 и 2 (в порядке снижения предпочтения). Внутридоменные пути используются для адресатов в одной из подключенных к маршрутизатору областей, междоменные пути ведут в другие области OSPF. Определение типа происходит на основе принятых summary-LSA. Внешние пути ведут в другие AS и детектируются на основе принятых AS-external-LSA.

#### *Cost*

Стоимость пути к адресату в единицах link state. Для всех путей кроме внешних типа 2 этот параметр описывает стоимость пути в целом. Для внешних путей типа 2 это поле описывает стоимость части пути внутри данной AS. Эта стоимость высчитывается как сумма стоимостей составляющих путь каналов.

#### *Type 2 cost*

Это поле применимо только для внешних путей типа 2 и показывает стоимость внешней части пути. Эта стоимость анонсируется граничным маршрутизатором AS и составляет большую часть общей стоимости пути. Например, внешний путь типа 2 с внешней стоимостью 5 всегда предпочтительнее пути типа 2 с внешней стоимостью 10, независимо от внутренней стоимости.

#### *Link State Origin*

Применяется только для внутридоменных путей и показывает запись LSA (router-LSA или network-LSA), напрямую указывающую адресата. Например, если адресатом является транзитная сеть, это будет network-LSA транзитной сети. Если адресатом является тупиковая сеть, это будет router-LSA для подключенного к ней маршрутизатора. Записи LSA определяются при расчете дерева кратчайших путей (см. параграф 16.1). Один адресат может указываться множеством LSA, однако схема tie-breaking (разрыв связей) сужает выбор до единственной записи LSA. Поле Link State Origin не используется протоколом OSPF, но применяется при расчете таблицы маршрутов в MOSPF (OSPF Multicast routing extensions).

При наличии множества однотипных путей с равной стоимостью (equal-cost paths – равноценные пути), эти пути сохраняются в одной записи таблицы маршрутов. Равноценные пути различаются следующими полями:

#### *Next hop*

Выходной интерфейс маршрутизатора, используемый для пересылки пакетов получателю. В широковещательных, Point-to-MultiPoint и NBMA-сетях поле next hop также включает IP-адрес первого маршрутизатора (если он есть) на пути к адресату.

#### *Advertising router*

Имеет смысл только для междоменных и внешних путей и содержит Router ID маршрутизатора, анонсирующего summary-LSA или AS-external-LSA для этого пути.

## 11.1. Просмотр таблицы маршрутов

При получении пакета данных IP маршрутизатор OSPF находит запись таблицы маршрутизации, наиболее точно соответствующую адресату. Из этой записи определяется выходной интерфейс маршрутизатора и следующий маршрутизатор для пересылки пакета. В этом параграфе описан процесс поиска наилучшего соответствия в таблице маршрутов для адреса получателя пакета.

До начала просмотра в таблицу должны быть вставлены «отвергнутые» записи для каждого активного диапазона адресов области в маршрутизаторе (см. параграф 3.5). Диапазон считается активным, если он включает по крайней мере одну сеть, доступную внутридоменным путем. Адресатом «отвергнутой» записи служит набор адресов, описываемый связанным с ним активным диапазоном и тип пути для каждой «отвергнутой» записи должен быть междоменным<sup>17</sup>.

Адресу получателя может соответствовать несколько записей маршрутной таблицы. В таких случаях лучшее соответствие обеспечивает запись таблицы с максимальным совпадением. Иными словами, лучшее совпадение обеспечивается записью с наиболее узким диапазоном IP-адресов<sup>18</sup>. Например, запись для пары 128.185.1.0, 0xfffff00 задает более точное соответствие, нежели 128.185.0.0, 0xffff0000. Принятый по умолчанию маршрут обеспечивает минимальное соответствие, поскольку он подходит для всех адресатов. Отметим, что для любой отдельной записи в таблице маршрутов может существовать множество путей. В таких случаях расчеты, описанные в параграфах 16.1, 16.2 и 16.4 всегда дают пути наиболее предпочтительного типа (см. главу 11).

Если в таблице не найдено соответствующей адресату записи или максимальное соответствие обеспечивает одна из «отвергнутых» записей, адресат считается недоступным и вместо пересылки пакет отбрасывается, а отправителю возвращается пакет ICMP destination unreachable (адресат недоступен).

## 11.2. Пример таблицы маршрутизации без областей

Вернемся к примеру автономной системы на рисунке 2. В этом примере не задано ни одной области OSPF. Метрика показана только для исходящих интерфейсов. Расчет таблицы маршрутизации для RT6 был описан в параграфе 2.2, а результаты этого расчета приведены в таблице 15. Адресаты в таблице обозначены буквами N (сети), R (маршрутизаторы) и H (хосты).

<sup>17</sup> "Отвержение" областей требуется для беспетлевого резюмирования маршрутов на границе области.

<sup>18</sup> При соответствии получателю двух диапазонов адресов предполагается, что один диапазон уже другого. Чтобы избавиться от этого допущения можно использовать несвязные (Non-contiguous) маски подсетей, но такие маски протокол OSPF не поддерживает.

В этом примере нет равноценных путей, а междоменные пути отсутствуют, поскольку нет областей. Маршрутизаторы RT5 и RT7 являются граничными для AS. Внутридоменные пути рассчитываются до маршрутизаторов RT5 и RT7. Это позволяет рассчитать внешние пути до адресатов, анонсируемых RT5 и RT7 (т. е., сетей N12, N13, N14 и N15). Предполагается, что все AS-external-LSA генерируемые RT5 и RT7 анонсируют внешнюю метрику типа 1. Это ведет к тому, что внешние пути типа 1 будут рассчитаны для адресатов в сетях N12-N15.

### 11.3. Пример таблицы маршрутизации с областями

Вернемся к предыдущему примеру, разделив AS на две области OSPF (конфигурация областей показана на рисунке 4). Таблица маршрутизации RT4 будет описана для этой конфигурации. Маршрутизатор RT4 соединен с областью 1 и магистральной областью. Это заставляет RT4 видеть AS как объединение (concatenation) двух графов, показанных в таблицах 6 и 7. Маршруты RT4 показаны в таблице 16.

Маршрутизаторы RT5 и RT7 являются граничными для AS, а RT3, RT4, RT7, RT10 и RT11 – для областей. Отметим, что для маршрутизатора RT3 в таблице две маршрутных записи, поскольку он имеет две области, общие с RT4 (область 1 и магистраль).

Магистральные пути рассчитываются до всех граничных маршрутизаторов областей. Эти пути служат для расчета междоменных маршрутов. Отметим, что все междоменные маршруты связаны с магистралью – это бывает всегда для случаев, когда рассчитывающий маршрутизатор является граничным для области. Маршрутная информация конденсируется на границах областей. В нашем примере предполагается, что область 3 определена так, что сети N9 - N11 и хост H1 находятся на одном маршруте, анонсируемом в магистраль маршрутизатором RT11. Отметим, что стоимость этого пути является максимальной из набора стоимостей отдельных компонент.

Между маршрутизаторами RT10 и RT11 организован виртуальный канал. Без этого канала маршрутизатор RT11 не сможет анонсировать маршрут для сетей N9 - N11 и хоста H1 в магистраль и в таблице RT4 не будет записи для этих сетей. В данном примере есть два равноценных пути в сеть N12, однако оба пути идут через один маршрутизатор next hop (RT5). Таблица маршрутизации RT4 улучшается (часть путей становится короче) при организации виртуального канала между RT4 и RT3. Новый виртуальный канал будет сам по себе ассоциироваться с первой записью в таблице граничного маршрутизатора области (RT3 в таблице 16) – внутридоменным путем через область 1. Это будет обеспечивать для виртуального канала стоимость 1. Изменения таблицы маршрутизации в связи с организацией виртуального канала показаны в таблице 17.

Между маршрутизаторами RT10 и RT11 организован виртуальный канал. Без этого канала маршрутизатор RT11 не сможет анонсировать маршрут для сетей N9 - N11 и хоста H1 в магистраль и в таблице RT4 не будет записи для этих сетей.

В данном примере есть два равноценных пути в сеть N12, однако оба пути идут через один маршрутизатор next hop (RT5). Таблица маршрутизации RT4 улучшается (часть путей становится короче) при организации виртуального канала между RT4 и RT3. Новый виртуальный канал будет сам по себе ассоциироваться с первой записью в таблице граничного маршрутизатора области (RT3 в таблице 16) – внутридоменным путем через область 1. Это будет обеспечивать для виртуального канала стоимость 1. Изменения таблицы маршрутизации в связи с организацией виртуального канала показаны в таблице 17.

## 12. Анонсы состояния каналов (LSA)

Таблица 17 Изменения в результате добавления виртуального канала.

Тип	Адресат	Область	Тип пути	Стоимость	Next Hop	Анонсирующий маршрутизатор
N	Ib	0	Внутридомен.	16	RT3	*
N	Ia	0	Внутридомен.	21	RT3	*
R	RT3	0	Внутридомен.	1	*	*
R	RT10	0	Внутридомен.	16	RT3	*
R	RT11	0	Внутридомен.	19	RT3	*
N	N9-N11,H1	0	Междоменный	30	RT3	RT11

свое назначение. Router-LSA и network-LSA описывают соединения между маршрутизаторами областей и сетями. Summary-LSA обеспечивает способ сжатого представления маршрутных данных области, а AS-external-LSA дают возможность прозрачного анонсирования внешней маршрутной информации внутри AS.

Таблица 15 Таблица маршрутизации RT6 (областей не задано).

Тип	Адресат	Область	Тип пути	Стоимость	Next Hop	Анонсирующий маршрутизатор
N	N1	0	Внутридомен.	10	RT3	*
N	N2	0	Внутридомен.	10	RT3	*
N	N3	0	Внутридомен.	7	RT3	*
N	N4	0	Внутридомен.	8	RT3	*
N	Ib	0	Внутридомен.	7	*	*
N	Ia	0	Внутридомен.	12	RT10	*
N	N6	0	Внутридомен.	8	RT10	*
N	N7	0	Междоменный	12	RT10	*
N	N8	0	Междоменный	10	RT10	*
N	N9	0	Междоменный	11	RT10	*
N	N10	0	Междоменный	13	RT10	*
N	N11	0	Междоменный	14	RT10	*
N	H1	0	Междоменный	21	RT10	*
R	RT5	0	Междоменный	6	RT5	*
R	RT7	0	Междоменный	8	RT10	*
N	N12	*	Внешн. типа 1	10	RT10	RT7
N	N13	*	Внешн. типа 1	114	RT5	RT5
N	N14	*	Внешн. типа 1	14	RT5	RT5
N	N15	*	Внешн. типа 1	17	RT10	RT7

Таблица 16 Таблица маршрутизации RT4 при наличии областей.

Тип	Адресат	Область	Тип пути	Стоимость	Next Hop	Анонсирующий маршрутизатор
N	N1	1	Внутридомен.	4	RT1	*
N	N2	1	Внутридомен.	4	RT2	*
N	N3	1	Внутридомен.	1	*	*
N	N4	1	Внутридомен.	3	RT3	*
R	RT3	1	Внутридомен.	1	*	*
N	Ib	0	Внутридомен.	22	RT5	*
N	Ia	0	Внутридомен.	27	RT5	*
R	RT3	0	Внутридомен.	21	RT5	*
R	RT5	0	Внутридомен.	8	*	*
R	RT7	0	Внутридомен.	14	RT5	*
R	RT10	0	Внутридомен.	22	RT5	*
R	RT11	0	Внутридомен.	25	RT5	*
N	N6	0	Междоменный	15	RT5	RT7
N	N7	0	Междоменный	19	RT5	RT7
N	N8	0	Междоменный	18	RT5	RT7
N	N9-N11,H1	0	Междоменный	36	RT5	RT11
N	N12	*	Внешн. типа 1	16	RT5	RT5, RT7
N	N13	*	Внешн. типа 1	16	RT5	RT5
N	N14	*	Внешн. типа 1	16	RT5	RT5
N	N15	*	Внешн. типа 1	23	RT5	RT7

Каждый маршрутизатор в AS генерирует один или несколько анонсов состояния каналов - LSA. В данной спецификации определены 5 различных типов LSA, описанных в параграфе 4.3. Набор LSA формирует базу данных о состоянии каналов. Каждый тип LSA имеет

Каждая запись LSA начинается стандартным 20-байтовым заголовком, описанным ниже.

## 12.1. Заголовок LSA

Заголовок LSA содержит поля type, Link State ID и Advertising Router. Комбинация этих трех полей уникальна для LSA.

В автономной системе может одновременно существовать несколько экземпляров LSA. В таких случаях требуется определять последний экземпляр, для чего проверяются поля LS sequence, LS checksum и LS age, содержащиеся в заголовке LSA.

Некоторые типы пакетов OSPF содержат списки LSA. Когда экземпляр не имеет значения, LSA идентифицируются полями LS type, Link State ID и Advertising Router (см. описание пакетов LSR). В остальных случаях принимаются во внимание также поля LS sequence number, LS age и LS. Детальное описание полей заголовка LSA приводится ниже.

### 12.1.1. Возраст LS

Это поле показывает возраст LSA в секундах и должно обрабатываться как беззнаковое 16-битовое целое. При генерации LSA поле имеет нулевое значение и должно увеличиваться на InfTransDelay при прохождении каждого интервала в процедуре лавинной рассылки. Старение LSA происходит также при их удержании в базе данных каждого маршрутизатора.

Возраст LSA не увеличивается сверх MaxAge и LSA с возрастом MaxAge не используются при расчете таблиц маршрутизации. Когда возраст LSA достигает MaxAge, для этой записи повторно используется лавинная рассылка. LSA с возрастом MaxAge окончательно удаляются из базы данных, когда они перестают быть нужными для синхронизации базы. Дополнительные сведения о старении LSA приведены в главе 14.

Поле LS age проверяется маршрутизатором при получении двух экземпляров LSA, имеющих одинаковый порядковый номер и контрольную суммы. Экземпляр с возрастом MaxAge в таких случаях всегда принимается как более новый, что позволяет быстрее удалять старые LSA из области маршрутизации. В остальных случаях, если разница в возрасте превышает MaxAgeDiff, экземпляра с меньшим возрастом принимается как более свежий<sup>19</sup> (см. подробности в параграфе 13.1).

### 12.1.2. Опции

Поле Options в заголовке LSA показывает дополнительные возможности, связанные с LSA. Дополнительные возможности OSPF описаны в параграфе 4.5. Единственная дополнительная возможность, определенная в данной спецификации представляется флагом E в поле Options. Нераспознанные биты поля Options должны иметь нулевые значения.

Бит E представляет OSPF ExternalRoutingCapability и должен устанавливаться во всех LSA, связанных с магистралью и нетупиковыми (non-stub) областями (см. параграф 3.6). Этот флаг должен также устанавливаться для всех AS-external-LSA. Во всех router-LSA, network-LSA и summary-LSA, связанных с тупиковыми областями, флаг должен быть сброшен. Для всех LSA установка бита E производится только в информационных целях и не влияет на расчет таблицы маршрутов.

### 12.1.3. Тип LS

Поле типа LS определяет формат и назначение LSA. LSA различных типов имеют разные имена (router-LSA, network-LSA и др.). Все типы LSA, определенных в данной спецификации, за исключением AS-external-LSA (LS типа 5), рассылаются лавинным способом через одну область. Анонсы AS-external-LSA рассылаются по всей AS, за исключением тупиковых областей (см. параграф 3.6). Каждый из типов LSA кратко описан в таблице 18.

### 12.1.4. Link State ID

Это поле идентифицирует часть области маршрутизации, которая будет описана в LSA. Значение поля Link State ID зависит от типа LSA (см. таблицу 19).

Для summary-LSA (тип 3) и AS-external-LSA (тип 5) Link State ID дополнительно может содержать несколько host-битов сети-получателя. Например, при генерации AS-external-LSA для сети 10.0.0.0 с маской 255.0.0.0 Link State ID может указывать на любую адрес в диапазоне от 10.0.0.0 до 10.255.255.255 включительно (хотя по возможности следует использовать 10.0.0.0). Установка хост-битов позволяет маршрутизатору генерировать отдельные LSA для двух сетей с одинаковым номером, но разными масками (см. Приложение E).

Когда LSA описывает сеть (типы LS 2, 3, 5), IP-номер сети легко определить, используя маску сети/подсети, содержащуюся в LSA, и значение Link State ID. Если LSA описывает маршрутизатор (типы LS 1 и 4), значение Link State ID всегда описывает OSPF Router ID. Для AS-external-LSA (тип 5), описывающих принятый по умолчанию маршрут, Link State ID=DefaultDestination (0.0.0.0).

### 12.1.5. Анонсирующий маршрутизатор

Это поле содержит идентификатор OSPF Router ID породившего LSA маршрутизатора. Для router-LSA это поле совпадает с полем Link State ID. Записи Network-LSA порождаются маршрутизаторами DR, Summary-LSA – граничными маршрутизаторами областей, а AS-external-LSA – граничными маршрутизаторами AS.

Таблица 18 Типы анонсов LSA протокола OSPF.

Тип LS	LSA	Описание
1	router-LSA.	Информация о состоянии интерфейсов маршрутизатора (см. параграф 12.4.1).
2	network-LSA	Описывает набор маршрутизаторов, подключенных к сети (см. параграф 12.4.2).
3 или 4	summary-LSA	Описывает внутридоменные маршруты и обеспечивает возможность конденсации маршрутных данных на границе области. Генерируемые граничными маршрутизаторами областей summary-LSA типа 3 описывают маршруты в сети, а summary-LSA типа 4 – к граничным маршрутизаторам AS.
5	AS-external-LSA	Генерируются граничными маршрутизаторами AS и описывают маршруты к адресатам за пределами AS. Эта запись может также содержать маршрут, принятый по умолчанию для AS.

Таблица 19 Значение Link State ID.

1	Router ID генерирующего маршрутизатора.
2	IP-адрес интерфейса маршрутизатора DR для сети.
3	IP-номер сети получателя.
4	Router ID для граничного маршрутизатора AS.
5	IP-номер сети получателя.

<sup>19</sup> MaxAgeDiff является архитектурной константой и показывает максимальное расхождение возраста (в секундах), которое может возникнуть для одного экземпляра LSA при лавинной рассылке в домене маршрутизации. Если разница в возрасте двух LSA превышает это значение, предполагается, что это два разных экземпляра LSA. Это может происходить в тех случаях, когда маршрутизатор был перезагружен и потерял информацию о порядковом номере предыдущей записи LSA (см. параграф 13.4).

### 12.1.6. Порядковый номер LS

Порядковый номер трактуется как 32-разрядное целое число со знаком и используется для обнаружения старых LSA и дубликатов. Пространство порядковых номеров линейно упорядочено и больший порядковый номер (при сравнении 32-битовых целых чисел со знаком) соответствует более новой записи LSA. Далее при описании порядковых номеров N будет обозначать  $2^{31}$ .

Порядковый номер -N (0x80000000) зарезервирован и не должен использоваться. Значение -N + 1 (0x80000001) является минимальным (следовательно, самым старым) порядковым номером и используется в качестве константы InitialSequenceNumber. Маршрутизатор использует номер InitialSequenceNumber при первой генерации LSA. После этого порядковые номера LSA увеличиваются на 1 при генерации каждого нового экземпляра LSA. При достижении порядкового номера N – 1 (0x7fffffff или MaxSequenceNumber), сначала нужно удалить текущий экземпляр LSA из домена маршрутизации. Для этого используется принудительное старение (prematurely aging) LSA (см. параграф 14.1) и повторная лавинная рассылка. После подтверждения этой рассылки всеми смежными маршрутизаторами возможна генерация новой записи с порядковым номером InitialSequenceNumber.

Для маршрутизатора может потребоваться форсирование анонса порядкового номера одной из его LSA при неожиданном приеме более нового экземпляра LSA в процессе лавинной рассылки. Это происходит редко и может говорить о наличии устаревших LSA, сгенерированных самим маршрутизатором при его последней загрузке, которые еще сохраняются в AS (см. параграф 13.4).

### 12.1.7. Контрольная сумма LS

Это поле включает контрольную сумму записи LSA без учета поля LS age для того, чтобы можно было увеличивать поле возраста без пересчета контрольной суммы. При расчете контрольных сумм используется тот же алгоритм, который применяется для дейтаграмм ISO без организации соединений – его часто называют алгоритмом Флэтчера. Этот алгоритм описан в работе [Ref6] (Annex B). Заголовок LSA содержит также размер LSA в байтах без учета поля LS age (два байта) для вычисления контрольной суммы.

Контрольная сумма служит для детектирования повреждений LSA, которые могут происходить при лавинной рассылке или хранении в памяти маршрутизатора. Поле контрольной суммы не может быть нулевым, значение 0 должно рассматриваться как ошибка. Иными словами, вычисление контрольной суммы является обязательным.

Контрольная сумма LSA проверяется в двух случаях: а) при получении пакета Link State Update и б) во время «состаривания» базы данных. Обнаружение ошибок в контрольной сумме ведет к различным ситуациям, описанным в главах 13 и 14.

Когда поля порядковых номеров двух записей LSA совпадают, сравниваются контрольные суммы. Если суммы различаются, более новым считается экземпляр с большим значением контрольной суммы<sup>20</sup> (см. параграф 13.1).

## 12.2. База данных link-state

Маршрутизатор имеет отдельную базу данных о каналах для каждой области, к которой он относится и все маршрутизаторы одной области имеют идентичные базы данных.

Базы данных каждой области всегда обрабатываются отдельно. Расчет кратчайших путей выполняется отдельно для каждой области (см. главу 16). Лавинная рассылка компонент базы данных области ведется только в пределах области. И, наконец, при организации отношений смежности между двумя маршрутизаторами синхронизируются только базы данных общей области.

База данных области состоит из router-LSA, network-LSA и summary-LSA (все они перечислены в структуре данных области). Для нетупиковых областей в базу данных включаются также AS-external-LSA (см. параграф 3.6).

Реализация протокола OSPF должна обеспечивать возможность доступа к отдельным частям базы данных области. Эта функция просмотра основана на полях LS type, Link State ID и Advertising Router<sup>21</sup>. Поиск будет давать один экземпляр (самый новый) каждой записи LSA в базе данных. Функции поиска в базе данных используются процедурой лавинной рассылки LSA (глава 13) и при расчете таблицы маршрутов (глава 16). Кроме того, используя эту функцию, маршрутизатор может определить не он ли породил данную запись LSA и (если он) с каким порядковым номером.

Запись LSA добавляется в базу данных маршрутизатора, если а) она получена в процессе лавинной рассылки (глава 13) или б) происходит о данного маршрутизатора (параграф 12.4). LSA удаляются из базы данных маршрутизатора, если а) получен более новый экземпляр в процессе лавинной рассылки (глава 13), б) маршрутизатор сам генерирует новый экземпляр LSA (параграф 12.4) или в) LSA удаляется из домена маршрутизации в результате старения (глава 14). При удалении LSA из базы данных, эта запись должна удаляться также из списков Link state retransmission всех соседей (см. главу 10).

## 12.3. Представление TOS

Для обратной совместимости с предыдущими версиями OSPF [Ref9], в router-LSA, summary-LSA и AS-external-LSA может включаться информация, связанная с TOS. Представление TOS в OSPF LSA описано в таблице 20, содержащей коды OSPF для представления полей TOS (определены в работе [Ref12]) пакетов IP. В OSPF используется десятичное представление, а в заголовках IP применяются битовые поля TOS, описанные в [Ref12].

<sup>20</sup> Когда контрольные суммы двух LSA различаются, предполагается, что это два разных экземпляра. Это может быть следствием перезагрузки маршрутизатора или потери маршрутизатором данных о предыдущем порядковом номере LSA. Если порядковые номера двух LSA совпадают, невозможно определить, которая из записей новее. Однако если в качестве новой будет принята некорректная запись LSA, генерирующий их маршрутизатор просто будет порождать другой экземпляр (см. параграф 13.4).

<sup>21</sup> Существует ситуация, когда поиск должен основываться на частичной информации. Это происходит при расчете таблицы маршрутов, когда нужно найти network-LSA, основываясь исключительно на Link State ID. Результаты поиска однозначны, поскольку не существует двух two network-LSA с одинаковыми полями Link State ID.



## 12.4. Генерация LSA

В каждой области OSPF маршрутизатор будет генерировать несколько типов LSA. Каждый маршрутизатор генерирует router-LSA. Если маршрутизатор является DR для любой из сетей области, он будет генерировать network-LSA для этой сети.

Граничные маршрутизаторы областей генерируют одну запись summary-LSA для каждого адресата с междоменным маршрутом. Граничные маршрутизаторы AS генерируют одну запись AS-external-LSA для каждого известного адресата за пределами AS. Адресаты анонсируются по одному, поэтому изменение отдельного маршрута не требует повтора лавинной рассылки для всего набора маршрутов. В процессе лавинной рассылки один пакет LSU может включать множество LSA.

В качестве примера рассмотрим маршрутизатор RT4 на рисунке 4, являющийся граничным в области 1 и связанный с магистралью. Маршрутизатор RT4 генерирует 5 разных LSA в магистраль (router-LSA и summary-LSA для каждой из сетей N1-N4). Маршрутизатор RT4 будет также порождать 8 различных LSA для области 1 (router-LSA и семь summary-LSA, показанных в таблице 6). Если RT4 будет выбран DR для сети N3, он будет генерировать в область 1 network-LSA для N3.

Маршрутизатор RT5 будет генерировать AS-external-LSA (для каждой из сетей N12 - N14). Эти записи будут рассылаться в лавинном режиме по всей AS, поскольку ни одна из областей не предполагается тупиковой. Однако если область 3 указать как тупиковую, AS-external-LSA для сетей N12-N14 не будут рассылаться в область 3 (см. параграф 3.6). Взамен этого маршрутизатор RT11 будет генерировать запись summary-LSA с принятым по умолчанию маршрутом, которая будет рассылаться в области 3 (см. параграф 12.4.3). В результате все внутренние маршрутизаторы области 3 будут передавать внешний трафик маршрутизатору RT11.

При создании нового экземпляра LSA увеличивается порядковый номер, устанавливается LS age=0, вычисляется контрольная сумма и LSA добавляется в базу данных, после чего происходит лавинная рассылка через соответствующие интерфейсы. Включение LSA в базу данных подробно описано в параграфе 13.2, а параграф 13.3 посвящен лавинной рассылке новых экземпляров LSA.

Ниже перечислены 10 событий, которые могут приводить к генерации новых экземпляров LSA:

- (1) Поле LS age одной из порожденных маршрутизатором LSA достигло значения LSRefreshTime. В таких случаях генерируется новый экземпляр LSA, несмотря на отсутствие изменений одержимого LSA (кроме заголовка). Это гарантирует периодическое обновление всех LSA и повышает устойчивость алгоритма link state. LSA, описывающие только недоступных адресатов, не должны обновляться и их нужно удалять из области маршрутизации (см. параграф 14.1).

Когда изменяется что-то из описываемого LSA, генерируется новый анонс LSA, однако в течение MinLSInterval не может порождаться более одного экземпляра LSA. Поэтому генерация нового экземпляра может быть задержана на время до MinLSInterval. Перечисленные ниже события могут приводить к изменениям содержимого LSA и должны порождать генерацию новых экземпляров тогда и только тогда, когда новая запись LSA будет отличаться от прежней.

- (2) Изменилось состояние интерфейса (см. параграф 9.1) и может потребоваться генерация нового экземпляра router-LSA.
- (3) Сменился маршрутизатор DR. Требуется генерация нового анонса router-LSA. Если данный маршрутизатор стал DR, нужно генерировать новые network-LSA. Если данный маршрутизатор перестал быть DR, все network-LSA, которые он мог породить для сети, должны быть удалены из домена маршрутизации (см. параграф 14.1).
- (4) Один из соседей перешел в состояние FULL или покинул его. Это требует генерации новых экземпляров router-LSA. Если данный маршрутизатор является DR для подключенной сети, он должен породить новый экземпляр network-LSA.

Следующие 4 события относятся только к граничным маршрутизаторам областей:

- (5) В таблице маршрутизации был изменен/добавлен/удален внутridoменный маршрут и может потребоваться генерация нового экземпляра summary-LSA (для этого маршрута) в каждую подключенную область (возможно и магистральную).
- (6) В таблице маршрутизации был изменен/добавлен/удален междоменный маршрут и может потребоваться генерация нового экземпляра summary-LSA (для этого маршрута) в каждую подключенную область (кроме магистральной).
- (7) К области подключен новый маршрутизатор и он должен породить summary-LSA в подключенную недавно область для всех имеющих отношение внутridoменных и междоменных маршрутов в таблице маршрутизатора (см. параграф 12.4.3).
- (8) Изменилось состояние одного из маршрутизаторов с настроенным виртуальным каналом и может потребоваться генерация новых router-LSA в транзитную область виртуального канала (см. описание бита V записи router-LSA в параграфе 12.4.1), а также в магистраль.

Последние 2 события относятся только к граничным маршрутизаторам AS (включая бывшие):

- (9) Изменился внешний маршрут, полученный от другого протокола маршрутизации (например, BGP) и от граничного маршрутизатора AS может потребоваться генерация нового экземпляра AS-external-LSA.
- (10) Маршрутизатор перестал быть граничным в AS (возможно после перезагрузки) и требуется удаление всех порожденных ранее AS-external-LSA (для этого используется принудительное старение, описанное в параграфе 14.1).

Подробное описание структуры всех типов LSA приведено ниже. В следующих разделах описывается содержимое LSA, а 20-байтовые заголовки LSA описаны в параграфе 12.1.

### 12.4.1. LSA для маршрутизатора

Маршрутизатор генерирует router-LSA для каждой области, к которой он относится. LSA описывает состояния каналов маршрутизатора в область. Анонсы LSA рассылаются в лавинном режиме внутри области, не покидая ее.

Маршрутизатор также показывает свою принадлежность к граничным маршрутизаторам области или AS, устанавливая биты B или E, соответственно, в router-LSA. Это позволяет сохранять пути к таким маршрутизаторам в таблице для последующей обработки summary-LSA и AS-external-LSA. Бит B должен устанавливаться, если маршрутизатор имеет активные соединения с двумя или более областями, даже если этот маршрутизатор не подключен к магистральной области OSPF. Бит E никогда не должен устанавливаться в router-LSA для тупиковых областей (такие области не могут включать граничные маршрутизаторы AS). В дополнение к этому маршрутизатор устанавливает бит V в router-LSA для области A тогда и только тогда, когда маршрутизатор является конечной точкой для одного или нескольких виртуальных каналов с полной смежностью, использующих область A в качестве транзитной. Установка бита V позволяет другим маршрутизаторам области A обнаруживать поддерживаемый областью трафик (см. TransitCapability в главе 6).

Таблица 20 Представление TOS в OSPF.

Код OSPF	Значения TOS RFC 1349
0	0000 нормальное обслуживание
2	0001 минимальная оплата
4	0010 максимальная надежность
6	0011
8	0100 максимальная пропускная способность
10	0101
12	0110
14	0111
16	1000 минимальная задержка
18	1001
20	1010
22	1011
24	1100
26	1101
28	1110
30	1111

Маршрутизатор RT11 будет генерировать запись summary-LSA с принятым по умолчанию маршрутом, которая будет рассылаться в области 3 (см. параграф 12.4.3). В результате все внутренние маршрутизаторы области 3 будут передавать внешний трафик маршрутизатору RT11.

При создании нового экземпляра LSA увеличивается порядковый номер, устанавливается LS age=0, вычисляется контрольная сумма и LSA добавляется в базу данных, после чего происходит лавинная рассылка через соответствующие интерфейсы. Включение LSA в базу данных подробно описано в параграфе 13.2, а параграф 13.3 посвящен лавинной рассылке новых экземпляров LSA.

Ниже перечислены 10 событий, которые могут приводить к генерации новых экземпляров LSA:

- (1) Поле LS age одной из порожденных маршрутизатором LSA достигло значения LSRefreshTime. В таких случаях генерируется новый экземпляр LSA, несмотря на отсутствие изменений одержимого LSA (кроме заголовка). Это гарантирует периодическое обновление всех LSA и повышает устойчивость алгоритма link state. LSA, описывающие только недоступных адресатов, не должны обновляться и их нужно удалять из области маршрутизации (см. параграф 14.1).

Когда изменяется что-то из описываемого LSA, генерируется новый анонс LSA, однако в течение MinLSInterval не может порождаться более одного экземпляра LSA. Поэтому генерация нового экземпляра может быть задержана на время до MinLSInterval. Перечисленные ниже события могут приводить к изменениям содержимого LSA и должны порождать генерацию новых экземпляров тогда и только тогда, когда новая запись LSA будет отличаться от прежней.

- (2) Изменилось состояние интерфейса (см. параграф 9.1) и может потребоваться генерация нового экземпляра router-LSA.
- (3) Сменился маршрутизатор DR. Требуется генерация нового анонса router-LSA. Если данный маршрутизатор стал DR, нужно генерировать новые network-LSA. Если данный маршрутизатор перестал быть DR, все network-LSA, которые он мог породить для сети, должны быть удалены из домена маршрутизации (см. параграф 14.1).
- (4) Один из соседей перешел в состояние FULL или покинул его. Это требует генерации новых экземпляров router-LSA. Если данный маршрутизатор является DR для подключенной сети, он должен породить новый экземпляр network-LSA.

Следующие 4 события относятся только к граничным маршрутизаторам областей:

- (5) В таблице маршрутизации был изменен/добавлен/удален внутridoменный маршрут и может потребоваться генерация нового экземпляра summary-LSA (для этого маршрута) в каждую подключенную область (возможно и магистральную).
- (6) В таблице маршрутизации был изменен/добавлен/удален междоменный маршрут и может потребоваться генерация нового экземпляра summary-LSA (для этого маршрута) в каждую подключенную область (кроме магистральной).
- (7) К области подключен новый маршрутизатор и он должен породить summary-LSA в подключенную недавно область для всех имеющих отношение внутridoменных и междоменных маршрутов в таблице маршрутизатора (см. параграф 12.4.3).
- (8) Изменилось состояние одного из маршрутизаторов с настроенным виртуальным каналом и может потребоваться генерация новых router-LSA в транзитную область виртуального канала (см. описание бита V записи router-LSA в параграфе 12.4.1), а также в магистраль.

Последние 2 события относятся только к граничным маршрутизаторам AS (включая бывшие):

- (9) Изменился внешний маршрут, полученный от другого протокола маршрутизации (например, BGP) и от граничного маршрутизатора AS может потребоваться генерация нового экземпляра AS-external-LSA.
- (10) Маршрутизатор перестал быть граничным в AS (возможно после перезагрузки) и требуется удаление всех порожденных ранее AS-external-LSA (для этого используется принудительное старение, описанное в параграфе 14.1).

Подробное описание структуры всех типов LSA приведено ниже. В следующих разделах описывается содержимое LSA, а 20-байтовые заголовки LSA описаны в параграфе 12.1.

Таблица 21 Описание каналов в router-LSA.

Тип	Описание	Идентификатор
1	«Точка-точка»	Neighbor Router ID
2	Канал в транзитную сеть	Адрес интерфейса DR
3	Канал в оконечную сеть	Номер IP-сети
4	Виртуальный канал	Neighbor Router ID

Router-LSA описывает работающие соединения (интерфейсы или каналы) маршрутизатора с областью. Тип каждого канала зависит от подключенной сети. Каждый канал имеет также идентификатор Link ID, используемый в качестве обозначения объекта на другой стороне соединения. В таблице 21 приведены значения полей Type и Link ID.

Для каждого канала используется также поле Link Data, содержащее 32 бита дополнительной информации о канале. Для каналов в транзитные сети, и адресуемых соединений «точка-точка» это поле содержит IP-адрес интерфейса маршрутизатора (он нужен для расчета таблицы маршрутов, описанного в параграфе 16.1.1). Для каналов в тупиковые сети это поле задает маску тупиковой сети. Для безадресных каналов point-to-point поле Link Data должно содержать значение MIB-II unnumbered-интерфейса - ifIndex [Ref8]. Наконец, задается стоимость использования канала для исходящего трафика. Стоимость канала может настраиваться и для всех каналов, кроме случая тупиковых сетей, стоимость должна быть больше 0.

Для дальнейшего рассмотрения процесса построения списка описаний каналов рассмотрим случай создания маршрутизатором анонса router-LSA для области A. Для этого маршрутизатор проверяет набор структур данных интерфейсов, выполняя для каждого интерфейса следующие операции:

- ◆ Если подключенная сеть не относится к области A, в LSA не добавляется каналов и проверяется следующий интерфейс.
- ◆ Если интерфейс находится в состоянии Down, каналов в LSA не добавляется.
- ◆ Для интерфейсов в состоянии Loopback добавляется канал типа 3 (тупиковая сеть), если этот интерфейс не ведет в безадресную сеть point-to-point. В поле Link ID должен помещаться IP-адрес интерфейса, а в поле Link Data – маска 0xffffffff (маршрут к хосту). Стоимость канала равна 0.
- ◆ В остальных случаях в router-LSA добавляется описание канала в зависимости от типа интерфейса OSPF. Описание канала для интерфейсов point-to-point приведено в параграфе 12.4.1.1, для виртуальных соединений – в параграфе 12.4.1.2, для широковещательных и NBMA-интерфейсов – в 12.4.1.3 и для интерфейсов Point-to-MultiPoint – в параграфе 12.4.1.4.

После рассмотрения всех интерфейсов маршрутизатора в router-LSA добавляются соединения с хостами путем проверки подключенных хостов, относящихся к области A. Маршруты хостам относятся к типу 3 (тупиковая сеть), поле Link ID содержит IP-адрес хоста, Link Data – маску 0xffffffff, а стоимость маршрута должна быть больше 0 (см. параграф C.7).

#### 12.4.1.1. Описание интерфейсов point-to-point

Для интерфейсов point-to-point в router-LSA добавляется одно или несколько описаний каналов:

- ◆ Если соседний маршрутизатор является полностью смежным, добавляется канал типа 1 (point-to-point). Поле Link ID должно содержать Router ID соседнего маршрутизатора. Для адресуемых сетей point-to-point поле Link Data должно содержать IP-адрес интерфейса, а для unnumbered-сетей – значение ifIndex (MIB-II [Ref8]). Стоимость канала должна совпадать с выходной стоимостью интерфейса point-to-point.
- ◆ В дополнение к этому, если интерфейс находится в состоянии Point-to-Point (независимо от состояния соседнего маршрутизатора), должен добавляться канал типа 3 (тупиковая сеть). Этот тупиковый канал может принимать две формы:

##### Опция 1

Предполагая, что IP-адрес соседнего маршрутизатора известен, устанавливаем для поля Link ID (тип 3) значение этого адреса IP, в поле Link Data помещаем маску 0xffffffff (маршрут к хосту), а стоимость устанавливаем больше нуля<sup>22</sup>.

##### Опция 2

Если для канала «точка-точка» выделена подсеть, устанавливаем в поле Link ID типа 3 IP-адрес подсети, в поле Link Data – ее маску и задаем положительное значение для стоимости<sup>23</sup>. [16]

#### 12.4.1.2. Описание интерфейсов в широковещательные и NBMA-сети

Для широковещательных и NBMA-интерфейсов в router-LSA добавляется одно описание канала:

- ◆ Если интерфейс находится в состоянии Waiting, добавляется канал типа 3 (тупиковая область), поле Link ID содержит IP-номер подключенной сети, Link Data – маску сети, а стоимость совпадает с выходной стоимостью интерфейса.
- ◆ Для остальных случаев должен быть известен выделенный маршрутизатор DR для подключенной сети. Если маршрутизатор является полностью смежным с DR или сам является DR и имеет полностью смежный маршрутизатор, добавляется канал типа 2 (транзитная сеть), поле Link ID содержит IP-адрес интерфейса DR (возможно интерфейс данного маршрутизатора), Link Data содержит IP-адрес интерфейса данного маршрутизатора, а стоимость совпадает с выходной стоимостью для интерфейса. Во всех прочих случаях добавляется один канал как для описанного выше состояния Waiting.

#### 12.4.1.3. Описание виртуальных соединений

Для виртуальных каналов описание добавляется в router-LSA только при полной смежности виртуальных соседей. В таких случаях добавляется канал типа 4 (виртуальный), Link ID = Router ID для виртуального соседа, Link Data содержит IP-адрес интерфейса, связанного с виртуальным каналом, а стоимость совпадает со стоимостью виртуального канала, рассчитанной при создании таблицы маршрутизации (глава 15).

#### 12.4.1.4. Описание интерфейсов Point-to-MultiPoint

Для интерфейсов Point-to-MultiPoint в router-LSA добавляется один или несколько каналов:

- ◆ Добавляется один канал типа 3 (тупиковая сеть), Link ID содержит IP-адрес интерфейса данного маршрутизатора, Link Data = 0xffffffff (маска хоста), а стоимость равна нулю.
- ◆ Для каждого полностью смежного соседа, связанного с интерфейсом, добавляется канал типа 1 (point-to-point), Link ID = Router ID соседнего маршрутизатора, поле Link Data содержит IP-адрес интерфейса, а стоимость равна выходной стоимости для интерфейса.

#### 12.4.1.5. Примеры router-LSA

Рассмотрим анонс router-LSA, генерируемые RT3 (см. рисунок 4). Область, содержащая RT3 (Area 1) с реальными сетевыми адресами показана на рисунке 11. Предположим, что последний байт адреса всех интерфейсов маршрутизатора RT3 равен 3 (адреса 192.1.1.3 и 192.1.4.3) и в других маршрутизаторах используется аналогичная схема адресации. Кроме того, предположим, что все соединения функционируют и в качестве Router ID используется меньший из IP-адресов интерфейсов.

RT3 генерирует два анонса router-LSA – для области 1 и для магистрали. Предположим, что маршрутизатор RT4 выбран в качестве DR для сети 192.1.1.0. Анонс router-LSA маршрутизатора RT3 для области 1 при оговоренных условиях показан ниже. Он показывает, что RT3 имеет два соединения с областью 1 – одно в транзитную сеть 192.1.1.0 и другое в тупиковую сеть 192.1.4.0.

<sup>22</sup> Это способ представляет канал «точка-точка» в соответствии с RFC 1583. Данный способ обеспечивает ряд преимуществ: а) не требуется выделения подсети для каналов «точка-точка», б) пакеты, адресованные интерфейсу point-to-point, будут реально приниматься через интерфейс (это полезно для диагностики) и в) поддерживается загрузка соседей через сеть (network bootstrapping) без необходимости поддержки bootstrap-загрузки реализацией протокола OSPF.

<sup>23</sup> Это представление каналов point-to-point более традиционно и используется протоколами типа RIP.

Отметим, что транзитная сеть идентифицируется IP-адресом интерфейса DR (т. е., Link ID = 192.1.1.4 – адрес интерфейса DR-маршрутизатора RT4 в сеть 192.1.1.0). Отметим также, что RT3 является граничным маршрутизатором области.

```
; router-LSA маршрутизатора RT3 для области 1
LS age = 0 ; всегда верно при создании
Options = (E-bit) ;
LS type = 1 ; показывает router-LSA
Link State ID = 192.1.1.3 ; Router ID для RT3
Advertising Router = 192.1.1.3 ; Router ID для RT3
bit E = 0 ; не является граничным для AS
bit B = 1 ; граничный маршрутизатор области
#links = 2
    Link ID = 192.1.1.4 ; IP-адрес интерфейса DR
    Link Data = 192.1.1.3 ; IP-адрес интерфейса RT3 в сеть
    Type = 2 ; соединение с транзитной сетью
# TOS metrics = 0
    metric = 1
    Link ID = 192.1.4.0 ; IP-номер сети
    Link Data = 0xffffffff00 ; маска сети
    Type = 3 ; соединение с тупиковой сетью
# TOS metrics = 0
    metric = 2
```

Далее показан анонс router-LSA маршрутизатора RT3 для магистрали. Он говорит, что RT3 имеет одно соединение с магистралью через безадресный канал point-to-point к маршрутизатору RT6. Показано также что RT3 является граничным для области.

```
; router-LSA маршрутизатора RT3 для магистрали
LS age = 0 ; всегда верно при создании
Options = (E-bit) ;
LS type = 1 ; показывает router-LSA
Link State ID = 192.1.1.3 ; Router ID для RT3
Advertising Router = 192.1.1.3 ; Router ID для RT3
bit E = 0 ; не является граничным для AS
bit B = 1 ; граничный маршрутизатор области
#links = 1
    Link ID = 18.10.0.6 ; Router ID соседнего маршрутизатора
    Link Data = 0.0.0.3 ; MIB-II ifIndex для канала «точка-точка»
    Type = 1 ; соединение с маршрутизатором
# TOS metrics = 0
    metric = 8
```

## 12.4.2. LSA для сети

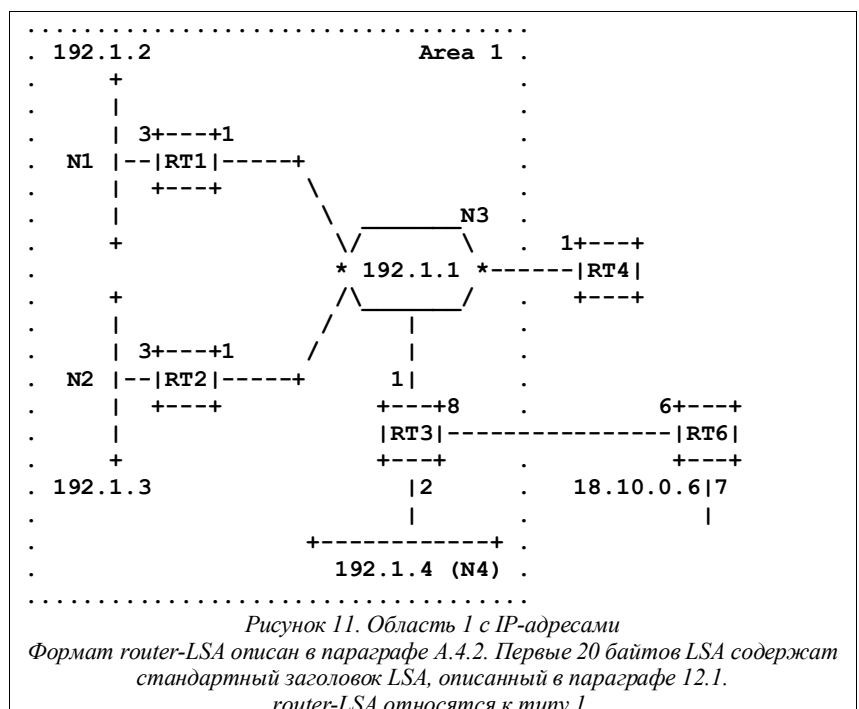
Анонсы network-LSA генерируются для каждой транзитной ширококвещательной или NBMA-сети (транзитной считается сеть с числом маршрутизаторов не менее 2) и описывает все маршрутизаторы, подключенные к сети. Эти анонсы порождаются маршрутизатором DR для сети. Для генерации LSA маршрутизатор DR должен иметь полную смежность по крайней мере с одним маршрутизатором сети. Анонсы network-LSA рассылаются в лавинном режиме по области, содержащей транзитную сеть, не выходя за пределы области. В network-LSA перечисляются маршрутизаторы полностью смежные с DR и каждый такой маршрутизатор идентифицируется OSPF Router ID. Маршрутизатор DR также содержится в списке.

Идентификатор Link State ID для network-LSA совпадает с IP-адресом интерфейса DR. Это значение после применения маски подсети, которая также включена в network-LSA дает номер IP-сети. Маршрутизатор, который утратил роль DR для сети, должен уничтожить ранее созданные network-LSA, которые больше не могут использоваться при расчете таблицы маршрутов. Это осуществляется с помощью принудительного увеличения возраста LSA до значения MaxAge и новой лавинной рассылки (см. параграф 14.1). Кроме того, в редких случаях, когда изменяется Router ID, все network-LSA с прежним значением Router ID также должны уничтожаться. Поскольку маршрутизатор может не знать о существовании с другим Router ID, эти network-LSA идентифицируются значением Link State ID, совпадающим с IP-адресом интерфейса маршрутизатора и полем Advertising Router, имеющим такое же значение и не совпадающим с текущим значением Router ID (см. параграф 13.4).

### 12.4.2.1. Примеры network-LSA

Вернемся к примеру сети на рисунке 4. Анонсы Network-LSA генерируются для сети N3 в области 1, сетей N6 и N8 в области 2 и сети N9 в области 3. Предположим, что RT4 выбран в качестве DR для сети N3 – в этом случае маршрутизатор RT4 породит анонс network-LSA для сети N3, показанный ниже (см. распределение адресов на рисунке 11):

```
; Network-LSA для сети N3
```





```

LS age = 0 ; всегда верно при создании
Options = (E-bit) ;
LS type = 2 ; указывает network-LSA
Link State ID = 192.1.1.4 ; IP-адрес DR
Advertising Router = 192.1.1.4 ; Router ID для RT4
Network Mask = 0xffffffff00
    Attached Router = 192.1.1.4 ; Router ID
    Attached Router = 192.1.1.1 ; Router ID
    Attached Router = 192.1.1.2 ; Router ID
    Attached Router = 192.1.1.3 ; Router ID
    
```

### 12.4.3. Summary-LSA

Анонсы summary-LSA описывают IP-сети, граничные маршрутизаторы AS и диапазоны адресов IP. Анонсы Summary-LSA рассылаются в лавинном режиме в пределах одной области. Описываемые адресаты находятся за пределами области, но входят в AS.

Генераторами Summary-LSA являются граничные маршрутизаторы областей. Точные маршруты для анонсирования в область определяются проверкой структуры маршрутной таблицы (см. главу 11) в соответствии с описанным ниже алгоритмом. Отметим, что в магистральную область анонсируются только внутридоменные маршруты, а в остальные области могут анонсироваться и междоменные маршруты.

Для определения маршрутов, анонсируемых в подключенную область А, каждая запись таблицы обрабатывается в соответствии с приведенным ниже алгоритмом. Напомним, что каждая запись маршрутной таблицы описывает набор равноценных путей к определенному адресату:

- ♦ В качестве Destination Type анонсов summary-LSA могут служить только сети и граничные маршрутизаторы AS. Если в записи таблицы маршрутов в качестве адресата указан граничный маршрутизатор области, обработка записи заканчивается.
- ♦ Внешние маршруты AS никогда не анонсируются в summary-LSA. Если запись таблицы включает внешний путь типа 1 или 2, обработка записи на этом завершается.
- ♦ Далее, если областью, связанной с набором путей, является сама область А, для этого маршрута не создается summary-LSA<sup>24</sup>.
- ♦ Далее, если связанные с записью маршрутизаторы next hop относятся к области А, для маршрута не создается summary-LSA<sup>25</sup> (это является логическим эквивалентом split horizon в алгоритмах Distance Vector).
- ♦ Далее, если стоимость пути для записи не меньше значения LSInfinity, summary-LSA не создается для такого маршрута.
- ♦ В остальных случаях, если адресатом является граничный маршрутизатор AS, анонс summary-LSA должен генерироваться тогда и только тогда, когда запись таблицы маршрутов описывает предпочтительный путь к граничному маршрутизатору AS (см. шаг 3 в параграфе 16.4). Если это так, для адресата генерируется summary-LSA типа 4, в которой поле Link State ID содержит Router ID граничного маршрутизатора AS, а метрика равна стоимости маршрута. Отметим, что такие LSA не должны генерироваться, если область А указана как тупиковая.
- ♦ В противном случае Destination type указывает сеть. Для междоменных маршрутов создается summary-LSA типа 3, поле Link State ID содержит адрес сети (при необходимости Link State ID может включать один или несколько битов адреса хоста, как сказано в Приложении E) и метрика равна стоимости маршрута в таблице.
- ♦ Последним случаем остается внутридоменный маршрут в сеть. Это означает, что указанная в маршруте сеть относится к области, напрямую подключенной к маршрутизатору. В общем случае эта информация может быть сжата до включения в summary-LSA. Помните, что область имеет список диапазонов адресов, каждый элемент которого содержит пару [адрес, маска] и данные о состоянии (Advertise или DoNotAdvertise). Для каждого диапазона создается по крайней мере один анонс summary-LSA типа 3. Когда для состояния диапазона указано Advertise (анонсировать), генерируется summary-LSA типа 3, поле Link State ID содержит диапазон адресов (при необходимости в Link State ID могут включаться биты из «адреса хоста», как описано в Приложении E), а стоимость равна наибольшей из стоимостей сетей-компонент. Для состояния DoNotAdvertise генерация summary-LSA типа 3 не используется и сети-компоненты остаются скрытыми для других областей. По умолчанию, если сеть не содержит указанного явно диапазона адресов, генерируются summary-LSA типа 3, в которых поле Link State ID содержит адрес сет (при необходимости могут включаться биты из адреса хоста, как описано в Приложении E), а метрика совпадает со стоимостью маршрута в таблице. Если область может поддерживать транзитный трафик (TransitCapability = TRUE), маршрутная информация о магистральных сетях не должна сжиматься до резюмирования в область. Не должны также подавляться анонсы магистральных сетей в область. Иными словами, заданные для магистрали диапазоны адресов должны игнорироваться при генерации summary-LSA в транзитные области.

Если маршрутизатор анонсирует summary-LSA для недоступного адресата, он должен удалить такие LSA из домена маршрутизации, устанавливая для них возраст MaxAge и повторяя лавинную рассылку (см. параграф 14.1). Также и в случаях, когда адресат по-прежнему доступен, но, тем не менее, не может больше анонсироваться согласно описанной выше процедуре (например, сейчас это междоменный маршрут, который при использовании был внутридоменным маршрутом, связанным с той или иной немагистральной областью – такой маршрут не может больше анонсироваться в магистраль), анонсы LSA должны удаляться из домена.

#### 12.4.3.1. Генерация summary-LSA в тупиковые области

Описанный в параграфе 12.4.3 алгоритм является обязательным, если область А является тупиковой. Граничные маршрутизаторы областей, подключенные к тупиковым областям могут генерировать summary-LSA в эти области, согласно алгоритму, описанному в параграфе 12.4.3, или создавать только подмножество summary-LSA, для задания которого можно использовать параметры конфигурации. Снижение числа генерируемых LSA уменьшает базу каналов тупиковой области и снижает потребности в ресурсах маршрутизатора. Однако, пропуск LSA может приводить к неоптимальной маршрутизации между областями.

Как сказано в параграфе 12.4.3, summary-LSA типа 4 (ASBR-summary-LSA) никогда не генерируются в тупиковые области.

<sup>24</sup> Междоменные маршруты не анонсируются в магистраль, поскольку они всегда связаны с магистральной областью.

<sup>25</sup> Это требование актуально только для случаев, когда немагистральная область А поддерживает транзитные данные (т. е., TransitCapability = TRUE). Например, в конфигурации, показанной на рисунке 4, область 2 может поддерживать транзитный трафик за счет существования виртуального канала между RT10 и RT11. В результате от маршрутизатора RT11 требуется генерация только одного анонса summary-LSA в область 2 (схлопнувшийся адресат N9-N11,H1), поскольку все другие подходящие маршруты RT11 имеет следующий маршрутизатор в самой области 2 и должны анонсироваться только граничными маршрутизаторами других областей (в примере - RT10 и RT7).



В тупиковых областях вместо импортированных внешних маршрутов каждый граничный маршрутизатор области генерирует в область default summary-LSA. Поле Link State ID в таких summary-LSA содержит значение DefaultDestination, а метрика задается для каждой области конфигурационным параметром StubDefaultCost. Отметим, что не требуется задавать одинаковые значения StubDefaultCost на всех граничных маршрутизаторах тупиковой области.

### 12.4.3.2. Примеры summary-LSA

Вернемся к конфигурации областей на рисунке 4. Маршрутизаторы RT3, RT4, RT7, RT10 и RT11 являются граничными для областей и, следовательно, порождают summary-LSA. Рассмотрим в частности маршрутизатор RT4. таблица маршрутов для него рассчитана в качестве примера в параграфе 11.3. RT4 генерирует summary-LSA в магистраль и область 1. В магистраль RT4 генерирует отдельные LSA для каждой из сетей N1 - N4, в область 1 – отдельные LSA для сетей N6 - N8 и граничных маршрутизаторов AS (RT5, RT7). Он также собирает маршруты к хостам Ia и Ib в один анонс summary-LSA. Наконец, маршруты в сети N9, N10, N11 и к хосту H1 анонсируются одним summary-LSA. Эта конденсация изначально выполняется маршрутизатором RT11.

Эти LSA показаны в таблицах 6 и 7. Ниже показаны два анонса summary-LSA, генерируемых маршрутизатором RT4. Реальные адреса для сетей и маршрутизаторов показаны на рисунке 11.

```
; Summary-LSA для сети N1 генерируется маршрутизатором RT4 в магистраль
LS age = 0 ; всегда верно при создании
Options = (E-bit) ;
LS type = 3 ; тип 3 - summary-LSA
Link State ID = 192.1.2.0 ; IP-номер сети N1
Advertising Router = 192.1.1.4 ; Идентификатор RT4
metric = 4

; Summary-LSA для граничного маршрутизатора AS RT7
; генерируется маршрутизатором RT4 в область 1
LS age = 0 ; всегда верно при создании
Options = (E-bit) ;
LS type = 4 ; тип 4 - summary-LSA
Link State ID = Router ID RT7
Advertising Router = 192.1.1.4 ; Идентификатор RT4
metric = 14
```

### 12.4.4. AS-external-LSA

Анонсы AS-external-LSA описывают маршруты ко внешним (по отношению к AS) адресатам. Большинство AS-external-LSA описывает маршруты к конкретным внешним адресатам – в таких случаях поле Link State ID содержит IP-номер сети получателя address (при необходимости можно включать и часть битов «номера хоста», как описано в Приложении E). Для описания принятого по умолчанию маршрута для AS может использоваться AS-external-LSA с Link State ID = DefaultDestination (0.0.0.0). Анонсы AS-external-LSA генерируются граничными маршрутизаторами AS, которые порождают по одному анонсу AS-external-LSA для каждого внешнего маршрута, полученного от другого протокола (типа BGP) или заданного конфигурацией.

Анонсы AS-external-LSA являются единственным типом LSA, для которого лавинная рассылка производится во всей AS; остальные типы LSA связаны с отдельными областями. Однако AS-external-LSA не рассылаются в тупиковые области и через такие области (см. параграф 3.6) для снижения размеров баз данных в маршрутизаторах тупиковых областей.

Для внешних маршрутов может использоваться два типа метрики. Метрика типа 1 сравнима с внутренней метрикой link state, а метрика типа 2 заведомо больше стоимости любого пути внутри AS.

Если маршрутизатор анонсирует AS-external-LSA для адресата, ставшего недоступным, он должен удалить LSA из домена маршрутизации путем установки для них возраста MaxAge и повтора лавинной рассылки (см. параграф 14.1).

#### 12.4.4.1. Примеры AS-external-LSA

В приведенном на рисунке 4 примере AS имеются два граничных маршрутизатора AS - RT5 и RT7. Маршрутизатор RT5 генерирует три AS-external-LSA для сетей N12 - N14, а RT7 – два анонса для сетей N12 и N15. Предположим, что маршрутизатор RT7 узнал о пути в сеть N12 от протокола BGP и хочет анонсировать в AS метрику типа. Тогда RT7 будет генерировать для сети N12 анонс:

```
; AS-external-LSA для сети N12, генерируется маршрутизатором RT7
LS age = 0 ; всегда верно при создании
Options = (E-bit) ;
LS type = 5 ; AS-external-LSA
Link State ID = IP-номер для сети N12
Advertising Router = Router ID RT7
bit E = 1 ; метрика типа 2
metric = 2
Forwarding address = 0.0.0.0
```

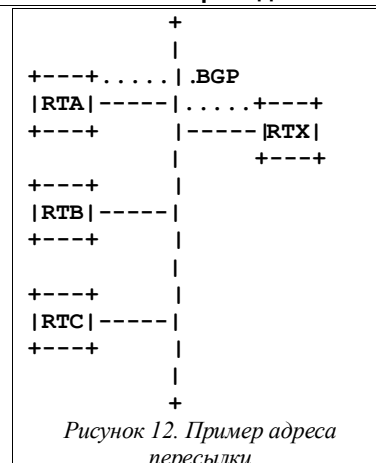
Значение 0.0.0.0 в качестве адреса пересылки показывает, что пакеты для внешних адресатов должны пересылаться анонсирующему маршрутизатору OSPF (RT7). В некоторых случаях такое решение нежелательно. Рассмотрим пример на рисунке 12, содержащий три маршрутизатора OSPF (RTA, RTB and RTC), подключенных к общей сети. Только маршрутизатор RTA обменивается информацией BGP с маршрутизатором RTX, не относящимся к OSPF. В этом случае RTA должен порождать AS-external-LSA для адресатов, полученных от RTX. Используя поле адреса пересылки в AS-external-LSA, RTA может задать пересылку пакетов в эти адреса непосредственно через RTX. Если это не использовать, маршрутизаторы RTB и RTC будут добавлять лишний интервал.

Отметим, что ненулевой адрес пересылки должен указывать на маршрутизатор в другой AS. Это поле может также содержать принятый по умолчанию адрес. Например, маршрутизатор RTA может указать, что все внешние пакеты должны по умолчанию пересылаться маршрутизатору RTX (BGP peer). Анонс AS-external-LSA для такого случая показан ниже. Отметим, что Link State ID = DefaultDestination.

```
;принятый по умолчанию маршрут от RTA
;пакеты пересылаются через RTX
LS age = 0 ;всегда верно при создании
Options = (E-bit) ;
LS type = 5 ;AS-external-LSA
Link State ID = DefaultDestination маршрут по умолчанию
Advertising Router = Router ID для RTA
bit E = 1 ;метрика типа 2
metric = 1
Forwarding address = IP-адрес RTX
```

Предположим, что маршрутизаторы RTA и RTB также обмениваются BGP-информацией с RTX. В таком случае RTA и RTB будут генерировать одинаковые анонсы AS-external-LSA.

Если в этих анонсах будет совпадать метрика, они будут эквивалентны функционально, поскольку указывают одинаковых получателей и общий адрес пересылки (RTX). Это ведет к простому дублированию работы. Если только один из маршрутизаторов (RTA или RTB) будет генерировать набор AS-external-LSA, маршрутизация не изменится, а размер базы данных уменьшится. Однако в таких случаях нужно точно указывать, какой из маршрутизаторов будет порождать LSA (иначе этого может не сделать ни один из них или начнется поочередная генерация). Для выбора маршрутизатора используется следующее правило – если два доступных один для другого маршрутизатора генерируют функционально эквивалентные AS-external-LSA (совпадают адресаты, стоимость и ненулевой адрес пересылки), используются LSA от маршрутизатора с большим значением OSPF Router ID. Другой маршрутизатор должен удалить свои LSA, как описано в параграфе 14.1.



### 13. Процедура лавинной рассылки (Flooding)

Пакет Link State Update (LSU) обеспечивают механизм лавинной рассылки LSA и могут содержать несколько различных LSA. Пакеты рассылаются лавинным способом на один интервал от точки генерации LSA. Для повышения надежности лавинной рассылки требуется раздельное подтверждение для каждого анонса LSA, передаваемое в пакетах Link State Acknowledgment. В одном пакете может содержаться множество раздельных подтверждений.

Процедура лавинной рассылки начинается при получении пакета LSU. До передачи пакета процедуре лавинной рассылки выполняется целый ряд проверок (см. параграф 8.2). В частности, пакет LSU ассоциируется с конкретным соседом и конкретной областью. Если сосед находится в состоянии ниже Exchange, пакет должен быть отброшен без дальнейшей обработки.

Все типы LSA (кроме AS-external-LSA) связываются с конкретной областью, однако LSA не содержат поля области и область определяется из заголовка пакета LSU.

Для каждого анонса LSA в пакете LSU выполняются следующие операции:

- (1) Проверка контрольной суммы LSA. В случае ошибки LSA отбрасывается с переходом к следующему анонсу в пакете LSU.
- (2) Проверка типа LSA. LSA неизвестных типов отбрасываются с переходом к следующему анонсу из пакета LSU. Данная спецификация определяет типы 1 - 5 (см. параграф 4.3).
- (3) Если принят анонс AS-external-LSA (тип 5), а область является тупиковой, LSA отбрасывается с переходом к следующему анонсу из пакета LSU. Для AS-external-LSA не используется лавинная рассылка в тупиковые области или через них (см. 3.6).
- (4) Если возраст LSA равен MaxAge, а в базе данных маршрутизатора нет текущего экземпляра LSA и ни один из соседних маршрутизаторов не находится в состоянии Exchange или Loading, выполняются следующие действия:
  - (a) Подтверждается прием LSA путем передачи пакета Link State Acknowledgment отправителю LSA (см. параграф 13.5);
  - (b) LSA отбрасывается с переходом к следующему анонсу (если он имеется) из пакета LSU.
- (5) В остальных случаях отыскивается текущий экземпляр LSA в базе данных маршрутизатора. Если в базе не найдено копии или LSA новее копии в базе данных (сравнение возраста LSA описано в параграфе 13.1), выполняются следующие операции:
  - (a) Если в базе данных уже есть копия, полученная в результате лавинной рассылки и установленная менее MinLSArrival секунд назад, новый анонс отбрасывается (без подтверждения) и проверяется следующий анонс (если он есть) в пакете LSU.
  - (b) В противном случае немедленно начинается лавинная рассылка нового анонса LSA через часть интерфейсов маршрутизатора (см. параграф 13.3). В некоторых случаях (например, если принимающий интерфейс находится в состоянии DR и анонс LSA был принят не от Backup DR) LSA рассылаются в лавинном режиме принявшим анонс интерфейсом. Эти случаи должны фиксироваться для последующего использования в процессе подтверждения (параграф 13.5).
  - (c) Текущая копия из базы данных удаляется из списков Link state retransmission всех соседей.
  - (d) Новая запись LSA помещается в базу данных взамен текущей копии. Эта замена может приводить к планируемому пересчету таблицы маршрутов. Для новой записи LSA фиксируется время ее поступления. Процедура лавинной рассылки не может переписывать новые LSA, пока не пройдет MinLSArrival секунд. Установка LSA описана в параграфе 13.2.
  - (e) По возможности подтверждается прием LSA путем передачи пакета Link State Acknowledgment через принявший анонс интерфейс (см. параграф 13.5).
  - (f) Если новый анонс сгенерирован принявшим его маршрутизатором (self-originated LSA), этот маршрутизатор должен выполнить специальные действия – обновить LSA или (в некоторых случаях) удалить анонс из домена маршрутизации. Обнаружение и обработка self-originated LSA описаны в параграфе 13.4.
- (6) Далее, если экземпляр LSA находится в списке Link state request передающего соседа, в процессе Database Exchange происходит ошибка. В таких случаях процесс Database Exchange перезапускается путем генерации для соседа и передачи ему события BadLSReq и прекращения обработки пакетов LSU.
- (7) Иначе, если LSA совпадает с экземпляром в базе данных (т. е., их возраст одинаков), выполняются две операции:
  - (a) Если LSA присутствует в списке Link state retransmission для принимающего смежного маршрутизатора, предполагается, что маршрутизатор самостоятельно подтверждает LSA. Принятые LSA следует трактовать как подтверждения, удаляя их из списка Link state retransmission (это называется «подразумеваемым подтверждением - implied acknowledgment»). Такие события должны документироваться для последующего использования в процессе подтверждения (параграф 13.5).
  - (b) Можно подтвердить прием LSA путем передачи пакета Link State Acknowledgment через принявший интерфейс (см. пример в параграфе 13.5).

(8) В остальных случаях копия в базе данных является более новой. Если возраст этой копии равен MaxAge и порядковый номер равен MaxSequenceNumber, принятый анонс LSA просто отбрасывается без подтверждения (это говорит о завершении цикла счетчика порядковых номеров и необходимости удаления LSA с номером MaxSequenceNumber до генерации новых экземпляров LSA). В противном случае, если копия из базы данных не была передана в пакете LSU в течение последних MinLSArrival секунд, эта копия должна быть отправлена передающему соседу в пакете LSU (пакет должен быть адресован соседу напрямую). В таких случаях копия LSA из базы данных не помещается в список link state retransmission для соседа и прием последнего анонса LSA не подтверждается.

### 13.1. Сравнение возраста LSA

Когда маршрутизатор встречает два экземпляра LSA, он должен определить более новый. Это выполняется путем сравнения принятой записи LSA с ее копией в базе данных. Сравнение также должно проводиться во время процедуры Database Exchange при организации отношений смежности.

Записи LSA идентифицируются полями LS type, Link State ID и Advertising Router. Для двух экземпляров LSA поля LS sequence number, LS age и LS checksum позволяют выбрать более новый экземпляр:

- ◆ Запись LSA с большим порядковым номером является более новой (описание нумерации приведено в параграфе 12.1.6). Если порядковые номера совпадают, выполняется следующая проверка
- ◆ Если экземпляры отличаются по контрольной сумме (16-битовое беззнаковое целое), более новым считается экземпляр с большим значением контрольной суммы.
- ◆ Если контрольные суммы совпадают и возраст одного из экземпляров равен MaxAge, этот экземпляр считается более новым.
- ◆ В противном случае, если возраст отличается более, чем на MaxAgeDiff, более новым считается экземпляр с меньшим возрастом.
- ◆ В остальных случаях экземпляры считаются идентичными.

### 13.2. Инсталляция LSA в базу данных

Включение в базу данных новых LSA в результате лавинной рассылки или самогенерации (self-originated LSA) может вызывать необходимость перерасчета таблицы маршрутов OSPF. Содержимое нового анонса LSA должно сравниваться со старым экземпляром, если такой имеется. При отсутствии различий не требуется пересчитывать таблицу маршрутов. Сравнение LSA с предыдущим экземпляром осуществляется по следующим критериям:

- ◆ Изменение поля Options в LSA.
- ◆ В одной из записей LSA возраст равен MaxAge, а в другой меньше.
- ◆ Изменилось поле длины в заголовке LSA.
- ◆ Изменилось тело LSA (содержимое LSA после 20-байтового заголовка). Отметим, что изменения LS Sequence Number и LS Checksum относятся к заголовку и не попадают в данную категорию.

При обнаружении различий должны пересчитываться следующие компоненты таблицы маршрутов (в зависимости от типа LSA):

#### **Router-LSA и network-LSA**

Требуется полное обновление таблицы маршрутов, начиная с расчета дерева кратчайших путей для каждой области (не только для той, где изменилась база данных link-state). Причина пересчета деревьев для каждой области заключается в том, что граничные маршрутизаторы AS могут входить в несколько областей. Изменения в области, обеспечивающей лучший на данный момент маршрут, могут вынудить маршрутизатор использовать внутридоменный маршрут из другой области<sup>26</sup>.

#### **Summary-LSA**

Должен пересчитываться лучший маршрут к адресату, описанный в summary-LSA (см. параграф 16.5). Если адресатом является граничный маршрутизатор AS, может также потребоваться повторная проверка всех AS-external-LSA.

#### **AS-external-LSA**

Должен пересчитываться лучший маршрут к адресату, описанный в AS-external-LSA (см. параграф 16.6).

Кроме того, все старые экземпляры LSA должны быть удалены из базы данных при установке новой записи LSA. Старые экземпляры должны также удаляться из списков Link state retransmission всех соседей (см. главу 10).

### 13.3. Следующий этап лавинной рассылки

При получении нового (более свежего) анонса LSA он должен рассылаться в лавинном режиме некоторыми интерфейсами маршрутизатора. В этом параграфе описана вторая часть процесса лавинной рассылки (первой частью является описанная выше в данной главе обработка), а именно – выбор интерфейсов для рассылки и добавление LSA в списки Link state retransmission подходящих соседей. Эта часть рассылки включает также поддержку списков LSR от соседей.

Этот параграф применим и к лавинной рассылке LSA, порожденных самим маршрутизатором (см. параграф 12.4). Для таких LSA этот параграф включает полное описание процесса лавинной рассылки (т. е., описанные выше в главе 13 операции не проводятся для LSA, генерируемых самим маршрутизатором).

В зависимости от типа LSA рассылка осуществляется через один или несколько интерфейсов, для определения которых используются следующие правила:

#### **AS-external-LSA (LS Type = 5)**

Анонсы AS-external-LSA рассылаются в лавинном режиме по всей AS, исключая лишь тупиковые области (см. параграф 3.6).

Для рассылки используются все интерфейсы маршрутизатора, кроме виртуальных каналов и интерфейсов, подключенных к тупиковым областям.

#### **Прочие типы**

Все прочие анонсы связаны с одной областью (далее, область A). Для рассылки анонсов используются все интерфейсы маршрутизатора в данную область. Если область является магистральной, используются также виртуальные каналы.

Должна обеспечиваться синхронизация баз данных для всех смежных пар, связанных с используемыми для рассылки интерфейсами. Это обеспечивается путем выполнения для каждого из этих интерфейсов перечисленных ниже операций. Следует отметить, что в результате описанной процедуры может быть принято решение не рассылать LSA через тот или иной интерфейс, если существует высокая вероятность того, что подключенные соседи уже получили LSA. Однако, в таких случаях должна быть абсолютная гарантия получения соседями анонсов LSA, поэтому LSA нужно включать в списки Link state retransmission каждого смежного маршрутизатора. Для всех используемых в рассылке интерфейсов нужно выполнить следующее:

(1) Проверяется каждый из соседей, подключенных к интерфейсу на предмет необходимости получения LSA:

(а) Если сосед еще не достиг состояния Exchange, он не участвует в рассылке.

<sup>26</sup> За счет хранения в таблице маршрутов дополнительной информации некоторые реализации могут ограничиться пересчетом дерева кратчайших путей для одной области. Существует алгоритм нарастающих изменений (incremental algorithm), который позволяет пересчитывать только часть дерева кратчайших путей одной области [Ref1]. Однако рассмотрение этого алгоритма выходит за пределы данной спецификации.

- (b) Если отношения смежности еще не полные (состояние соседа Exchange или Loading), проверяется список запросов состояния канала, связанный со смежностью. Если в списке присутствует экземпляр нового анонса LSA, это говорит о том, что соседний маршрутизатор уже имеет экземпляр LSA и нужно сравнить его с новым:
    - Если новый анонс LSA старше имеющегося, переход к следующему соседу.
    - Если это две копии одного экземпляра, нужно удалить LSA из списка запросов состояния канала и перейти к следующему соседу<sup>27</sup>.
    - В остальных случаях новая запись LSA является более свежей. Удалите LSA из списка Link state request.
  - (c) Если новый анонс LSA был получен от этого соседа, проверяется следующий сосед.
  - (d) Поскольку нельзя сказать определенно, что сосед имеет свежий экземпляр LSA, нужно добавить новый анонс в список Link state retransmission для смежного маршрутизатора. Это обеспечивает надежность процедуры лавинной рассылки - LSA передаются повторно с заданным интервалом, пока от соседа не придет подтверждение.
- (2) Маршрутизатор должен решить вопрос о необходимости лавинной рассылки LSA через данный интерфейс. Если на этапе 1 анонс LSA **не был** добавлен ни в один из списков Link state retransmission, нет необходимости рассылать LSA через данный интерфейс и можно переходить к следующему интерфейсу.
- (3) Если новый анонс LSA был принят через данный интерфейс от маршрутизатора DR или Backup DR, существует вероятность, что все соседи уже получили копию LSA и можно переходить к следующему интерфейсу.
- (4) Если новый анонс LSA получен через этот интерфейс и последний находится в состоянии Backup (маршрутизатор служит Backup DR), проверяется следующий интерфейс. Маршрутизатор DR будет слать анонсы на этот интерфейс, однако при «падении» DR данный маршрутизатор (т. е., Backup DR) будет завершать повторную рассылку анонсов.
- (5) Если проверка дошла до этого этапа, требуется лавинная рассылка LSA через данный интерфейс. Следует передать пакет LSU, содержащий новый анонс LSA в своем теле. Возраст LSA должен быть увеличен на InfTransDelay при копировании анонса в пакет LSU (если возраст не равен MaxAge).
- В широковещательных сетях пакеты LSU рассылаются по групповым адресам. Указываемый в пакете LSU IP-адрес получателя зависит от состояния интерфейса. Если интерфейс находится в состоянии DR или Backup, используется адрес AllSPFRouters, в остальных случаях - AllDRouters.
- В сетях без широковещания передается отдельная копия пакета LSU каждому смежному соседу (состояние не ниже Exchange). В качестве получателя пакета указывается IP-адрес соседа.

### 13.4. Прием собственных (self-originated) LSA

Получение своих (self-originated) анонсов LSA в результате лавинной рассылки является нормальным событием. Для детектирования свих self-originated LSA маршрутизатор может использовать 2 признака: 1) поле Advertising Router в LSA совпадает с Router ID или 2) поле Link State ID в network-LSA совпадает с IP-адресом интерфейса маршрутизатора.

Если принятый экземпляр self-originated LSA оказывается новее последнего экземпляра, реально порожденного маршрутизатором, требуются специальные меры. Получение таких LSA показывает наличие в области маршрутизации LSA, порожденных маршрутизатором до его последней перезагрузки. В большинстве случаев маршрутизатору следует установить порядковый номер LSA на 1 больше номера принятого анонса и сгенерировать новый экземпляр LSA.

Возможны ситуации, когда маршрутизатор не захочет генерировать принятые LSA:

- 1) Анонс является summary-LSA или AS-external-LSA, а маршрутизатор больше не имеет анонсируемого пути к получателю;
- 2) Анонс является network-LSA, а маршрутизатор больше не является DR для сети;
- 3) Анонс является network-LSA и поле Link State ID содержит IP-адрес одного из интерфейсов маршрутизатора, но поле Advertising Router не совпадает с Router ID данного маршрутизатора (этот редкий случай говорит об изменении Router ID после генерации LSA).

Во всех этих случаях вместо обновления LSA, требуется их удаление из области маршрутизации путем установки LS age = MaxAge и повтора лавинной рассылки (см. параграф 14.1).

### 13.5. Передача LSA

Прием каждого нового анонса LSA должен подтверждаться. Обычно для подтверждения используются пакеты Link State Acknowledgment (LSAck), однако возможно и неявное подтверждение с помощью пакетов LSU (см. п. 7а в параграфе 13).

Можно группировать множество подтверждений в один пакет LSAck, передавая этот пакет обратно через принявший анонсы интерфейс. Пакет подтверждения можно передавать 2 способами – с задержкой или непосредственно. Использование той или иной стратегии зависит от обстоятельств приема LSA.

Передача подтверждений с задержкой предполагает несколько условий:

- 1) Компоновка множества подтверждений в один пакет LSAck.
- 2) Возможность использования одного пакета LSAck для индикации подтверждений нескольким соседям одновременно (групповая адресация).
- 3) Произвольная (случайная) передача пакетов LSAck, передаваемых разным маршрутизаторам, подключенным к общей сети. Фиксированный интервал между отложенными передачами должен быть меньше RxmtInterval, иначе повтор станет ненужным. Первые подтверждения передаются непосредственно соседу в ответ на получение дубликата LSA сразу же после приема дубликата. В сетях с множественным доступом такие подтверждения могут посылаются напрямую по IP-адресу соседа.

Точные процедуры передачи пакетов LSAck описаны в таблице 22. Обстоятельства приема LSA показаны в левой колонке, а выполняемые действия для различных состояний в остальных колонках. Действия зависят от состояния интерфейса – реакция для состояния Backup отличается от действий в других состояниях. Отложенные подтверждения должны доставляться всем смежным маршрутизаторам, связанным с интерфейсом. В широковещательных сетях это осуществляется путем передачи пакетов LSAck с использованием групповых адресов. Используемое значение Destination IP зависит от состояния интерфейса – для состояний DR и Backup используется групповой адрес AllSPFRouters, а для прочих состояний - AllDRouters. В сетях без широковещания отложенные пакеты LSAck должны передаваться отдельно каждому смежному маршрутизатору (т. е., соседу в состоянии >= Exchange).

Таблица 22 Передача анонсов состояния каналов.

Обстоятельства	Действия, выполняемые в состоянии	
	Backup	Остальные состояния
Приемный интерфейс будет рассылать LSA обратно, используя flooding (см. главу 13, шаг 5b).	Подтверждение не передается.	Подтверждение не передается.
LSA новее копии в базе данных, но не рассылается обратно принявшим ее интерфейсом.	Если анонс получен от DR, передается отложенное подтверждение, в противном случае – ничего.	Передается отложенное подтверждение.

<sup>27</sup> После опустошения списка Link state request сосед так или иначе переходит в состояние Full (см. параграф 10.9).



LSA является дубликатом и трактовалась как косвенное подтверждение (см. главу 13, шаг 7а).	Если анонс получен от DR, передается отложенное подтверждение, в противном случае – ничего.	Подтверждение не передается.
LSA является дубликатом и трактовалась как косвенное подтверждение.	Передается прямое подтверждение.	Передается прямое подтверждение.
Возраст LSA MaxAge, отсутствует текущий экземпляр LSA в базе данных и ни один из соседей маршрутизатора не находится в состоянии Exchange или Loading (см. главу 13, шаг 4).	Передается прямое подтверждение.	Передается прямое подтверждение.

Причины использования групповой адресации поясним на примере. Вернемся к сети, показанной на рисунке 11, и предположим, что маршрутизатор RT4 играет роль DR, а RT3 - Backup DR для сети N3. Когда RT4 рассылает новый анонс LSA в сеть N3, его принимают маршрутизаторы RT1, RT2 и RT3, которые не будут посылать LSA обратно в сеть N3, но по-прежнему должны гарантировать синхронизацию своих баз данных со смежными соседями. В результате маршрутизаторы RT1, RT2 и RT4 ожидают подтверждений от RT3, а маршрутизаторы RT4 и RT3 ждут подтверждений от RT1 RT2. В такой ситуации удобны групповые адреса.

Логика подтверждений для Backup DR несколько отличается за счет различий в процедуре лавинной рассылки LSA (см. параграф 13.3, п. 4).

## 13.6. Повторная передача LSA

Анонсы LSA, рассылаемые в лавинном режиме смежным маршрутизатором, будут помещаться в список Link state retransmission для него. Для обеспечения надежности лавинной рассылки повтор передачи LSA длится до приема подтверждения. Интервал между повторами задается для каждого интерфейса параметром RxmtInterval. Если установленное для интерфейса значение слишком велико, повтор передачи будет бесполезен. При слишком малом периоде повторов это может повлиять на скорость лавинной рассылки.

Несколько передаваемых повторно LSA можно поместить в один пакет LSU. При повторной передаче LSA должно посылаться только то число анонсов, которое можно поместить в один пакет LSU. Другие пакеты с повторами могут быть переданы после подтверждения некоторых LSA или при следующем завершении периода повтора.

Пакеты LSU, содержащие повторы анонсов, всегда передаются напрямую соседу. В сетях с множественным доступом это означает передачу повторов непосредственно по IP-адресу соседа. Возраст каждого анонса LSA должен увеличиваться на InfTransDelay (> 0) при копировании в пакет LSU (если возраст не достигает MaxAge).

Если смежный маршрутизатор не работает, повтор может происходить до тех пор, пока смежность не будет устранена с помощью протокола Hello. При разрушении смежности список Link state retransmission очищается.

## 13.7. Подтверждение приема состояния канала

Прежде, чем передавать полученные пакеты LSAck на обработку процедуре лавинной рассылки, выполняется множество проверок на непротиворечивость. В частности пакеты ассоциируются с определенным соседом. Если состояние этого соседа ниже Exchange, пакеты LSAck должны отбрасываться. В остальных случаях для каждого подтверждения в пакете LSAck выполняются следующие операции:

- ♦ Если не подтверждается экземпляр LSA из списка Link state retransmission для соседа, переход к следующему подтверждению
- ♦ Если подтверждается тот же самый экземпляр, последний удаляется из списка с переходом к следующему подтверждению.
- ♦ Протоколируется спорное подтверждение с переходом к следующему подтверждению.

## 14. Старение базы данных LS

Каждая запись LSA имеет поле возраста LS, выраженного в секундах. Возраст LSA увеличивается при нахождении записи в базе данных; при копировании записи в пакет LSU для рассылки через какой-либо из интерфейсов возраст также увеличивается на InfTransDelay.

Возраст LSA никогда не превышает значения MaxAge и LSA, достигшие возраста MaxAge, не используются при расчетах таблиц маршрутов. При увеличении возраста базы данных поле LS age в LSA может достигнуть значения MaxAge<sup>28</sup>. В таких случаях маршрутизатор должен предпринять попытку удаления LSA из области маршрутизации путем повторной лавинной рассылки LSA с возрастом MaxAge как свежесгенерированного анонса LSA (см. параграф 13.3).

При создании списка Database summary для формируемых отношений смежности все LSA с возрастом MaxAge в базе данных добавляются в список Link state retransmission для соседа вместо списка Database summary для этого соседа (см. параграф 10.3).

LSA с возрастом MaxAge должны незамедлительно удаляться из базы данных маршрутизатора, если а) они не содержатся более ни в одном из списков Link state retransmission для соседей и б) ни один из соседей маршрутизатора не находится в состоянии Exchange или Loading.

Когда в процессе увеличения возраста базы данных поле LS age требует многократного обращения к CheckAge, должна проверяться контрольная сумма LSA. Если контрольная сумма некорректна, это говорит о наличии ошибок в программе или памяти – в таких случаях рекомендуется перезагрузить маршрутизатор.

### 14.1. Принудительное старение LSA

Анонсы LSA могут удаляться из домена маршрутизации путем установки LS age = MaxAge с сохранением порядкового номера и повторной лавинной рассылки LSA. Эта процедура совпадает с процедурой, применяемой для LSA, возраст которых достиг MaxAge естественным путем (см. параграф 14). В частности LSA удаляются из базы данных маршрутизатора, если а) эти анонсы больше не включены в списки Link state retransmission ни одного из соседей и б) ни один из соседних маршрутизаторов не находится в состоянии Exchange или Loading. Будем называть процесс установки для LSA возраста MaxAge принудительным старением (premature aging).

Принудительное старение используется при достижении максимального порядкового номера для собственных (self-originated) LSA. В таких случаях текущий экземпляр LSA (порядковый номер MaxSequenceNumber) должен быть принудительно состарен и

<sup>28</sup> Анонсы LSA сравнительно редко достигают возраста MaxAge; обычно они заменяются новыми экземплярами раньше.

удален из домена маршрутизации до того, как будет порожден новый экземпляр с порядковым номером InitialSequenceNumber (см. параграф 12.1.6).

Принудительное старение может также использоваться в тех случаях, когда какой-либо из анонсированных маршрутизатором внешних путей перестал быть доступным. В таких случаях маршрутизатор может удалить свои AS-external-LSA из домена маршрутизации с помощью принудительного старения. Эта процедура является более предпочтительной, нежели генерация для недоступного маршрута новых LSA с метрикой LSInfinity. Принудительное старение также используется при неожиданном получении собственных LSA в процессе лавинной рассылки (см. параграф 13.4).

Маршрутизатор может использовать принудительное старение только для собственных LSA. Анонсы LSA считаются собственными (self-originated), если 1) поле Advertising Router в LSA содержит значение Router ID самого маршрутизатора или 2) поле Link State ID в network-LSA содержит IP-адрес одного из интерфейсов маршрутизатора.

## 15. Виртуальные соединения

Каждая область автономной системы должна иметь соединение с магистралью (Area ID = 0.0.0.0), иначе некоторые области AS перестанут быть доступными. Для организации и поддержки соединений с магистральной областью могут использоваться виртуальные каналы через области, не входящие в магистраль AS. Виртуальные каналы служат для подключения физически отделенных компонент к магистрали. Конечными точками виртуального канала являются граничные маршрутизаторы областей. Виртуальное соединение должно быть настроено на обоих маршрутизаторах. Конфигурационные параметры каждого маршрутизатора содержат сведения о другой стороне виртуального канала (граничный маршрутизатор области) и немагистральной области, являющейся общей для обеих конечных точек (транзитная область). Виртуальные каналы не могут проходить через тупиковую область (см. параграф 3.6).

Виртуальное соединение трактуется как сеть point-to-point, относящаяся к магистрали и соединяющая пару граничных маршрутизаторов областей. Через виртуальный канал маршрутизаторы пытаются организовать отношения смежности. После организации смежности виртуальный канал включается в магистральные анонсы router-LSA и пакеты OSPF, принадлежащие магистральной области, будут передаваться через виртуальный канал с отношениями смежности (виртуальная смежность или virtual adjacency).

В каждой из конечных точек доступность и стоимость виртуального канала определяется путем просмотра в таблице маршрута для другой точки этого соединения (связанная с маршрутом область должна быть задана как транзитная). Когда связанный с виртуальным соединением путь становится доступным, генерируется событие InterfaceUp. И наоборот, при невозможности использования пути через виртуальный канал генерируется событие InterfaceDown. Иными словами, доступность виртуального канала определяется существованием внутридоменного маршрута между конечными точками через транзитную область. Отметим, что виртуальные каналы, для которых стоимость превышает значение 0xffff (максимальная стоимость интерфейса в router-LSA), должны рассматриваться как неработоспособные (т. е., путь между конечными точками виртуального канала не существует).

Некоторые особенности виртуальных каналов перечислены ниже:

- ♦ Анонсы AS-external-LSA **никогда** не рассылаются через виртуальные каналы со смежностью, поскольку такая рассылка ведет к дублированию (те же AS-external-LSA уже разосланы через транзитную область виртуального соединения). По этой же причине AS-external-LSA не резюмируются через виртуальные каналы со смежностью в процессах Database Exchange.
- ♦ Стоимость виртуального соединения **не задается** конфигурационными параметрами, а определяется как стоимость внутридоменного маршрута между двумя определенными граничными маршрутизаторами областей. Эта стоимость появляется в соответствующей виртуальному соединению записи таблицы маршрутов. При изменении стоимости виртуального маршрута должен генерироваться новый анонс router-LSA для магистральной области.
- ♦ Подобно тому, как стоимость и доступность виртуального соединения определяются в процессе построения таблицы маршрутов (создание в таблице записи для соседа по виртуальному каналу), для виртуального канала существуют IP-адреса локального и удаленного интерфейсов. Эти адреса используются при передаче пакетов OSPF через виртуальный канал. Отметим, когда одна или обе конечные точки виртуального соединения подключены к транзитной области через безадресное соединение point-to-point, определение IP-адреса (своего или виртуального соседа) может оказаться невозможным, что приведет к «падению» виртуального канала.
- ♦ В анонсах router-LSA, генерируемых конечными точками для магистральной области, виртуальный канал представляется типом 4 - поле Link ID содержит значение OSPF Router ID соседа, а Link Data содержит IP-адрес виртуального интерфейса (см. параграф 12.4.1).
- ♦ Немагистральная область может передавать транзитный трафик данных (т. е., использоваться как транзитная область) в тех и только в тех случаях, когда эта область задана в качестве транзитной для одного или нескольких виртуальных каналов с полной смежностью (см. TransitCapability в параграфах 6 и 16.1). Такие области требуют специальной трактовки при резюмировании в них информации о магистральных сетях (см. параграф 12.4.3) и в процессе расчета маршрутов (см. параграф 16.3).
- ♦ Для виртуальных каналов задается время повтора передачи данных о состоянии каналов – RxmtInterval. Это время должно превышать ожидаемое значение времени кругового обхода (round-trip delay) между двумя маршрутизаторами. Оценка этого времени для виртуального канала может быть достаточно сложной и лучше будет переоценить задержку, нежели наоборот.

## 16. Расчет таблицы маршрутизации

В этой главе рассмотрен процесс расчета таблицы маршрутизации OSPF. Используя базу данных о каналах подключенной области, маршрутизатор использует описанный ниже алгоритм для построения таблицы маршрутов. На каждом этапе построения маршрутизатор должен обращаться к отдельным частям базы данных (например, к анонсам router-LSA одного из маршрутизаторов). Эти обращения используют функции просмотра базы данных, описанные в параграфе 12.2. Функция поиска может возвращать LSA с возрастом MaxAge. Такие не должны использоваться при расчете маршрутов и результат поиска трактуется как отрицательный.

Организация таблицы маршрутов OSPF описана в главе 11, где в качестве примера были построены таблицы для двух разных случаев (параграфы 11.2 и 11.3). Процесс построения таблицы можно разбить на несколько этапов:

- (1) Существующая таблица маршрутов объявляется некорректной и начинается построение новой таблицы. Старая таблица сохраняется для идентификации изменений.
- (2) Внутридоменные маршруты рассчитываются путем построения дерева кратчайших путей для области. В частности, на этом этапе рассчитываются все записи таблицы маршрутов, для которых адресатами являются граничные маршрутизаторы области. Процесс выполняется в два приема – сначала строится дерево, содержащее только каналы между маршрутизаторами и транзитными сетями, а потом к дереву добавляются транзитные сети. При построении дерева кратчайших путей рассчитывается также значение TransitCapability, используемое на этапе 4.

- (3) Рассчитываются междоменные маршруты путем просмотра анонсов summary-LSA. Если маршрутизатор подключен к нескольким областям (т. е., является граничным для области), принимаются во внимание только summary-LSA от магистралей.
- (4) В граничных маршрутизаторах областей, подключенных по крайней мере к одной транзитной области (не относящаяся к магистральной) область с TransitCapability = TRUE), проверяются summary-LSA от транзитных областей для проверки существования через транзитную область более короткого пути по сравнению с маршрутами, найденными на этапах 2 - 3.
- (5) Рассчитываются внешние маршруты путем обработки AS-external-LSA. Местоположение граничных маршрутизаторов было определено на этапах 2-4.

Детальное описание этапов 2-5 приведено ниже.

Изменения, вносимые в таблицу маршрутов, могут потребовать от протокола OSPF дальнейших действий. Например, изменение внутрисетового маршрута от граничных маршрутизаторов области может потребоваться генерация новых анонсов summary-LSA (см. параграф 12.4). Полное описание действий OSPF в результате изменения таблицы маршрутов приведено в параграфе 16.7.

## 16.1. Расчет кратчайшего пути для области

Этот расчет дает набор внутрисетовых маршрутов, связанных с областью (назовем ее областью A). Маршрутизатор, рассчитывающий дерево, использует себя в качестве корня<sup>29</sup>. Дерево строится в два этапа. На первом этапе рассматриваются только соединения между маршрутизаторами и транзитными сетями и на базе алгоритма Dijkstra строится часть дерева, к которой на втором этапе добавляются каналы в тупиковые области.

Процедура построения дерева рассмотрена в главе 2. База данных области представляется в виде направленного графа, вершинами которого служат маршрутизаторы, транзитные и тупиковые сети. На первом этапе расчета тупиковые не рассматриваются. При расчете кратчайшего пути с каждой транзитной вершиной связываются следующие данные:

### Vertex (node) ID

32-битовый номер, который вместе с типом вершины (сеть или маршрутизатор) создает уникальное обозначение вершины. Для вершин маршрутизаторов поле Vertex ID содержит OSPF Router ID, для сетевых вершин – IP-адрес DR для этой сети.

### An LSA

Каждая транзитная вершина имеет связанную с ней запись LSA. Для вершин маршрутизаторов это router-LSA, для транзитных сетей - network-LSA (генерируется сетевым DR). В любом случае поле Link State ID в LSA равно Vertex ID.

### List of next hops

Список следующих маршрутизаторов для текущего набора кратчайших путей от корня к вершине. Поддержка equal-cost multipath capability позволяет существовать множеству равноценных путей. Каждое значение next hop показывает выходной интерфейс маршрутизатора для передачи пакетов адресату. В широкополосных, Point-to-MultiPoint и NBMA-сетях next hop включает также IP-адрес следующего маршрутизатора (если он есть) на пути передачи пакетов.

### Distance from root

Стоимость текущего набора кратчайших путей от корня до вершины. Стоимость пути рассчитывается как сумма стоимостей каналов на этом пути (в соответствии с router-LSA и network-LSA). Путь с меньшей стоимостью считается более коротким.

При каждой итерации алгоритма Dijkstra существует список вершин-кандидатов. Пути от корня до таких вершин существуют, но не обязательно являются кратчайшими. Однако пути к вершинам-кандидатам, расположенным наиболее близко к корню, гарантированно будут кратчайшими. Такие вершины добавляются в дерево кратчайших путей и удаляются из списка кандидатов, а соседние с ними вершины проверяются на предмет возможности добавления (обновления) в список кандидатов. После этого выполняется новая итерация алгоритма. Процесс завершается, когда список кандидатов становится пустым.

Ниже приведено детальное описание процедуры построения дерева. Напомним, что расчет ведется для области A и все обращения к базе данных относятся к этой области.

- (1) Инициализация структур данных алгоритма, очистка списка вершин-кандидатов. Инициализация дерева кратчайших путей к корню (маршрутизатор, рассчитывающий дерево). Установка для области A значения TransitCapability = FALSE
- (2) Проверка записи LSA, связанной с добавляемой вершиной (назовем ее V) – поиск в базе данных области A по значению Vertex ID. Если запись является router-LSA и бит V (см. параграф A.4.2) в ней установлен, задаем для области A TransitCapability = TRUE. В любом случае, каждый канал, описанный в LSA, дает стоимость смежной с ним вершины. Для каждого описанного канала (скажем, между вершинами V и W) выполняются следующие операции:
  - (a) Если канал ведет в тупиковую область, переходим к следующей вершине V в LSA, поскольку тупиковые области не рассматриваются на первом этапе расчета.
  - (b) В остальных случаях W является транзитной вершиной (маршрутизатор или транзитная сеть) и просматривается LSA (router-LSA или network-LSA) для этой вершины (W) в базе данных области A. Если LSA не существует, возраст записи равен MaxAge или не существует канала обратно к вершине V, переходим к следующему каналу LSA вершины V<sup>30</sup>.
  - (c) Если вершина W уже включена в дерево кратчайших путей, проверяется следующий канал в LSA.
  - (d) Расчет стоимости D результирующего пути от корня к вершине W. Значение D равно сумме стоимостей каналов (уже рассчитанной) для кратчайшего пути к вершине V и анонсируемой дистанции между вершинами V и W.
    - Если D превышает значение, уже полученное для вершины W в списке кандидатов, проверяется следующий канал.
    - Если D совпадает со стоимостью для вершины W в списке кандидатов, рассчитывается набор next hop, используемых с рассчитываемым каналом. Исходными данными для расчета являются получатель (W) и его родительская вершина (V). Расчет показан в параграфе 16.1.1. Набор интервалов должен добавляться к значениям next hop для вершины W в списке кандидатов.
    - Если D меньше стоимости для вершины W в списке кандидатов или W отсутствует в этом списке, организуется запись для вершины W в списке кандидатов, чтобы показать дистанцию D от корня. Рассчитывается также список next hop для использования с анонсируемым каналом и соответствующим образом устанавливаются значения next hop для вершины W. Расчеты next hop приведены в параграфе 16.1.1; в них используются адресат (W) и его родитель (V).
- (3) Если на этом этапе список кандидатов пуст, это говорит о завершении процесса построения дерева кратчайших путей для транзитных вершин. Если список содержит вершины-кандидаты, из него выбирается вершина, наиболее близкая к корню и добавляется в дерево кратчайших путей (из списка кандидатов эта вершина удаляется). Отметим, что при наличии нескольких вершин, одинаково удаленных от корня, следует выбирать сначала вершины, связанные с сетями (а не маршрутизаторами), чтобы можно было включить все равноценные пути. Это связано с использованием tie-breaker в модифицированном алгоритме Dijkstra, применяемом в MOSPF (OSPF Multicast routing extensions).

<sup>29</sup> Строго говоря, в результате поддержки равноценных путей результат расчета будет отличаться от дерева, но мы будем придерживаться термина «дерево» поскольку он повсеместно используется в литературе по данному вопросу.

<sup>30</sup> Отметим, что достаточно наличия любого канала обратно к вершине V и не требуется, чтобы он был «обратной копией» канала от V к W. Достаточно обеспечить гарантию передачи данных между соседними маршрутизаторами для синхронизации баз данных.



(4) На этом этапе возможно обновление таблицы маршрутов. Для измененных записей таблицы маршрутизации указывается связанная область A, внутридоменный тип маршрута и стоимость, определенная в процессе расчета дерева. Если недавно добавленная вершина является граничным маршрутизатором области или AS, в таблицу маршрутов добавляется запись с типом адресата router. Поле опций, найденной в связанной записи router-LSA, копируется в поле дополнительных возможностей записи в таблице маршрутов (Optional capabilities). Назовем новую вершину X. Если маршрутизатор X является конечной точкой одного из виртуальных каналов ведущего расчет маршрутизатора и этот канал использует область A как транзитную, декларируется активность виртуального канала, в качестве адреса виртуального интерфейса устанавливается IP-адрес выходного интерфейса, рассчитанного выше для Router X, а в качестве адреса соседа устанавливается IP-адрес интерфейса Router X (содержится в router-LSA от маршрутизатора X), который указывает обратно на корень дерева кратчайших путей (этот интерфейс также указывает на родительскую вершину для Router X в дереве кратчайших путей – подобный расчет приведен в параграфе 16.1.1).

Если добавленная вершина является транзитной сетью, в таблице маршрутов создается запись для сети. Поле Destination ID содержит IP-номер сети, получаемый маскированием Vertex ID (Link State ID) – маска сети содержится в теле network-LSA. Если такая запись уже существует в таблице маршрутов, на одну сеть отображается несколько вершин графа. Например, такая ситуация может наблюдаться при выборе нового маршрутизатора DR – в этом случае текущая запись таблицы маршрутов должна заменяться в тех и только в тех случаях, когда новый путь короче указанного в текущей записи маршрута и поле Link State Origin в текущей записи меньше значения Link State ID в LSA добавленной вершины.

Если для сети отсутствует запись в таблице (обычная ситуация), такая запись добавляется. Поле Link State Origin для новой записи должно содержать LSA добавляемой вершины.

(5) Новая итерация алгоритма, начиная с этапа 2.

После завершения расчета дерева для транзитных вершин в него добавляются тупиковые сети. На этом этапе заново проверяются вершины всех маршрутизаторов. Вершины, определенные на первом этапе как недоступные, отбрасываются. Для вершины каждого доступного маршрутизатора (назовем ее V) в базе данных отыскивается router-LSA. После этого просматриваются LSA для всех тупиковых сетей и выполняются следующие операции:

(1) Рассчитывается дистанция D тупиковой сети от корня, равная сумме дистанции от корня до вершины-маршрутизатора (вычисляется на этапе 1) и анонсируемой стоимости тупиковой сети. Полученное значение сравнивается с текущим значением лучшей стоимости тупиковой сети, путем просмотра текущей записи для тупиковой сети в таблице маршрутов. Если значение D превышает текущую стоимость, проверяется LSA для следующей тупиковой сети.

(2) При достижении этого этапа таблица маршрутизации тупиковой сети должна быть обновлена. Рассчитывается набор next hop в результате использования канала в тупиковую сеть (этот расчет показан в параграфе 16.1.1) с использованием адресата (тупиковая сеть) и родительской вершины (маршрутизатор). Если D совпадает со стоимостью в текущей записи таблицы маршрутов, набор next hop просто добавляется в список next hop записи в таблице. В таких случаях запись в таблице уже имеет поле Link State Origin. Если оно указывает на router-LSA с Link State ID < Router ID для маршрутизатора V, в качестве Link State Origin устанавливается router-LSA маршрутизатора V.

Если D меньше текущей стоимости, запись в таблице заменяется путем установки стоимости D и нового списка next hop. В поле Link State Origin записи в таблице маршрутов помещается router-LSA маршрутизатора V. После этого обрабатывается следующая тупиковая сеть.

Для всех записей таблицы добавляемых или изменяемых на втором этапе, устанавливается внутридоменный тип маршрута, связанный с областью A. Второй этап завершается, когда список доступных router-LSA становится пустым. К этому моменту все маршруты, связанные с областью A, будут определены.

Данная спецификация не требует использования описанного выше двухэтапного метода расчета дерева кратчайших путей. Однако при использовании другого алгоритма должно генерироваться идентичное дерево. По этой причине важно отметить, что каналы между транзитными вершинами должны быть двухсторонними для включения их в дерево. Отметим также, что для расчета дерева существуют более эффективные алгоритмы (например, алгоритм SPF с нарастающими изменениями, описанный в [Ref1]).

### 16.1.1. Расчет next hop

В этом параграфе описан способ расчета набора next hop (следующий маршрутизатор) для использования с адресатом. Каждый параметр next hop содержит интерфейс, используемый для пересылки пакетов адресату и IP-адрес следующего маршрутизатора (если он есть). Расчет next hop используется всякий раз при обнаружении кратчайшего пути к адресату, что может происходить на любом этапе построения дерева кратчайших путей (см. параграф 16.1). На первом этапе расчета дерева это происходит при добавлении найденного кратчайшего пути в список кандидатов или при изменении записи для кандидата (п. 2d этапа 1). На втором этапе кратчайший путь определяется всякий раз при изменении записи в таблице маршрутов (п. 2 этапа 2).

Набор next hop, используемый для адресата, может пересчитываться несколько раз в процессе построения дерева кратчайших путей. В конце расчетов запись таблицы маршрутов всегда будет содержать набор next hop для самого короткого пути (путей).

Исходными данными для расчета next hop является а) получатель и б) его родитель в текущем кратчайшем пути между корнем (рассчитывающий дерево маршрутизатор) и получателем. Родитель всегда является транзитной вершиной (маршрутизатор или транзитная сеть).

Если текущий кратчайший путь между адресатом и корнем включает по крайней мере один промежуточный маршрутизатор, адресат просто наследует набор next hop от родителя. В противном случае возможны два варианта. В первом случае родительской вершиной является корень (ведущий расчет маршрутизатор). Это означает, что адресатом могут служить непосредственно подключенные сеть или маршрутизатор. Выходным интерфейсом в таких ситуациях является интерфейс OSPF подключенный к сети/маршрутизатору-адресату. Если адресатом является маршрутизатор, подключенный через сеть Point-to-MultiPoint, IP-адреса next hop можно определить, просматривая router-LSA для адресата - каждый канал, ведущий обратно к рассчитывающему маршрутизатору и имеющий поле Link Data, относящееся к сети Point-to-MultiPoint, обеспечивает IP-адрес следующего маршрутизатора (next hop). Если адресатом является напрямую подключенная сеть или маршрутизатор, соединенный с ведущим расчет маршрутизатором через интерфейс point-to-point, для next hop не требуется адреса IP. Если маршрутизатор-адресат подключен к рассчитывающему маршрутизатору через виртуальный канал, установка next hop должна быть отложена до выполнения расчетов, описанный в 16.3.

Во втором случае родительская вершина является сетью, которая напрямую подключает рассчитывающий маршрутизатор к маршрутизатору-получателю. Список следующих интервалов в этом случае определяется проверкой router-LSA для получателя. Для каждого канала в router-LSA, указывающего обратно на родительскую сеть, поле Link Data для канала содержит IP-адрес следующего маршрутизатора. Используемый в качестве выходного интерфейса можно в результате определить по IP-адресу следующего маршрутизатора или он может быть унаследован от родительской сети.



## 16.2. Расчет междоменных маршрутов

Междоменные маршруты рассчитываются на основе summary-LSA. Если маршрутизатор имеет активные соединения с несколькими областями, принимаются во внимание только summary-LSA магистральной области. Маршрутизаторы, подключенные к одной области, используют только summary-LSA своей области. В любом случае все анонсы summary-LSA, рассматриваемые ниже являются частью базы данных одной области (назовем ее областью А).

Анонсы Summary-LSA генерируются граничными маршрутизаторами областей. При расчете междоменных маршрутов принимаются во внимание все summary-LSA в области А. Напомним, что адресатом, описанным в summary-LSA, может быть сеть (summary-LSA типа 3) или граничный маршрутизатор AS (summary-LSA типа 4). Для каждого анонса summary-LSA выполняется следующее:

- (1) Если в LSA указана стоимость LSInfinity или возраст LSA равен MaxAge, переходим к следующему анонсу LSA.
- (2) Если анонс LSA порожден рассчитывающим маршрутизатором, переходим к следующему анонсу LSA.
- (3) Если summary-LSA относится к типу 3, набор адресатов, описываемых в summary-LSA, совпадает с одним из заданных для области маршрутизатора диапазонов адресов (см. параграф 3.5) и адресный диапазон отдельной области является активным, этот анонс summary-LSA следует игнорировать (активность означает наличие по крайней мере одной сети в диапазоне области, доступной по внутридоменному пути).
- (4) Для остальных случаев назовем адресата, описываемого LSA, N (для summary-LSA типа 3 адрес N получается путем наложения маски сети/подсети из тела LSA на значение Link State ID в LSA), а породивший LSA граничный маршрутизатор области - BR. Будем искать в таблице маршрутов запись для BR имеющую область А в качестве связанной. Если такой записи для BR не существует (т. е., BR недостижим в области А), данный анонс LSA следует пропустить и перейти к следующему. При наличии записи LSA описывает внутридоменный путь к адресату N, стоимость которого равна сумме дистанции до BR и стоимости, указанной в LSA. Обозначим стоимость этого междоменного пути как IAC.
- (5) Далее будем искать в таблице маршрутов запись для адресата N (если N является граничным маршрутизатором AS, будем искать в таблице запись router, связанную с областью А). Если в таблице нет записи для N или запись включает внешний путь типа 1 или 2, устанавливаем междоменный путь к адресату N со связанной областью А, стоимостью IAC, значением next hop, совпадающим со списком next hop к маршрутизатору BR и Advertising router = BR.
- (6) В противном случае, если присутствующие в таблице пути являются внутридоменными, с LSA не нужно делать ничего (внутридоменные пути всегда имеют предпочтение).
- (7) В остальных случаях присутствующие в таблице пути также являются междоменными. Устанавливаем новый путь через BR, если он дешевле, переписывая запись в таблице маршрутов. Если новый путь имеет ту же стоимость, добавляем его к списку путей в записи таблицы маршрутов.

## 16.3. Просмотр summary-LSA от транзитных областей

Этот этап выполняется маршрутизаторами, подключенными к одной или нескольким немагистральным областям, способным передавать транзитный трафик (на этапе 2 алгоритма Dijkstra устанавливается TransitCapability = TRUE, как описано в параграфе 16.1).

Целью описанного ниже расчета является проверка транзитных областей на предмет поиска лучшего (более короткого) пути по сравнению с результатами расчетов, описанных в параграфах 16.1 и 16.2. Любые пути, которые не хуже вычисленных ранее, добавляются в таблицу маршрутизации. Расчет также определяет актуальные значения next hop для тех адресатов, у которых значения next hop, рассчитанные в 16.1 и 16.2 указывают на виртуальные каналы. После завершения описанных ниже расчетов все пути, рассчитанные в 16.1 и 16.2, которые имеют непреобразованные (unresolved) виртуальные значения next hop, должны удаляться.

При расчете просматриваются summary-LSA от всех транзитных областей. Каждый анонс summary-LSA описывает маршрут через транзитную область А в сеть N (адрес N определяется наложением маски сети/подсети из тела LSA на значение Link State ID в LSA) или (для summary-LSA типа 4) к граничному маршрутизатору AS. Предположим, что summary-LSA порождается граничным маршрутизатором области - BR.

- (1) Если стоимость, анонсируемая в summary-LSA, равна LSInfinity или LS age = MaxAge, проверяется следующая запись LSA.
- (2) Если анонс summary-LSA был порожден ведущим расчет маршрутизатором, переходим к проверке следующего анонса LSA.
- (3) Поиск в таблице маршрутов записи для N (если N является граничным маршрутизатором AS, нужно отыскать запись для маршрутизатора, связанную с магистральной областью). Если записи не существует в таблице, путь для записи является внутридоменным или междоменным, или область, связанная с записью в таблице, не относится к магистрали, переходим к проверке следующего анонса LSA. Иными словами, этот расчет только обновляет внутридоменные пути для магистральной области, определенные в параграфе 16.1 и междоменные пути, найденные в 16.2.
- (4) Поиск в таблице записи для анонсирующего маршрутизатора BR, связанного с областью А. Если маршрутизатор недоступен, проверяется следующий анонс LSA. В противном случае стоимость доставки в сеть N равна сумме стоимости в записи базы области А для BR и стоимости, анонсируемой в LSA. Назовем суммарную стоимость IAC.
- (5) Если эта стоимость меньше стоимости доставки в N из записи в таблице маршрутов, переписываем запись next hop для N, указывая next hop для BR, и значение IAC в качестве стоимости для N. Если IAC совпадает с текущей стоимостью для N, добавляем список next hop маршрутизатора BR к списку next hop сети N. В таких случаях область, связанная с записью таблицы маршрутов для N должна оставаться в магистрали, а тип пути (внутридоменный или междоменный) не должен изменяться. Важно подчеркнуть, что приведенный выше расчет никогда не показывает недоступность достижимых адресатов, но может находить более короткие пути к доступным получателям. В результате расчета лучшие пути включаются в таблицу маршрутов и заново анонсируются с помощью summary-LSA в другие области.

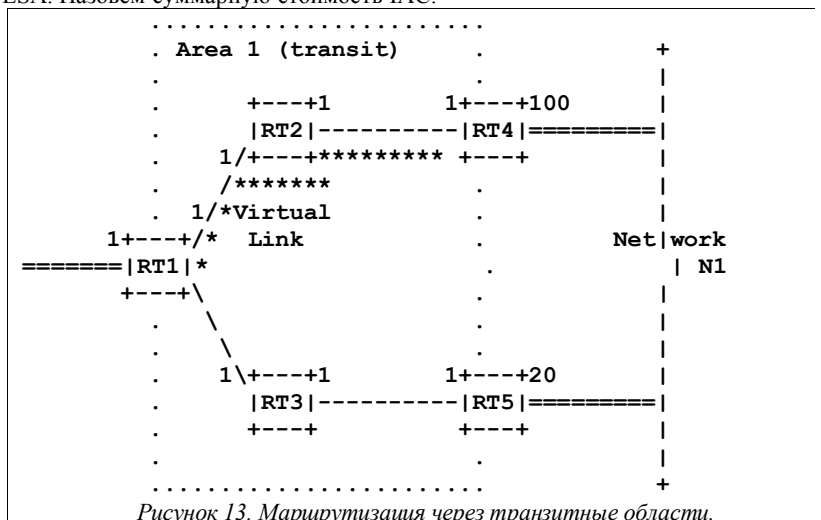


Рисунок 13. Маршрутизация через транзитные области.

В качестве примера расчета рассмотрим автономную систему, показанную на рисунке 13. Здесь имеется одна немагистральная область (Area 1), физически делящая магистраль на две части. Для создания связной магистрали организуется виртуальный канал между маршрутизаторами RT1 и RT4. Сеть N1 в правой части рисунка относится к магистрали. Линии из точек показывают, что

существует значительно более короткий внутридоменный путь в сеть N1 от маршрутизатора RT5 (стоимость 20), нежели от маршрутизатора RT4 (стоимость 100). Оба маршрутизатора RT4 и RT5 будут передавать свои summary-LSA для сети N1 в область Area 1.

После расчета дерева кратчайших путей для магистрали (см. параграф 16.1) маршрутизатор RT1 (левая сторона виртуального канала) будет иметь рассчитанный путь через RT4 для всех пакетов в сеть N1. Однако, поскольку RT5 обеспечивает значительно более короткий путь в сеть N1, все внутренние маршрутизаторы области 1 (например, RT2 и RT3) будут пересылать свои пакеты для сети N1 через маршрутизатор RT5 вместо RT4. После проверки summary-LSA для области 1 с помощью описанной выше процедуры маршрутизатор RT1 также будет пересылать пакеты для сети N1 в направлении RT5. Отметим, что в этом примере виртуальный канал позволяет пересылать транзитный трафик через область 1, но реальные пакеты данных не проходят через виртуальный канал. Иными словами, виртуальный канал делает возможным транзитный трафик, но не требует передачи такого трафика именно через виртуальное соединение.

## 16.4. Расчет внешних маршрутов AS

Внешние маршруты AS рассчитываются путем обработки всех имеющихся анонсов AS-external-LSA. Большинство AS-external-LSA описывают маршруты к конкретным адресатам, но некоторые содержат описания принятых по умолчанию маршрутов для AS (Destination ID = DefaultDestination, маска сети/подсети = 0x00000000). Для AS-external-LSA выполняются следующие операции:

- (1) LSA со стоимостью LSInfinity или возрастом MaxAge исключаются из рассмотрения.
- (2) LSA, порожденные рассчитывающим маршрутизатором, также не рассматриваются.
- (3) Обозначим адресата, описываемого в LSA, как N. Адрес N определяется наложением на Link State ID в LSA маски сети/подсети, содержащейся в теле LSA. В таблице маршрутов ищем запись (потенциально имеется одна запись на подключенную область) для граничного маршрутизатора AS (ASBR), породившего LSA. Если записи для маршрутизатора ASBR не существует (т. е., ASBR недоступен), переходим к следующему анонсу LSA в списке. В остальных случаях анонс LSA описывает внешний путь к адресату N. Адрес пересылки в AS-external-LSA указывает IP-адрес, по которому должны переправляться пакеты для адресата. Если в качестве адреса пересылки указано значение 0.0.0.0, пакеты должны пересылаться маршрутизатору ASBR. При наличии в таблице нескольких записей для ASBR выбирается предпочтительный маршрут с учетом приведенных ниже условий. Если совместимость с RFC-1583 (RFC1583Compatibility) отключена, сокращаем набор записей для ASBR в маршрутной таблице, как описано в параграфе 16.4.1. Из оставшихся записей выбираем маршрут с минимальной стоимостью. Если таких маршрутов несколько, выбираем среди них запись с максимальным значением OSPF Area ID (рассматривая как беззнаковое целое число) для связанной области. Если адрес для пересылки не равен 0, отыскивается адрес пересылки в таблице маршрутов<sup>31</sup>. Соответствующая запись таблицы должна указывать внутридоменный или междоменный путь; если такого пути не существует, с LSA ничего не нужно делать и переходим к следующему анонсу в списке.
- (4) Пусть X указывает стоимость предпочтительного маршрута для ASBR/адреса пересылки, а Y – стоимость, указанную в LSA. X выражается в единицах метрики link state, а Y является внешней метрикой типа 1 или 2.
- (5) Просматриваем таблицу маршрутизации для адресата N. Если записи для N не существует, устанавливается внешний путь к N, для которого next hop совпадает со списком next hop к адресам пересылки, а для анонсирующего маршрутизатора указано значение ASBR. Если внешний путь имеет метрику типа 1, для пути устанавливается тип 1 (external) со стоимостью X+Y. Если внешняя метрика относится к типу 2, для пути устанавливается тип 2 (external), компонента стоимости link state имеет значение X, а стоимость типа 2 равна Y.
- (6) Сравниваем внешний путь, описанный в LSA с существующими в таблице маршрутов путями в сеть N. Если новый путь является предпочтительным, он помещается в таблицу вместо существующих путей в сеть N. Если новый путь имеет такие же предпочтения, он добавляется в список путей маршрутной записи для сети N.
  - (a) Внутридоменные и междоменные маршруты всегда более предпочтительны, нежели внешние маршруты.
  - (b) Внешние пути типа 1 предпочтительней, нежели пути типа 2. Когда все внешние пути относятся к типу 2, среди них предпочтительным является путь с наименьшей анонсируемой метрикой.
  - (c) Если новый внешний путь не отличается от текущего пути в маршрутной записи для сети N и совместимость RFC1583Compatibility отключена, предпочтительный путь выбирается на основе оценки внутренних путей к ASBR/адресу пересылки, как описано в параграфе 16.4.1.
  - (d) Если новый внешний путь по прежнему не отличается от текущего пути в маршрутной записи для сети N, выбираем предпочтительный путь, сравнивая стоимости. Внешние пути типа 1 сравниваются с учетом суммы дистанции до адреса пересылки и анонсируемой метрики типа 1 (X+Y). Внешние пути типа 2, анонсирующие метрику типа 2, сравниваются по дистанции до адреса пересылки.

### 16.4.1. Предпочтения для внешнего пути

При наличии в AS множества путей к ASBR/адресу пересылки для выбора предпочтительного пути используются приведенные здесь правила. Эти правила применяются когда один маршрутизатор ASBR доступен через множество областей или когда требуется отдать предпочтение одному из анонсов AS-external-LSA. В первом случае все пути завершаются на одном маршрутизаторе ASBR, а во втором – на разных ASBR/адресах пересылки. В обоих случаях каждый из путей представляется отдельной записью в таблице маршрутов (см. главу 11).

Приведенные в этом параграфе правила применимы не только при отключенной совместимости RFC1583Compatibility.

Порядок выбора предпочтительного пути показан ниже. Отметим, что в результате применения этих правил может остаться несколько путей с одинаковыми уровнями предпочтения. В таких случаях путь выбирается на основе стоимости (см. параграф 16.4).

- ◆ Сначала выбираются пути внутри области, не использующие магистраль.
- ◆ Внутренние пути через магистраль и междоменные пути имеют одинаковый уровень предпочтения.

## 16.5. Нарастающие обновления – summary-LSA

При получении нового анонса summary-LSA необходимо пересчитывать всю таблицу маршрутизации. Обозначим адресата, анонсируемого в summary-LSA как N (адрес N определяется наложением на значение Link State ID из LSA маски сети/подсети из тела LSA), а область, к которой относится LSA – A. Возможны два разных случая:

- (1) Область A является магистральной и/или маршрутизатор не является граничным. В таких случаях выполняются приведенные ниже расчеты.
  - ◆ Сначала, если присутствует внутридоменный маршрут в сеть N, запись для N объявляется некорректной, но сохраняется для последующего сравнения.

<sup>31</sup> Отличный от 0 адрес пересылки должен указывать на маршрутизатор, относящийся к другой AS (см. параграф 12.4.4).

- ◆ После этого выполняются расчеты, описанные в параграфе 16.2 для одного адресата N. В этом расчете участвуют все summary-LSA области A, описывающие путь в сеть N.
  - ◆ Дополнительно, если маршрутизатор является граничным и подключен к одной или нескольким транзитным областям, нужно снова провести расчеты, описанные в параграфе 16.3, для одного адресата. Если результаты этих расчетов изменяют стоимость/путь к граничному маршрутизатору AS (как в случае summary-LSA типа 4) или любому из адресов пересылки, нужно заново просмотреть все AS-external-LSA, используя расчет из параграфа 16.4. В противном случае, если сеть N недавно стала недоступной, нужно повторить расчеты из параграфа 16.4 для одного адресата N в случае существования альтернативных внешних путей в сеть N.
- (2) Область A является транзитной, а маршрутизатор – граничным для области. В таких случаях требуется провести описанные ниже расчеты.
- ◆ Сначала, если запись в таблице маршрутов для N содержит один или несколько междоменных путей с использованием области A, эти пути следует удалить. Если из записи удаляются все пути, запись отмечают как некорректную. Старые данные из записи должны быть сохранены для последующего сравнения.
  - ◆ После этого выполняются расчеты, описанные в параграфе 16.3 для одного адресата N. Если в результате расчетов стоимость пути в N возрастает, нужно заново пересчитать всю таблицу маршрутизации с использованием алгоритма Dijkstra, как описано в параграфе 16.1.
  - ◆ В остальных случаях, если стоимость/путь к граничному маршрутизатору AS (как в случае summary-LSA типа 4) или любому из адресов пересылки, нужно заново просмотреть все AS-external-LSA, используя расчет из параграфа 16.4. В противном случае, если сеть N недавно стала недоступной, нужно повторить расчеты из параграфа 16.4 для одного адресата N в случае существования альтернативных внешних путей в сеть N.

## 16.6. Нарастающие обновления – AS-external-LSA

При получении нового анонса AS-external-LSA не требуется пересчета всей таблицы маршрутов. Обозначим описываемого AS-external-LSA адресата как N. Адрес N можно определить, применяя маску из тела LSA к значению поля Link State ID. Если к адресату имеется внутридоменный или междоменный маршрут, не требуется никаких расчетов (внутренний путь имеет предпочтение).

При отсутствии внутренних путей используется расчет, описанный в параграфе 16.4, выполняемый только для анонсов AS-external-LSA, указывающих путь в сеть N. Перед выполнением расчетов запись в таблице для адресата N объявляется некорректной.

## 16.7. События, генерируемые при изменении таблицы маршрутов

Изменения в таблице маршрутов иногда требуют от граничных маршрутизаторов OSPF выполнения добавочных действий. К числу таких изменений относятся следующие:

- ◆ Изменилась стоимость или тип пути для записи таблицы. Если данная запись содержит путь в сеть или к граничному маршрутизатору AS и это не простое изменение внешнего маршрута, может потребоваться генерация нового анонса summary-LSA (возможно для каждой из подключенных областей, включая магистральную). Подробная информация приведена в параграфе 12.4.3. Если ранее анонсированная запись была удалена или больше не может анонсироваться для определенной области, требуется удалить LSA из домена маршрутизации путем установки для них возраста MaxAge и повторения лавинной рассылки (см. параграф 14.1).
- ◆ Запись таблицы маршрутов, связанная с виртуальным каналом изменена. Это изменение показывает смену стоимости или доступности виртуального соединения.  
Если запись показывает, что граничный маршрутизатор области стал доступен недавно, соответствующий виртуальный канал работает. Для виртуального канала должно генерироваться событие InterfaceUp, которое будет инициировать формирование смежности (см. параграф 10.3). В этот момент также определяется IP-адрес интерфейса и Neighbor IP виртуального соседа. Если запись показывает, что граничный маршрутизатор области более не доступен, виртуальный канал и связанные с ним отношения смежности уничтожаются. Это означает необходимость генерации события InterfaceDown для связанного виртуального канала.  
Если стоимость для записи изменена и существует полная виртуальная смежность, должен генерироваться новый анонс router-LSA для магистрали, что может потребовать дальнейших изменений в таблице маршрутизации.

## 16.8. Множество равноценных путей

Протокол OSPF поддерживает множество равноценных путей ко всем получателям. Эту поддержку можно видеть в описанных выше этапах расчета таблицы маршрутов и в определении структуры таблицы маршрутизации.

Все маршруты из множества равноценных имеют один тип (внутридоменный, междоменный, внешний типа 1 или 2), равную стоимость и связаны с одной областью. Однако каждый маршрут может указывать свое значение next hop и Advertising router.

От маршрутизаторов OSPF не требуется сохранять все возможные равноценные пути к адресату. Разработчики могут ограничиться сохранением фиксированного числа равноценных путей к любому из адресатов – это не повлияет на работу описанных алгоритмов.

## Литература

- [Ref1] McQuillan, J., I. Richer and E. Rosen, "ARPANET Routing Algorithm Improvements", BBN Technical Report 3803, April 1978.  
[Ref2] Digital Equipment Corporation, "Information processing systems -- Data communications -- Intermediate System to Intermediate System Intra-Domain Routing Protocol", October 1987.  
[Ref3] McQuillan, J., et.al., "The New Routing Algorithm for the ARPANET", IEEE Transactions on Communications, May 1980.  
[Ref4] Perlman, R., "Fault-Tolerant Broadcast of Routing Information", Computer Networks, December 1983.  
[Ref5] Postel, J., "Internet Protocol", STD 5, RFC 791<sup>32</sup>, September 1981.  
[Ref6] McKenzie, A., "ISO Transport Protocol specification ISO DP 8073", RFC 905, April 1984.  
[Ref7] Deering, S., "Host extensions for IP multicasting", STD 5, RFC 1112<sup>33</sup>, May 1988.

<sup>32</sup> Перевод спецификации протокола IP вы можете найти на сайте [www.protocols.ru](http://www.protocols.ru). Прим. перев.

<sup>33</sup> Перевод RFC 1112 имеется на сайте [www.protocols.ru](http://www.protocols.ru). RFC 2236 содержит ряд дополнений и изменений для этого документа. Прим. перев.

- [Ref8] McCloghrie, K., and M. Rose, "Management Information Base for network management of TCP/IP-based internets: MIB-II", STD 17, RFC 1213<sup>34</sup>, March 1991.
- [Ref9] Moy, J., "OSPF Version 2", RFC 1583<sup>35</sup>, March 1994.
- [Ref10] Fuller, V., T. Li, J. Yu, and K. Varadhan, "Classless Inter-Domain Routing (CIDR): an Address Assignment and Aggregation Strategy", RFC1519<sup>36</sup>, September 1993.
- [Ref11] Reynolds, J., and J. Postel, "Assigned Numbers", STD 2, RFC 1700<sup>37</sup>, October 1994.
- [Ref12] Almquist, P., "Type of Service in the Internet Protocol Suite", RFC 1349<sup>38</sup>, July 1992.
- [Ref13] Leiner, B., et.al., "The DARPA Internet Protocol Suite", DDN Protocol Handbook, April 1985.
- [Ref14] Bradley, T., and C. Brown, "Inverse Address Resolution Protocol", RFC 1293<sup>39</sup>, January 1992.
- [Ref15] deSouza, O., and M. Rodrigues, "Guidelines for Running OSPF Over Frame Relay Networks", RFC 1586, March 1994.
- [Ref16] Bellovin, S., "Security Problems in the TCP/IP Protocol Suite", ACM Computer Communications Review, Volume 19, Number 2, pp. 32-38, April 1989.
- [Ref17] Rivest, R., "The MD5 Message-Digest Algorithm", RFC 1321<sup>36</sup>, April 1992.
- [Ref18] Moy, J., "Multicast Extensions to OSPF", RFC 1584, March 1994.
- [Ref19] Coltun, R., and V. Fuller, "The OSPF NSSA Option", RFC 1587, March 1994.
- [Ref20] Ferguson, D., "The OSPF External Attributes LSA", work in progress.
- [Ref21] Moy, J., "Extending OSPF to Support Demand Circuits", RFC 1793, April 1995.
- [Ref22] Mogul, J., and S. Deering, "Path MTU Discovery", RFC 1191<sup>36</sup>, November 1990.
- [Ref23] Rekhter, Y., and T. Li, "A Border Gateway Protocol 4 (BGP-4)", RFC 1771<sup>40</sup>, March 1995.
- [Ref24] Hinden, R., "Internet Routing Protocol Standardization Criteria", BBN, October 1991.
- [Ref25] Moy, J., "OSPF Version 2", RFC 2178<sup>41</sup>, July 1997.
- [Ref26] Rosen, E., "Vulnerabilities of Network Control Protocols: An Example", Computer Communication Review, July 1981.

<sup>34</sup> RFC 2011 – RFC 2013 содержат ряд дополнений и изменений для этого документа. *Прим. перев.*

<sup>35</sup> Этот документ устарел и заменен RFC 2178, который, в свою очередь, отменяет данная спецификация. *Прим. перев.*

<sup>36</sup> Перевод этого документа имеется на сайте <http://www.protocols.ru>. *Прим. перев.*

<sup>37</sup> В соответствии с RFC 3232 документ STD 2 утратил силу. Значения Assigned Numbers следует искать в базе данных, доступной на сайте [www.iana.org/numbers.html](http://www.iana.org/numbers.html). *Прим. перев.*

<sup>38</sup> Этот документ устарел и заменен RFC 2474. *Прим. перев.*

<sup>39</sup> Этот документ устарел и заменен RFC 2390. *Прим. перев.*

<sup>40</sup> Этот документ устарел и заменен RFC 4271. Перевод имеется на сайте <http://www.protocols.ru>. *Прим. перев.*

<sup>41</sup> Этот документ устарел и отменяется настоящей спецификацией. *Прим. перев.*



## Приложения

### А. Форматы данных OSPF

В этом приложении описаны форматы пакетов OSPF и анонсов OSPF LSA. Протокол OSPF работает непосредственно поверх протокола сетевого уровня IP. Перед описанием форматов детально рассматривается инкапсуляция OSPF.

Далее рассматривается поле опций OSPF (Options), описывающее различные дополнительные возможности, которые могут поддерживаться отдельными частями домена маршрутизации OSPF. Поле опций OSPF содержится в пакетах Hello, DD и анонсах LSA.

Форматы пакетов OSPF описаны в разделе А.3, а описание LSA дается в разделе А.4.

#### А.1 Инкапсуляция пакетов OSPF

OSPF работает непосредственно поверх протокола IP и пакеты OSPF, следовательно, инкапсулируются с использованием только заголовков IP и канального уровня.

OSPF не определяет способа фрагментации для своих пакетов и полагается на фрагментацию IP при передаче в сетях с малым значением MTU. При необходимости размер пакетов OSPF может достигать 65 535 байтов (включая заголовок IP). Пакеты тех типов OSPF, которые могут достигать большого размера (DD, LSR, LSU и LSAck), обычно можно разделить на несколько более мелких пакетов без потери функциональности. Рекомендуется поступать именно так, избегая фрагментации IP. По тем же причинам следует предпринимались попытки ограничения размера пакетов OSPF, передаваемых через виртуальные соединения, значением 576 байтов, пока не используется Path MTU Discovery [Ref22].

Другими важными особенностями инкапсуляции OSPF являются:

- ◆ Использование IP multicast. Некоторые сообщения OSPF являются групповыми при их передаче через широковещательные сети. Используются два разных адреса для групповой передачи. Передаваемые с использованием этих адресов пакеты не должны пересылаться, это значит, что они передаются только на один интервал (hop). Для предотвращения дальнейшей пересылки должно ограничиваться время жизни пакетов (TTL = 1).

##### AllSPFRouters

Эта константа обозначает групповой адрес 224.0.0.5. Все маршрутизаторы OSPF должны быть готовы к приему пакетов, переданных по этому адресу. Пакеты Hello всегда передаются по адресу AllSPFRouters. Кроме того, этот адрес используется для передачи некоторых пакетов OSPF в процессе лавинной рассылки.

##### AllDRouters

Эта константа используется для обозначения группового адреса 224.0.0.6. Оба маршрутизатора DR и Backup DR должны быть готовы к приему пакетов, направленных по этому адресу. Некоторые пакеты OSPF передаются по этому адресу в процессе лавинной рассылки.

- ◆ Протокол OSPF имеет в IP номер 89, зарегистрированный в NIC. Распределение номеров для протоколов IP описано в [Ref11].
- ◆ Все служебные пакеты OSPF передаются с использованием значения TOS для нормального обслуживания (0000 в соответствии с работой [Ref12]).
- ◆ Для пакетов протокола маршрутизации уровень предпочтений IP устанавливается в Internetwork Control. Пакеты OSPF должны иметь преимущество перед обычными пакетами данных IP как для приема, так и для передачи. Установка в поле IP precedence заголовка IP значения Internetwork Control [Ref5] помогает обеспечить такое преимущество.

#### А.2 Поле Options

Поле опций OSPF присутствует в пакетах Hello, DD и всех типах LSA. Это поле позволяет маршрутизаторам OSPF поддерживать (или не поддерживать) дополнительные возможности и сообщать уровень поддержки другим маршрутизаторам OSPF. Этот механизм позволяет смешивать разные маршрутизаторы в домене маршрутизации OSPF.

При использовании в пакетах Hello поле Options позволяет маршрутизатору отказаться от соседа при рассогласовании возможностей. Зная о возможностях соседей из пакетов DD, маршрутизатор может не передавать некоторые анонсы LSA соседям, не поддерживающим соответствующих функций. И, наконец, перечисление возможностей в LSA позволяет маршрутизаторам пересылать трафик в обход маршрутизаторов с ограниченными возможностями, исключая такие маршрутизаторы при расчете путей для таблицы маршрутизации.

В поле OSPF Options выделены 5 битов, хотя данная спецификация полностью описывает только один из них (E). Краткое описание всех флагов приведено ниже. Маршрутизатор должен сбрасывать в 0 нераспознанные биты поля Options при передаче пакетов Hello и DD, а также при генерации LSA. Маршрутизатор, получивший неизвестные биты опций в пакетах Hello и DD или анонсах LSA, должен игнорировать эти биты, обрабатывая известные флаги как обычно.

**E**

Описывает способ лавинной рассылки AS-external-LSA (см. параграфы 3.6, 9.5, 10.8 и 12.1.2).

**MC**

Этот бит устанавливается при рассылке групповых (multicast) дейтаграмм IP в соответствии со спецификацией [Ref18].

**N/P**

Флаг обработки LSA типа 7 в соответствии с [Ref19].

**EA**

Флаг приема и пересылки анонсов External-Attributes-LSA в соответствии с [Ref20].

**DC**

Флаг обработки demand circuit в соответствии с [Ref21].

*	*	D	E	N/	M	E	*
C	A	P	C				

Поле Options

#### А.3 Форматы пакетов OSPF

Существует 5 различных форматов пакетов OSPF. Все пакеты OSPF начинаются со стандартного 24-байтового заголовка, описанного в следующем параграфе. После заголовка описаны форматы каждого типа пакетов и рассмотрены поля этих пакетов. Пакеты OSPF всех типов за исключением пакетов Hello, предназначены для работы со списками LSA. Например, пакеты LSU обеспечивают лавинную рассылку LSA в домене маршрутизации OSPF. По этой причине пакеты протокола OSPF невозможно разобрать, не понимая форматов LSA, описанных в разделе А.4.

Обработка принимаемых пакетов OSPF описана в параграфе 8.2, а передача пакетов OSPF – в параграфе 8.1.

### A.3.1 Заголовок пакета OSPF

Каждый пакет OSPF начинается со стандартного заголовка размером 24 байта, содержащего всю информацию, требуемую для дальнейшей обработки пакета (см. параграф 8.2).

#### Version #

Номер версии протокола OSPF. Данная спецификация описывает версию 2.

#### Type

Типы пакетов OSPF перечислены в таблице справа и описаны в параграфах A.3.2 - A.3.6.

#### Packet length

Размер пакета OSPF в байтах с учетом стандартного заголовка OSPF.

#### Router ID

Идентификатор Router ID отправителя пакета.

#### Area ID

32-битовый идентификатор области, к которой относится пакет. Все пакеты OSPF связываются с определенной (единственной) областью. Большинство пакетов передается лишь на один интервал (hop). Пакеты, передаваемые через виртуальные каналы, помечаются идентификатором магистральной области 0.0.0.0.

#### Checksum

Стандартная контрольная сумма IP для всего пакета, начиная с заголовка OSPF, но без учета 64-битового поля аутентификации. Если длина пакета не равна целому числу 16-битовых слов, пакет дополняется справа соответствующим числом нулей до вычисления контрольной суммы. Определение контрольной суммы рассматривается как часть процесса аутентификации, но для некоторых типов аутентификации подсчет контрольной опускается.

#### AuType

Определяет используемую для пакета процедуру аутентификации (см. Приложение D).

#### Authentication

64-битовое поле, используемое схемой аутентификации (см. Приложение D).

Тип	Название
1	Hello
2	Database Description
3	Link State Request
4	Link State Update
5	Link State Acknowledgment

										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								
Version #										Type										Packet length																			
Router ID																																							
Area ID																																							
Checksum															AuType																								
Authentication																																							
Authentication																																							

### A.3.2 Пакет Hello

Пакеты Hello относятся к типу 1 и передаются периодически во все интерфейсы (включая виртуальные каналы) для организации и поддержки соседских отношений. Кроме того, пакеты Hello рассылаются по групповым адресам в сетях, обеспечивающих такую возможность, для динамического обнаружения соседних маршрутизаторов.

Для всех маршрутизаторов, подключенных к общей сети, должны быть согласованы параметры Network mask, HelloInterval и RouterDeadInterval. Эти параметры включаются в пакеты Hello, поэтому рассогласование может влиять на формирование соседских отношений. Подробное описание обработки приема пакетов Hello приведено в параграфе 10.5, а передача пакетов Hello описана в параграфе 9.5.

#### Network mask

Маска сети, связанной с интерфейсом. Например, сети класса В имеют маску 0xfffff00.

#### Options

Дополнительные возможности, поддерживаемые маршрутизатором (см. параграф A.2).

#### HelloInterval

Число секунд между передачей пакетов Hello.

#### Rtr Pri

Значение Router Priority для маршрутизатора, используемое при выборе (Backup) DR. Нулевое значение блокирует возможность стать (Backup) DR.

#### RouterDeadInterval

Число секунд до объявления «затихшего» маршрутизатора неработоспособным.

#### Designated Router

Указывает выделенный маршрутизатор DR для данной сети с точки зрения передающего маршрутизатора. DR идентифицируется по IP-адресу в сети. Если маршрутизатора DR нет, поле имеет значение 0.0.0.0.

#### Backup Designated Router

Идентифицирует маршрутизатор Backup DR с точки зрения передающего пакет маршрутизатора. Backup DR идентифицируется адресом IP для интерфейса в данную сеть. При отсутствии Backup DR это поле содержит значение 0.0.0.0.

#### Neighbor

Значения Router ID для каждого маршрутизатора, чьи пакеты Hello были недавно (RouterDeadInterval секунд) видны в сети.

### A.3.3 Пакет Database Description

Пакеты DD относятся в OSPF к типу 2. Обмен этими пакетами происходит при инициализации отношений смежности. Пакеты описывают содержимое базы данных о каналах. Для описания базы данных может использоваться множество пакетов с использованием процедуры poll-response (опрос-отклик). Один из маршрутизаторов является ведущим (master), а второй – ведомым (slave). Ведущий маршрутизатор передает пакеты DD, которые подтверждаются пакетами DD от ведомого маршрутизатора. Отклики связываются с опросом через порядковые номера пакетов DD.

Формат пакетов DD очень похож на формат пакетов LSR и LSack. Все эти три типа пакетов содержат список элементов, каждый из которых описывает часть базы данных о каналах маршрутизатора. Передача пакетов DD описана в параграфе 10.8, а обработка принимаемых пакетов DD – в параграфе 10.6.

#### Interface MTU

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								
Version #										1										Packet length																			
Router ID																																							
Area ID																																							
Checksum															AuType																								
Authentication																																							
Authentication																																							
Network Mask																																							
HelloInterval															Options										Rtr Pri														
RouterDeadInterval																																							
Backup Designated Router																																							
Neighbor																																							
...																																							

Размер (в байтах) максимальной дейтаграммы IP, которая может быть передана через интерфейс без фрагментации. Значения MTU для типовых каналов Internet можно найти в таблице 7-1 работы [Ref22]. В пакетах DD, передаваемых через виртуальные каналы, должно устанавливаться Interface MTU = 0.

### Options

Дополнительные возможности маршрутизатора (см. параграф A.2).

**I**

Бит Init, устанавливаемый для первого (по порядку) пакета DD.

**M**

Бит More, указывающий на присутствие других (последующих) пакетов DD.

**MS**

Бит Master/Slave, определяющий отношения маршрутизаторов в процессе Database Exchange (MS=1 – ведущий, 0 – ведомый).

### DD sequence number

Используется для нумерации пакетов DD. Начальное значение (указывается флагом Init) должно быть уникальным. Далее порядковый номер DD увеличивается на 1 для каждого пакета вплоть до передачи всей базы данных.

Остальная часть пакета содержит (возможно неполный) список частей базы данных link-state. Каждая запись LSA в базе описывается заголовком LSA (см. параграф A.4.1), содержащим все данные для уникальной идентификации LSA и текущего экземпляра.

## A.3.4 Пакет Link State Request

Пакеты LSR относятся в OSPF к типу 3. После обмена пакетами DD с соседом маршрутизатор может понять, что часть его базы данных устарела. Пакеты LSR служат для запроса более современных фрагментов базы данных у соседа. При обновлении может использоваться множество пакетов LSR.

Маршрутизатор, передающий запрос LSR, должен помнить конкретный экземпляр запрашиваемого фрагмента базы данных. Экземпляры идентифицируются порядковым номером, возрастом и контрольной суммой, но эти поля не указываются в самом пакете LSR.

Маршрутизатор в ответ на запрос может получить даже более свежий (по сравнению с запрошенным) экземпляр. Передача пакетов LSR описана в параграфе 10.9, а прием – в параграфе 10.7.

Каждый запрошенный анонс LSA указывается его типом, а также значениями Link State ID и Advertising Router. Эти параметры уникально описывают LSA, но не конкретный экземпляр анонса. Понятно, что пакеты LSR используются для запроса последнего экземпляра анонса.

## A.3.5 Пакет Link State Update

Пакеты LSU относятся в OSPF к типу 4 и используются для лавинной рассылки LSA. Каждый пакет LSU передает набор LSA на один интервал от точки их происхождения. В одном пакете может содержаться множество LSA.

Пакеты LSU являются групповыми для сетей поддерживающих широковещательную и групповую адресацию. Для обеспечения надежности процедуры лавинной рассылки отправленные LSA подтверждаются в пакетах LSAck. Если требуется повтор передачи некоторых LSA, они всегда передаются соседу напрямую. Дополнительные сведения о надежности лавинной рассылки приведены в главе 13.

### # LSAs

Число LSA, включенных в этот пакет обновления.

Тело пакетов LSU содержит список LSA, начинающихся со стандартного 20-байтового заголовка, описанного в A.4.1. Описания различных типов LSA приведены в A.4.

## A.3.6 Пакет Link State Acknowledgment

Пакеты LSAck относятся в OSPF к типу 5. Для обеспечения надежности лавинной рассылки LSA, все анонсы должны подтверждаться в явной форме. Подтверждения обеспечиваются с помощью пакетов LSAck, каждый из которых может подтверждать прием множества LSA.

В зависимости от состояния передающего интерфейса и отправителя соответствующего пакета LSU, пакеты LSAck передаются по групповым (AllSPFRouters или AllDRouters) или индивидуальным адресам. Передача пакетов LSAck описана в параграфе 13.5, а обработка принятых пакетов – в параграфе 13.7.

Формат пакетов подтверждения походит на формат пакетов DD. Тело пакета просто содержит список заголовков подтверждаемых LSA.

Каждый подтверждаемый анонс LSA описывается заголовком LSA (см. параграф A.4.1), содержащим все сведения для идентификации LSA и текущего экземпляра.

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								
Version #										2										Packet length																			
Router ID																																							
Area ID																																							
Checksum															AuType																								
Authentication																																							
Authentication																																							
Interface MTU															Options					0	0	0	0	0	0	1	M	MS											
DD sequence number																																							
LSA Header																																							

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								
Version #										3										Packet length																			
Router ID																																							
Area ID																																							
Checksum															AuType																								
Authentication																																							
Authentication																																							
LS type																																							
Link State ID																																							
Advertising Router																																							
...																																							

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
Version #										4										Packet length											
Router ID																															
Area ID																															
Checksum																AuType															
Authentication																															
Authentication																															
# LSA																															
LSA																															
...																															

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	
Version #										5										Packet length												
Router ID																																
Area ID																																
Checksum															AuType																	
Authentication																																
Authentication																																
LSA Header																																

## A.4 Форматы LSA

В этом документе описывается 5 различных типов LSA. Каждый анонс LSA начинается со стандартного 20-байтового заголовка LSA, описанного в параграфе A.4.1. В последующих параграфах описаны LSA всех типов.

Каждый LSA описывает часть маршрутного домена OSPF. Каждый маршрутизатор порождает router-LSA. В дополнение к этому при выборе маршрутизатора в качестве Designated Router, этот маршрутизатор порождает network-LSA. Кроме того, маршрутизаторы могут порождать и другие типы LSAs (см. Параграф 12.4). Для всех LSA выполняется лавинная рассылка через домен маршрутизации OSPF. Алгоритм лавинной рассылки является надежным способом обеспечения доставки всем маршрутизаторам одного набора LSA. Дополнительные сведения о лавинной маршрутизации приведены в главе 13. Рассылаемый набор LSA называют базой данных о состоянии каналов.

Из базы данных о состоянии каналов каждый маршрутизатор создает дерево кратчайших путей с собой в качестве корня. Это дает в результате таблицу маршрутизации (см. главу 11). Детальное описание процесса построения таблицы маршрутизации дано в главе 16.

### A.4.1 Заголовок LSA

Все записи LSA начинаются с однотипного заголовка размером 20 байтов. Информации из заголовка достаточно для уникальной идентификации LSA (LS type, Link State ID и Advertising Router). В области маршрутизации может одновременно существовать множество экземпляров LSA. Для определения последнего из них служат поля LS age, LS sequence number и LS checksum из заголовка LSA.

 $LS\ age$ 

Время (в секундах) с момента генерации LSA.

## Options

Дополнительные возможности, которые поддерживаются в описываемой части домена маршрутизации. Дополнительные возможности OSPF описаны в параграфе A.2.

*LS type*

Тип LSA, определяющий формат анонса. Определенные в данной спецификации типы LSA показаны в таблице справа и описаны в параграфе 12.1.3.

**Link State ID**

Это поле идентифицирует часть сети, описываемую анонсом LSA. Содержимое поля зависит от типа LSA. Например, в network-LSA поле Link State ID содержит IP-адрес интерфейса маршрутизатора DR для данной сети (по этому адресу определяется номер сети IP). Описание поля Link State ID приведено также в параграфе 12.1.4.

Тип	Описание
1	Router-LSA
2	Network-LSA
3	Summary-LSA (IP-сеть)
4	Summary-LSA (ASBR)
5	AS-external-LSA

### Advertising Router

Значение Router ID маршрутизатора, породившего LSA (в network-LSA это поле содержит Router ID маршрутизатора DR).

*LS sequence number*

Порядковый номер служит для обнаружения дубликатов и старых LSA. Следующий экземпляр LSA имеет следующий порядковый номер (см. параграф 12.1.6).

*LS checksum*

Контрольная сумма Флэтчера (Fletcher) для всего содержимого LSA, включая заголовок LSA, но без учета поля возраста LS. Детальное описание приведено в параграфе 12.1.7.

*length*

Размер LSA в байтах с учетом 20-байтового заголовка LSA.

### A.4.2 Router-LSA

Router-LSA относятся к типу 1 и порождаются каждым маршрутизатором области. Эти LSA описывают состояние и стоимость каналов (интерфейсов) маршрутизатора в область. Все каналы маршрутизатора в область должны описываться в одном анонсе router-LSA. Описание router-LSA приведено в параграфе 12.4.1.

В router-LSA поле Link State ID содержит OSPF Router ID. Анонсы Router-LSA рассылаются в пределах области.

$$V$$

Этот флаг устанавливается в тех случаях, когда маршрутизатор является конечной точкой одного или нескольких виртуальных каналов с полной смежностью, для которых описываемая область является транзитной ( $V$  – конечная точка виртуального канала).

 $E$ 

Этот флаг устанавливается для граничных маршрутизаторов AS (E - external).

$$B$$

Этот флаг устанавливается для граничных маршрутизаторов области (B - border).

# *links*

Число каналов маршрутизатора, описываемых в LSA (должно совпадать в числом интерфейсов маршрутизатора в область).

Перечисленные ниже поля используются для описания каждого канала (т. е., интерфейса) маршрутизатора. Каждый канал имеет определенный тип (поле Type) – канал в транзитную сеть, к другому маршрутизатору или в тупиковую сеть. Значения всех остальных полей, описывающих канал маршрутизатора, зависят от типа канала. Например, каждый канал имеет связанное с ним 32-битовое поле Link Data. Для каналов в тупиковые сети это поле содержит маску адреса, а для остальных типов – IP-адрес интерфейса маршрутизатора.

## Type

Краткое описание канала маршрутизатора (см. таблицу справа) Отметим, что маршруты к хостам классифицируются как каналы в тупиковые сети с маской 0xfffffff.

										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								
LS age										Options										LS type																			
Link State ID																																							
Advertising Router																																							
LS sequence number																																							
LS checksum																length																							

										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								
LS age															Options										1														
Link State ID																																							
Advertising Router																																							
LS sequence number																																							
LS checksum															length																								
0		V		E		B		0							# links																								
Link ID																																							
Link Data																																							
Type					# TOS										metric																								
...																																							
TOS					0										TOS metric																								
Link ID																																							
Link Data																																							
...																																							

Тип	Описание
1	Соединение Point-to-point с другим маршрутизатором.
2	Соединение с транзитной сетью.
3	Соединение с тупиковой сетью.
4	Виртуальный канал.



Тип	Идентификатор
1	Router ID соседнего маршрутизатора
2	IP-адрес маршрутизатора DR
3	IP-номер сети/подсети
4	Router ID соседнего маршрутизатора

Идентифицирует объект, к которому подключен маршрутизатор. Содержимое поля зависит от типа соединения (поле Type). При соединении с объектом, генерирующим LSA (другой маршрутизатор или транзитная сеть) поле Link ID содержит значение Link State ID из LSA соседа. Это обеспечивает ключ поиска LSA соседей в базе данных при расчете таблицы маршрутов (см. 12.2).

#### Link Data

Содержимое этого поля также зависит от типа соединения (поле Type). Для соединений с тупиковыми сетями поле Link Data содержит маску IP, для безадресных соединений point-to-point – значение ifIndex (MIB-II для интерфейса [Ref8]), а для остальных типов соединений – IP-адрес интерфейса маршрутизатора. Это поле обеспечивает последнюю часть информации, требуемой для построения таблицы маршрутов когда рассчитывается IP-адрес следующего маршрутизатора (next hop). Более подробное описание приведено в параграфе 16.1.1.

#### # TOS

Число различных метрик TOS, заданных для данного соединения, без учета требуемой метрики канала (обозначается как метрика TOS 0, [Ref9]). Например, при отсутствии дополнительной метрики TOS это поле имеет значение 0.

#### metric

Стоимость использования этого канала маршрутизатора.

В анонс может также включаться дополнительная информация, связанная с TOS и используемая для совместимости с предыдущими версиями OSPF [Ref9]. Для каждого канала, которому нужна информация TOS, могут использоваться следующие поля:

#### TOS

Тип сервиса (Type of Service), к которому относится метрика. Представление TOS в OSPF LSA описано в параграфе 12.3.

#### TOS metric

Связанная с TOS метрическая информация.

### A.4.3 Network-LSA

Network-LSA относятся к типу 2 и генерируются для каждой широковещательной и NBMA-сети в области, поддерживающей более 1 маршрутизатора. Анонсы network-LSA порождаются выделенным маршрутизатором DR и описывают все подключенные к сети маршрутизаторы, включая DR. Поле Link State ID содержит IP-адрес интерфейса DR. Дистанция от сети до всех подключенных маршрутизаторов равна 0, поэтому поле метрики не включается в network-LSA. Описание network-LSA дано в параграфе 12.4.2.

#### Network Mask

Адресная маска сети (например, IP-сеть класса А имеет маску 0xff000000).

#### Attached Router

Идентификаторы Router ID каждого из подключенных к сети маршрутизаторов. Реально перечисляются только маршрутизаторы, установившие полные отношения смежности с DR, и сам маршрутизатор DR. Число включенных в список маршрутизаторов можно определить из поля длины в заголовке LSA.

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		
LS age										Options										LS type													
Link State ID																																	
Advertising Router																																	
LS sequence number																																	
LS checksum																				length													
Network Mask																																	
Attached Router																																	

### A.4.4 Summary-LSA

Summary-LSA относятся к типам 3 и 4 и генерируются граничными маршрутизаторами областей. Анонсы Summary-LSA описывают междоменные маршруты (см. параграф 12.4.3).

Анонсы summary-LSA типа 3 используются в тех случаях, когда адресатом является сеть IP. В этом случае поле Link State ID содержит IP-номер сети (при необходимости Link State ID может включать также биты адреса хоста, как описано в Приложении Е). Когда адресатом является граничный маршрутизатор AS, используется summary-LSA типа 4 и поле Link State ID содержит OSPF Router ID граничного маршрутизатора AS (необходимость анонсирования местоположения каждого ASBR рассмотрена в параграфе 16.4.). За исключением различий в использовании поля Link State ID анонсы summary-LSA типов 3 и 4 идентичны.

Для тупиковых областей summary-LSA типа 3 могут также использоваться для описания принятых по умолчанию маршрутов (по областям). Такие маршруты используются в тупиковых областях взамен лавинной рассылки полного набора внешних маршрутов. При описании используемого по умолчанию маршрута поле Link State ID в summary-LSA всегда содержит значение DefaultDestination (0.0.0.0), а Network Mask = 0.0.0.0.

#### Network Mask

Для summary-LSA типа 3 это поле показывает маску сети адресата. Например, при анонсировании адресата из сети класса А будет использоваться значение 0xff000000. Это поле не используется в summary-LSA типа 4 и должно иметь нулевое значение.

#### metric

Стоимость маршрута, выраженная в таких же единицах, как стоимость в анонсах router-LSA.

Дополнительная информация, связанная с TOS, может включаться для обеспечения обратной совместимости с предыдущей версией OSPF [Ref9]. Для всех желаемых уровней TOS информация представляется следующим образом:

#### TOS

Тип сервиса (Type of Service), к которому относится метрика. Представление TOS в OSPF LSA описано в параграфе 12.3.

#### TOS metric

Связанная с TOS метрическая информация.

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																																
LS age																Options																3 или 4																															
Link State ID																																																															
Advertising Router																																																															
LS sequence number																																																															
LS checksum																								length																																							
Network Mask																																																															
0												metric																																																			
TOS												TOS metric																																																			

### A.4.5 AS-external-LSA

AS-external-LSA относятся к типу 5 и генерируются граничными маршрутизаторами AS для описания внешних маршрутов из AS. Детальное описание AS-external-LSA приведено в параграфе 12.4.3.

Анонсы AS-external-LSA обычно описывают маршрут к отдельному адресату (сети). Для таких LSA поле Link State ID содержит IP-номер сети (при необходимости в него могут также включаться некоторые биты адреса хоста, как описано в Приложении E). Анонсы AS-external-LSA также используются для описания принятых по умолчанию маршрутов (эти маршруты применяются в случае отсутствия другого пути к адресату). В таких случаях поле Link State ID всегда содержит значение DefaultDestination (0.0.0.0), а поле Network Mask имеет значение 0.0.0.0.

#### Network Mask

Адресная маска IP для анонсируемого получателя. Например, для сети класса A маска имеет значение 0xff000000.

**E**

Тип внешней метрики. E=1 говорит о внешней метрике типа 2, которая заведомо больше метрики любого из внутренних путей. E=0 говорит о внешней метрике типа 1, использующей такие же единицы, как и внутренняя метрика link state.

#### metric

Стоимость маршрута. Интерпретация этого поля зависит индикации типа внешней метрики (бит E).

#### Forwarding address

Пакеты данных для анонсируемого получателя будут пересылаться по этому адресу. Если Forwarding address = 0.0.0.0, пакеты будут пересылаться породившему LSA маршрутизатору (т. е., граничному маршрутизатору AS).

#### External Route Tag

32-битовое поле, подключаемое к внешним маршрутам. Протокол OSPF сам по себе не использует это поле, оно может применяться при обмене информацией между граничными маршрутизаторами AS (описание этого обмена выходит за пределы данной спецификации).

Для обратной совместимости с предыдущими версиями OSPF [Ref9] может включаться информация, связанная с TOS. Для каждого желаемого значения TOS эта информация представляется следующим образом:

#### TOS

Тип обслуживания для следующих полей. Кодирование значений TOS в анонсах OSPF LSA описано в параграфе 12.3.

**E**

Используется для обратной совместимости с [Ref9].

#### TOS metric

Связанная с TOS метрика.

#### Forwarding address

Используется для обратной совместимости с [Ref9].

#### External Route Tag

Для обратной совместимости с [Ref9].

## B. Архитектурные константы

Некоторые параметры протокола OSPF имеют фиксированные значения (архитектурные константы). Предпочтительно для указания таких параметров в тексте использовать их имена (например, LSRefreshTime). Для настраиваемых параметров протокола также используются имена, описанные в Приложении C.

Ниже перечислены имена архитектурных констант, их значения и краткие комментарии.

#### LSRefreshTime

Максимальный интервал между двумя генерациями отдельного анонса LSA. Если поле возраста одного из порожденных маршрутизатором анонсов LSA достигает значения LSRefreshTime, генерируется новый экземпляр LSA, даже при отсутствии изменений в содержимом LSA (за исключением заголовка LSA). Значение LSRefreshTime составляет 30 минут.

#### MinLSInterval

Минимальный интервал между двумя генерациями отдельного анонса LSA. Значение MinLSInterval составляет 5 секунд.

#### MinLSArrival

Для любого анонса LSA этот параметр задает минимальное время до восприятия нового анонса LSA в процессе лавинной рассылки. Экземпляры LSA, полученные раньше, отбрасываются. Для MinLSArrival установлено значение 1 секунда.

#### MaxAge

Максимальный возраст LSA. Если LS age = MaxAge, анонс LSA повторно рассылается в лавинном режиме в целях удаления из домена маршрутизации (см. главу 14). LSA с возрастом MaxAge не используются при расчете таблицы маршрутов. Используйте для MaxAge значение 1 час.

#### CheckAge

Если возраст LSA в базе данных многократно запрашивается в течение периода CheckAge, нужно проверить контрольную сумму LSA. Расхождение в контрольной сумме говорит о серьезной ошибке. Значение CheckAge составляет 5 минут.

#### MaxAgeDiff

Максимальная дисперсия (расхождение) во времени лавинной рассылки LSA в автономной области. Большую часть этого времени занимает пребывание LSA в выходных очередях маршрутизаторов (без старения) в процессе лавинной рассылки. Значение MaxAgeDiff составляет 15 минут.

#### LSInfinity

Значение метрики, используемое для недоступных адресатов в анонсах LSA. Это значение используется для summary-LSA и AS-external-LSA как альтернатива принудительному старению (см. параграф 14.1). Значение LSInfinity составляет 0xfffff.

#### DefaultDestination

Значение Destination ID, указывающее принятый по умолчанию маршрут (используется при отсутствии других путей к адресату). Принятые по умолчанию маршруты анонсируются только в AS-external-LSA и summary-LSA типа 3 тупиковых областей. Для маршрута по умолчанию IP имеет значение 0.0.0.0 и связанное с ним поле Network Mask также содержит 0.0.0.0.

#### InitialSequenceNumber

Значение, используемое в качестве порядкового номера (LS Sequence Number) при создании первого анонса LSA - 0x80000001.

#### MaxSequenceNumber

Максимальное значение порядкового номера LSA (LS Sequence Number) - 0x7ffffff (целое число со знаком).

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								
LS age															Options										5														
Link State ID																																							
Advertising Router																																							
LS sequence number																																							
LS checksum															length																								
Network Mask																																							
E	0									metric																													
Forwarding address																																							
External Route Tag																																							
E	TOS									TOS metric																													
Forwarding address																																							
External Route Tag																																							

## С. Настраиваемые константы

Протокол OSPF имеет незначительное число конфигурационных параметров. Все эти параметры описаны ниже. Параметры группируются в несколько функциональных категорий (параметры области, интерфейса и т. п.). Для всех параметров приведены примеры значений.

Установка некоторых параметров должна быть согласованной для группы маршрутизаторов. Например, все маршрутизаторы области должны быть согласны с параметрами области, а все маршрутизаторы, подключенные к одной сети, должны использовать адрес и маску, выделенные для этой сети.

Некоторые параметры могут определяться с использованием алгоритмов, выходящих за пределы данной спецификации (например, адрес хоста, подключенного к маршрутизатору по каналу SLIP). С точки зрения OSPF такие параметры остаются конфигурационными.

### С.1 Глобальные параметры

В общем случае каждая область OSPF использует свою копию протокола. По этой причине большинство параметров OSPF задается на уровне области. Ниже перечислены глобальные параметры протокола.

#### **Router ID**

32-битовое целое число, являющееся уникальным идентификатором маршрутизатора в пределах AS. Одним из алгоритмов выделения значений Router ID является присваивание маршрутизатору наибольшего или наименьшего из свободных значений. Если значение OSPF Router ID изменяется программы OSPF в маршрутизаторе должны быть перезагружены для того, чтобы вступило в силу новое значение Router ID. До перезагрузки с целью замены Router ID маршрутизатор должен удалить порожденные им LSA из домена маршрутизации (см. параграф 14.1), поскольку иначе они будут сохраняться еще MaxAge минут.

#### **RFC1583Compatibility**

Управляет правилами выбора предпочтений (параграф 16.4) при наличии множества анонсов AS-external-LSA для одного адресата. Значение enabled задает использование предпочтений в соответствии с RFC 1583 ([Ref9]), а для отключенной совместимости (disabled) используются правила, описанные в параграфе 16.4.1, которые предотвращают возникновение петель, когда AS-external-LSA для одного получателя происходят из разных областей. По умолчанию установлен режим enabled.

Для снижения вероятности возникновения петель все маршрутизаторы в домене OSPF должны иметь одинаковое значение параметра RFC1583Compatibility. Если в домене присутствуют маршрутизаторы, не поддерживающие функций, описанных в параграфе 16.4.1, должно использоваться значение RFC1583Compatibility = enabled. В остальных случаях следует устанавливать значение disabled для предотвращения петель.

### С.2 Параметры области

Все относящиеся к области маршрутизаторы должны соглашаться с конфигурацией области. Рассогласование двух маршрутизаторов ведет к невозможности организации между ними смежности и возникновению препятствий в передаче служебного трафика и пакетов данных. Для области должны настраиваться следующие элементы:

#### **Area ID**

32-битовый идентификатор области, записываемый обычно по аналогии с IP-адресами. Значение 0.0.0.0 зарезервировано для магистрали AS. Если область представляет сеть, разбитую на подсети, в качестве Area ID может служить IP-номер сети.

#### **List of address ranges**

Область OSPF определяется как список адресных диапазонов вида:

[IP address, mask]

Диапазон описывает набор включенных в него адресов IP. Сети и хосты связываются с областью в зависимости от попадания их адресов в один из диапазонов, определяющих область. Маршрутизаторы относятся к нескольким областям в зависимости от принадлежности подключенных к маршрутизатору сетей.

#### **Status**

Имеет значение Advertise или DoNotAdvertise. Маршрутная информация рассматривается на границе области. За пределы области анонсируется по крайней мере один маршрут (в summary-LSA) для каждого диапазона адресов. Маршрут анонсируется тогда и только тогда, когда Status = Advertise. Неанонсируемые диапазоны позволяют иметь сети, скрытые от других областей. По умолчанию Status = Advertise.

В качестве примера предположим, что поделенная на подсети сеть IP является областью OSPF. Область указана как диапазон адресов, в котором выделены адреса подсетей IP с естественными масками класса A, B или C. За пределы области может анонсироваться один маршрут, описывающий всю сеть (вместе с подсетями).

#### **ExternalRoutingCapability**

Определяет рассылку AS-external-LSA в данную область или через нее. Если рассылка AS-external-LSA исключена для области, такая область считается тупиковой (stub). В тупиковых областях маршрутизация ко внешним адресатам базируется только на принятом по умолчанию маршруте. Магистральная область не может быть тупиковой. Кроме того, через тупиковые области нельзя организовать виртуальные соединения. Дополнительные данные о тупиковых областях приведены в параграфе 3.6.

#### **StubDefaultCost**

Если область является тупиковой, а маршрутизатор служит граничным для области, поле StubDefaultCost показывает стоимость для принятого по умолчанию анонса summary-LSA для данной области.

### С.3 Параметры интерфейса маршрутизатора

Некоторые из конфигурационных параметров интерфейса маршрутизатора (например, IP-адрес и маска) на самом деле являются свойствами подключенной сети и, следовательно, должны быть одинаковыми для всех подключенных к сети маршрутизаторов. К числу параметров интерфейса маршрутизатора относятся:

#### **IP interface address**

IP-адрес интерфейса, являющийся уникальным идентификатором интерфейса в масштабе internet. Использование адресов IP не требуется для сетей point-to-point, которые по этой причине называют безадресными (unnumbered).

#### **IP interface mask**

Маска интерфейса (называется также маской сети/подсети) показывающая число битов в сетевой части IP-адреса подключенной сети. Использование маски для IP-адреса интерфейса позволяет определить IP-номер для подключенной сети. В сетях point-to-point и на виртуальных каналах маска не задается. В таких случаях канал, как таковой, не имеет связанного с ним IP-номера, а адреса IP на каждой стороне выделяются независимо или не используются совсем.

#### **Area ID**

Область OSPF, к которой относится подключенная сеть.

#### **Interface output cost**

Стоимость передачи пакета через интерфейс, выраженная в единицах метрики. Это значение анонсируется как стоимость канала в генерируемых маршрутизатором router-LSA. Стоимость всегда должна быть больше 0.

#### ***RxmtInterval***

Число секунд между повторами передачи LSA для смежных маршрутизаторов, связанных с данным интерфейсом. Это значение используется также при повторе пакетов DD и LSR. Период повтора должен превышать время кругового обхода (round-trip delay) между двумя маршрутизаторами через подключенную сеть. При установке слишком малого значения будут возникать ненужные повторы. Для локальных сетей разумно использовать значение 5 секунд.

#### ***InfTransDelay***

Оценка времени (в секундах) передачи пакета LSU через интерфейс. Возраст LSA, содержащихся в пакете обновления, должен увеличиваться на величину этого параметра до их передачи через интерфейс. Это значение должно приниматься во внимание при определении для интерфейса времени задержки при передаче и распространении. Значение этого параметра должно быть больше 0. Для локальных сетей рекомендуется устанавливать значение 1 сек.

#### ***Router Priority***

Беззнаковое 8-битовое число, используемое для выбора маршрутизатора на роль DR. Если два маршрутизатора выдвигают на эту роль себя, из них выбирается тот, у которого больше значение Router Priority. Если значения приоритета совпадают, выбирается маршрутизатор с большим значением Router ID. Маршрутизаторы с Router Priority = 0 не могут быть DR для подключенной сети. Параметр Router Priority задается только для интерфейсов в широковещательные и NBMA-сети.

#### ***HelloInterval***

Период (в секундах) между передачей двух последовательных пакетов Hello через один интерфейс. Это значение анонсируется маршрутизатором в пакетах Hello и должно совпадать для всех подключенных к общей сети маршрутизаторов. Снижение HelloInterval ускоряет обнаружение топологических изменений, но увеличивает объем служебного трафика OSPF. Для сетей X.25 PDN рекомендуется значение 30 сек, для ЛВС - 10 сек.

#### ***RouterDeadInterval***

Задаёт время (в секундах) между прекращением приема пакетов Hello от маршрутизатора и констатацией его неработоспособности. Это время также анонсируется в пакетах Hello (поле RouterDeadInterval). Значение этого параметра должно в несколько раз (скажем, в 4) превышать значение HelloInterval. Во всех маршрутизаторах, подключенных к общей сети, этот параметр должен совпадать.

#### ***AuType***

Определяет процедуру аутентификации, используемую в подключенной сети. Это поле должно быть одинаковым для всех подключенных к одной сети маршрутизаторов. Описания различных типов аутентификации даны в Приложении D.

#### ***Authentication key***

Это поле данных позволяет проверить аутентичность пакетов протокола OSPF, принятых через интерфейс.

Например, если AuType указывает простой пароль, поле Authentication key будет содержать 64-битовую строку пароля в явном виде. Значение Authentication key для других типов аутентификации OSPF рассмотрено в Приложении D.

## **C.4 Параметры виртуального канала**

Виртуальные каналы используются для организации соединений с магистралью или увеличения числа таких соединений. Виртуальный канал можно организовать между любой парой граничных маршрутизаторов разных областей, если они имеют интерфейсы в общую область за пределами магистральной. Виртуальный канал появляется на графе магистральной как безадресное соединение point-to-point. Виртуальный канал должен быть настроен для обоих маршрутизаторов.

Виртуальный канал появляется в router-LSA (для магистральной) как отдельный интерфейс маршрутизатора в магистральную область. В связи с этим, он имеет все присущие интерфейсу параметры (см. параграф C.3). Хотя виртуальный канал работает подобно безадресному соединению «точка-точка», с ним всегда связан IP-адрес интерфейса. Этот адрес используется как адрес отправителя в пакетах OSPF, передаваемых через виртуальный канал, и устанавливается динамически в процессе построения таблицы маршрутизации. Выходная стоимость интерфейса также задается динамически и является стоимостью внутридоменного пути между двумя маршрутизаторами. Должен быть задан параметр RxmtInterval и требуется точная оценка времени кругового обхода (round-trip delay) между двумя маршрутизаторами. Такая оценка для виртуального канала может оказаться непростой, поэтому лучше поставить достаточно большое значение. Параметр Router Priority для виртуальных каналов не используется.

Виртуальный канал определяется двумя конфигурационными параметрами – значением Router ID для маршрутизатора на другой стороне канала и областью (не магистралью), через которую организован виртуальный канал (транзитная область виртуального соединения). Виртуальные каналы нельзя организовать через тупиковые области.

## **C.5 Параметры сетей NBMA**

Протокол OSPF трактует сети NBMA аналогично широковещательным сетям. Поскольку к сети может быть подключено множество маршрутизаторов, среди них нужно выбрать маршрутизатор DR для сети, который будет генерировать network-LSA со списком всех маршрутизаторов, подключенных к NBMA-сети.

Однако, в силу ограниченных возможностей широковещания, может потребоваться дополнительная настройка для выбора DR. Конфигурационные параметры задаются только в тех маршрутизаторах, которые могут претендовать на роль DR (Router Priority > 0), при отсутствии процедуры автоматического обнаружения соседей. В число таких параметров входят:

#### ***List of all other attached routers***

Список всех остальных маршрутизаторов, подключенных к сети NBMA с указанием IP-адреса интерфейса в сеть. Для каждого маршрутизатора в списке также должна быть указана возможность использования в качестве DR. При активизации интерфейса в сеть NBMA маршрутизатор передает пакеты Hello только тем соседям, которые могут играть роль DR, пока не будет обнаружен текущий маршрутизатор DR.

#### ***PollInterval***

Если соседний маршрутизатор теряет активность (нет пакетов Hello в течение RouterDeadInterval сек.), может потребоваться передача пакетов Hello «мертвому» соседу. Эти пакеты будут передаваться с интервалом PollInterval, значительно превышающим HelloInterval. Разумным значением PollInterval для сетей PDN X.25 будет 2 мин.

## **C.6 Параметры сетей Point-to-MultiPoint**

В сетях Point-to-MultiPoint может потребоваться указание набора соседей, доступных напрямую через сеть Point-to-MultiPoint. Каждый сосед идентифицируется IP-адресом в сети Point-to-MultiPoint. Маршрутизаторы DR не задаются для сетей Point-to-MultiPoint, поэтому поле Designated Router eligibility не имеет смысла.

Соседей в сети Point-to-MultiPoint можно определять и с помощью протоколов нижних уровней (например, Inverse ARP - [Ref14]).



## C.7 Параметры Host route

Маршруты к хостам анонсируются в router-LSA как маршруты в тупиковые сети с маской 0xffffffff. Такая маска говорит о подключении к сети point-to-point, закольцовывании интерфейса маршрутизатора или хосте IP, подключенном непосредственно к маршрутизатору (например, по каналу SLIP). Для каждого подключенного напрямую хоста должны задаваться следующие поля:

### Host IP address

IP-адрес хоста.

### Cost of link to host

Стоимость передачи пакета хосту в единицах метрики link state. Поскольку хосты в большинстве случаев имеют единственное подключение к Internet, на практике эта стоимость часто не имеет значения (т. е., не влияет на маршрутизацию).

### Area ID

Область OSPF, к которой относится хост.

## D. Аутентификация

Весь обмен данными протокола OSPF использует аутентификацию. Заголовок пакета OSPF (см. параграф A.3.1) включает поле типа аутентификации и 64-битовое поле данных, используемое соответствующей типу схемой аутентификации.

Тип аутентификации задается для каждого интерфейса (или, что эквивалентно, для сети/подсети). Дополнительные данные аутентификации также задаются для каждого интерфейса. В данной спецификации определены три типа аутентификации - 0, 1 и 2. Все остальные значения типа аутентификации зарезервированы для распределения IANA ([iana@ISI.EDU](mailto:iana@ISI.EDU)). Список текущих значений приведен в таблице 23.

Таблица 23 Типы аутентификации OSPF.

AuType	Описание
0	Без аутентификации (Null authentication)
1	Простой пароль
2	Криптографическая аутентификация
Прочие	Зарезервированы для распределения IANA ( <a href="mailto:iana@ISI.EDU">iana@ISI.EDU</a> )

### D.1 Null-аутентификация

Использование этого типа означает передачу пакетов через сеть/подсеть без аутентификации. 64-битовое поле аутентификации в заголовке OSPF не содержит информации и не проверяется на приемной стороне. При использовании Null-аутентификации все содержимое каждого пакета OSPF (за исключением 64-битового поля аутентификации) используется для расчета контрольной суммы, позволяющей детектировать ошибки.

### D.2 Парольная аутентификация

При использовании этого типа аутентификации 64-битовое поле задается в масштабе отдельной сети. Все пакеты, передаваемые в эту сеть, должны иметь соответствующее значение в 64-битовом поле данных аутентификации заголовка OSPF. Значение этого поля представляет собой 64-битовый пароль в незашифрованном виде. В дополнение этому все содержимое каждого пакета OSPF (за исключением 64-битового поля аутентификации) используется при расчете контрольной суммы.

Простая парольная аутентификация предотвращает неанонсированное присоединение маршрутизатора к домену маршрутизации – каждый маршрутизатор должен сначала предъявить установленный для сети пароль и только после проверки пароля возможно включение в домен маршрутизации. Однако простая парольная аутентификация уязвима для пассивных атак, часто встречающихся в Internet (см. [Ref16]). Любой человек, имеющий физический доступ в сеть, может узнать пароль и обойти систему безопасности домена маршрутизации OSPF.

### D.3 Криптографическая аутентификация

При использовании криптографической аутентификации задается публичный ключ (shared secret key) для всех маршрутизаторов, подключенных к общей сети/подсети. Для каждого пакета OSPF используется ключ при генерации и проверке цифровых подписей (message digest), добавляемых в конце пакета OSPF. Цифровая подпись создается с помощью необратимой (one-way) функции на основе пакета OSPF и секретного ключа. Поскольку секретный ключ никогда не передается через сеть в открытом виде, это обеспечивает защиту против пассивных атак.

Алгоритм, используемый при генерации и проверке цифровой подписи, определяется секретным ключом. Данная спецификация полностью определяет применение криптографической аутентификации OSPF при использовании алгоритма MD5.

В дополнение к цифровой подписи используется неубывающий порядковый номер, включаемый в каждый пакет OSPF для защиты от replay-атак (подмена). Это обеспечивает высокий уровень защиты, однако сохраняется возможность подмены пакета OSPF пока изменяется порядковый номер. Для предотвращения такой возможности структура данных для каждого соседа содержит новое поле «криптографический порядковый номер» (cryptographic sequence number).

Это поле инициализируется нулевым значением и снова устанавливается в 0 при каждом переходе соседа в состояние Down. Всякий раз, когда подтверждается аутентичность пакета OSPF, в это поле помещается порядковый номер принятого пакета.

Данная спецификация не предлагает процедуры обновления порядкового номера после использования всего диапазона значений. Когда криптографический порядковый номер на передающем маршрутизаторе достигает максимального значения, маршрутизатор должен снова установить нулевое значение для порядкового номера. После такой операции соседний маршрутизатор будет отвергать пакеты OSPF в течение интервала RouterDeadInterval, а потом маршрутизаторы должны будут заново организовать отношения смежности через этот интерфейс. Однако предполагается, что во многих реализациях в качестве криптографического порядкового номера будет использоваться «число секунд с момента перезагрузки» (seconds since reboot) или аналогичное значение (например, seconds since 1960). Такой выбор будет в значительной мере решать проблему, поскольку для порядкового номера используется 32-битовое поле.

Криптографическая аутентификация OSPF не обеспечивает конфиденциальности.

При использовании криптографической аутентификации 64-битовое поле Authentication в стандартных заголовках пакетов OSPF используется несколько иначе (см. таблицу 24).

#### Key ID

Это поле идентифицирует алгоритм и секретный ключ, используемые для создания цифровой подписи, добавляемой к пакетам OSPF. Идентификаторы Key Id являются уникальными и задаются для каждого интерфейса (или, эквивалентно, для подсети).

#### Auth Data Len

Размер (в байтах) цифровой подписи (message digest), добавляемой к пакету OSPF.

#### Cryptographic sequence number

Беззнаковый 32-битовый порядковый номер, который может только увеличиваться и служит для предотвращения replay-атак.

Таблица 24 Использование поля Authentication в заголовке пакетов OSPF с криптографической аутентификацией.

0011																													
--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--	--

Добавляемая к пакету OSPF цифровая подпись не рассматривается как часть пакета OSPF и не учитывается в поле размера в заголовке OSPF, хотя и учитывается в общем размере пакета IP.

Каждый ключ задается комбинацией интерфейса и поля Key ID. Интерфейс может иметь несколько активных ключей одновременно, что обеспечивает возможность простой и безопасной смены ключа. С каждым ключом связано 4 временных параметра, в которых может использоваться время суток (time-of-day) или время по локальным часам маршрутизатора (например, число секунд с момента перезагрузки):

#### **KeyStartAccept**

Время, когда маршрутизатор начнет принимать пакеты, созданные с этим ключом.

#### **KeyStartGenerate**

Время, когда маршрутизатор начнет использование ключа для генерации пакетов.

#### **KeyStopGenerate**

Время, когда маршрутизатор прекратит использование ключа для генерации пакетов.

#### **KeyStopAccept**

Время, когда маршрутизатор прекратит принимать пакеты, созданные с этим ключом.

Для упрощения смены ключей значение KeyStartAccept должно быть меньше KeyStartGenerate, а KeyStopGenerate меньше, чем KeyStopAccept. Если значения KeyStopGenerate и KeyStopAccept не заданы, время жизни ключа становится бесконечным. Когда новый ключ приходит на смену прежнему, время KeyStartGenerate для нового ключа должно быть не больше времени KeyStopGenerate для старого ключа.

Для предотвращения проблем ключи должны сохраняться при перезагрузке маршрутизатора и аппаратном сбросе. В тех случаях, когда истекает время действия последнего ключа, связанного с интерфейсом, недопустим переход в состояние работы без аутентификации и нарушение маршрутизации. Следовательно, маршрутизатор должен передавать уведомление last authentication key expiration (срок действия последнего ключа заканчивается) и трактовать время жизни ключа как бесконечное, пока это время не будет увеличено и ключ не будет удален системой сетевого управления или не будет порожден новый ключ.

## **D.4 Генерация сообщений**

После подготовки содержимого пакета OSPF вызывается процедура аутентификации, заданная для передающего интерфейса полем Autype. Процедура аутентификации вносит в пакет OSPF требуемые изменения.

### **D.4.1 Генерация Null-аутентификации**

При использовании Null-аутентификации пакет изменяется следующим образом:

- (1) Для поля Autype в стандартном заголовке OSPF устанавливается значение 0.
- (2) Поле контрольной суммы в стандартном заголовке OSPF содержит стандартную контрольную сумму IP всего содержимого пакета, начиная с заголовка OSPF, но без учета 64-битового поля аутентификации. Если длина пакета не равна целому числу 16-битовых слов, он дополняется справа соответствующим количеством нулей.

### **D.4.2 Генерация простой парольной аутентификации**

При использовании простого пароля пакет изменяется следующим образом:

- (1) Для поля Autype в стандартном заголовке OSPF устанавливается значение 1.
- (2) Поле контрольной суммы в стандартном заголовке OSPF содержит стандартную контрольную сумму IP всего содержимого пакета, начиная с заголовка OSPF, но без учета 64-битового поля аутентификации. Если длина пакета не равна целому числу 16-битовых слов, он дополняется справа соответствующим количеством нулей.
- (3) 64-битовое поле аутентификации в стандартном заголовке пакета OSPF содержит 64-битовый пароль (ключ аутентификации), заданный для интерфейса.

### **D.4.3 Генерация криптографической аутентификации**

При использовании криптографической аутентификации для интерфейса может быть указано несколько ключей. В таких случаях из числа ключей, корректных для генерации сообщений ( $\text{KeyStartGenerate} \leq \text{текущее время} < \text{KeyStopGenerate}$ ) выбирается ключ с самым последним значением KeyStartGenerate. Пакеты модифицируются с использованием этого ключа:

- (1) Для поля Autype в стандартном заголовке OSPF устанавливается значение 2.
- (2) Контрольная сумма не рассчитывается и соответствующее поле заголовка OSPF имеет нулевое значение.
- (3) В поле Key ID (см. таблицу 24) устанавливается значение Key ID выбранного ключа.
- (4) В поле Auth Data Len указывается размер (в байтах) цифровой подписи, которая будет добавляться к пакету OSPF. При использовании алгоритма MD5 поле Auth Data Len имеет значение 16.
- (5) В 32-битовое поле криптографического порядкового номера (см. таблицу 24) помещается неубывающее значение (т. е., значение, по крайней мере равное последнему значению, переданному с использованием данного интерфейса). Выбор значений криптографического порядкового номера зависит от реализации (это может быть простой счетчик или значение, базирующееся на показаниях системных часов).
- (6) Рассчитывается цифровая подпись и добавляется к пакету OSPF. Используемый для расчета алгоритм аутентификации указывается ключом. Исходными данными для расчета является содержимое пакета OSPF и секретный ключ. При использовании алгоритма MD5 цифровая подпись рассчитывается следующим образом:
  - a) К пакету OSPF добавляется 16-байтовый ключ MD5.
  - b) Добавляются поля заполнения в соответствии с [Ref17].
  - c) Алгоритм MD5 используется для конкатенации пакета OSPF, секретного ключа, заполнения и поля длины, давая в результате 16-байтовую цифровую подпись (см. [Ref17]).
  - d) Подпись MD5 записывается поверх ключа OSPF (т. е., добавляется в конце исходного пакета OSPF). Подпись не учитывается в поле длины в заголовке OSPF, но включается в длину пакета IP. Любые добавления или заполнение после подписи не учитываются и не передаются.

## **D.5 Верификация сообщений**

Когда принимается пакет OSPF, должна быть проверена его аутентичность. Процедура унификации задается полем Autype в стандартном заголовке OSPF; соответствующее значение Autype должно быть задано для приемного интерфейса OSPF.

Если аутентичность принятого пакета OSPF подтверждена, продолжается обработка пакета, как описано в параграфе 8.2. Пакеты, не прошедшие аутентификацию, отбрасываются.

### **D.5.1 Проверка Null-аутентификации**

При использовании Null-аутентификации должно проверяться поле контрольной суммы в заголовке OSPF, содержащее стандартную контрольную сумму IP. Если длина пакета не кратна целому числу 16-битовых слов, пакет дополняется справа нулями.

### D.5.2 Проверка простой парольной аутентификации

При использовании простой парольной аутентификации проверка пакетов OSPF происходит следующим образом:

- (1) Должно проверяться поле контрольной суммы в заголовке OSPF (стандартная контрольная сумма IP). Если размер пакета не кратен 16 битам, он дополняется нулями справа до вычисления контрольной суммы.
- (2) 64-битовое поле аутентификации в заголовке пакета OSPF должно содержать 64-битовый пароль (ключ аутентификации), заданный для интерфейса.

### D.5.3 Проверка криптографической аутентификации

При использовании криптографической аутентификации проверка принятых пакетов OSPF осуществляется следующим образом:

- (1) Находится ключ, заданный для приемного интерфейса, имеющий значение Key ID, равное указанному в принятом пакете OSPF (см. таблицу 24). Если ключ не найден или некорректен (т. е.,  $\text{current time} < \text{KeyStartAccept}$  или  $\text{current time} \geq \text{KeyStopAccept}$ ), пакет OSPF отбрасывается.
- (2) Если криптографический порядковый номер, найденный в заголовке OSPF header (см. таблицу 24), меньше криптографического порядкового номера, записанного в структуре данных о соседе, пакет OSPF отбрасывается.
- (3) Проверяется цифровая подпись, добавленная к пакету:
  - (a) Полученная подпись сохраняется отдельно.
  - (b) Рассчитывается новая подпись в соответствии с п. 6 параграфа D.4.3.
  - (c) Рассчитанное и полученное значения сравниваются. При несоответствии пакет OSPF отбрасывается. Если подписи совпадают, пакет считается аутентичным и в поле cryptographic sequence number структуры данных о соседе помещается порядковый номер из заголовка пакета OSPF.

## E. Алгоритм установки Link State ID

Поле Link State ID в записях AS-external-LSA и summary-LSA обычно содержит IP-адрес описываемой сети. Однако при необходимости это поле может включать также биты адреса хоста. Это позволяет маршрутизатору разделять LSA для сетей с одинаковым номером, но разными масками, которые могут встречаться при использовании сверхсетей и «нулевых» подсетей (см. [Ref10]).

В этом приложении рассматривается один из возможных алгоритмов установки битов адреса хоста в поле Link State ID. Выбор подходящего алгоритма не задается спецификацией и определяется локально. Разные маршрутизаторы могут использовать различные алгоритмы, поскольку этот выбор влияет только на LSA, генерируемые данным маршрутизатором. Единственным требованием к алгоритму является использование по возможности IP-номера сети в качестве значения поля Link State ID, поскольку это повышает уровень интероперабельности с реализациями OSPF, предшествующими RFC 1583.

Описанный ниже алгоритм описан на примере AS-external-LSA, но может использоваться и для summary-LSA. Предположим, что маршрутизатор хочет сгенерировать анонс AS-external-LSA для сети, имеющей адрес NA и маску NM1. Для определения Link State ID выполняются следующие операции:

- (1) Выясняется генерировал ли маршрутизатор ранее AS-external-LSA с Link State ID = NA (в таких LSA маршрутизатор будет указываться как Advertising Router). Если этого не было, устанавливается Link State ID = NA и работа алгоритма завершается.
- (2) Определяется маска сети из тела уже существующих AS-external-LSA (обозначим эту маску NM2). Возможны 2 случая:
  - o NM1 больше, чем NM2 (сетевая часть адреса содержит больше битов). В этом случае устанавливаем для поля Link State ID в новом анонсе LSA значение [NA,NM1] со всеми дополнительными битами из адреса хоста (т. е., NA OR NOT NM1) или объединение номера сети с ширококестельным адресом сети [NA,NM1]).
  - o NM2 больше по сравнению с NM1. В этом случае изменяем существующий анонс LSA (Link State ID = NA), чтобы он указывал на новую сеть [NA,NM1], увеличиваем порядковый номер, а также изменяем маску в теле анонса на NM1 и вставляем значение стоимости для новой сети. После этого генерируется новый анонс LSA для старой сети [NA,NM2] с Link State ID = NA, плюс биты, которые отсутствуют в маске NM2 (ширококестельный адрес [NA,NM2]).

Описанный выше алгоритм предполагает, что все маски являются непрерывными – для двух сетей с одним номером в таких случаях одна маска будет более специфична<sup>42</sup>, нежели другая. Предполагается также, что не существует сетей, имеющих адрес, который совпадает с ширококестельным адресом другой сети. С учетом этих предположений описанный выше алгоритм всегда обеспечивает уникальное значение Link State ID. Этот алгоритм можно описать другими словами. При генерации AS-external-LSA пытаемся использовать номер сети в качестве Link State ID. При возникновении конфликта проверяются конфликтующие сети, одна из которых будет подмножеством другой. Для более широкой сети используем в качестве Link State ID номер сети, а для другой сети используем ширококестельный адрес (т. е., устанавливаем 1 для всех битов «адреса хоста»). Если более узкая сеть обрабатывается раньше, придется генерировать два анонса LSA сразу.

В качестве примера использования алгоритма рассмотрим его работу при описанной ниже последовательности событий в одном маршрутизаторе (Router A).

- (1) Router A хочет сгенерировать анонс AS-external-LSA для сети [10.0.0.0,255.255.255.0]:
  - a. Используется Link State ID = 10.0.0.0.
- (2) Router хочет породить AS-external-LSA для сети [10.0.0.0,255.255.0.0]:
  - a. LSA для [10.0.0.0,255.255.255.0] генерируется заново с Link State ID = 10.0.0.255.
  - b. Для сети [10.0.0.0,255.255.0.0] используется Link State ID = 10.0.0.0.
- (3) После этого маршрутизатор Router A хочет сгенерировать AS-external-LSA для [10.0.0.0,255.0.0.0]:
  - a. Заново порождается LSA для сети [10.0.0.0,255.255.0.0] с новым значением Link State ID = 10.0.255.255.
  - b. Для сети [10.0.0.0,255.0.0.0] используется Link State ID = 10.0.0.0.
  - c. Для сети [10.0.0.0,255.255.255.0] сохраняется Link State ID = 10.0.0.255.

## F. Множество интерфейсов в одну сеть/подсеть

Существует по крайней мере два способа поддержки множества физических интерфейсов в одну подсеть IP. Оба метода будут интероперабельны с RFC 1583 (и, естественно, с данной спецификацией). Ниже кратко описаны два способа в предположении что каждому интерфейсу выделен свой IP-адрес (в противном случае вопрос поддержки множества интерфейсов переносится скорее на канальный уровень или уровень ARP).

### Метод 1:

Все функции OSPF поддерживаются на обоих интерфейсах (прием и передача приветствий, лавинная рассылка, поддержка раздельных FSM для интерфейса и соседа и т. п.). Все остальные маршрутизаторы подсети будут трактовать эти интерфейсы как разных соседей, идентифицируемых (в ширококестельных и NBMA-сетях) по их IP-адресам.

Метод 1 имеет следующие недостатки:

- (1) Увеличивается общее число соседей и смежных маршрутизаторов.

<sup>42</sup> Меньше битов в адресе хоста. Прим. перев.



- (2) Теряется возможность проверки двухсторонней связи на обоих интерфейсах поскольку двунаправленность обеспечивается на основе Router ID.
- (3) При выборе DR оба интерфейса рассматриваются вместе, поскольку выбор обоих в качестве DR может привести к неоднозначности (к кому относится Router ID).

## Метод 2:

Протокол OSPF работает только на одном интерфейсе (назовем его основным или первичным), но в Router-LSA включаются оба интерфейса.

Метод 2 также имеет недостатки:

- (1) Теряется проверка двунаправленности для вторичного интерфейса.
- (2) При отказе основного интерфейса требуется перевести второй интерфейс в состояние основного.

## G. Отличия от RFC 2178

В этом приложении рассмотрены различия между данной спецификацией и RFC 2178. Все различия являются обратно совместимыми. Реализации данной спецификации и RFC 2178, 1583, 1247 будут интероперабельны.

### G.1 Изменение лавинной рассылки

В процедуру лавинной рассылки (глава 13) внесено три изменения.

1. *Этап 4 параграфа 13.* Анонсы LSA возраста MaxAge подтверждаются и после этого отбрасываются только при выполнении двух условий: а) в базе данных нет копии LSA и б) ни один из соседних маршрутизаторов не находится в состоянии Exchange или Loading. Во всех прочих случаях LSA с возрастом MaxAge обрабатываются подобно остальным LSA – устанавливаются в базу данных и рассылаются в лавинном режиме через подходящий интерфейс, когда копия в базе данных более старая (этап 5 в параграфе 13). Эти изменения влияют на содержимое таблицы 22.
2. *Этап 5а в параграфе 13.* Проверка MinLSArrival выполняется только для LSA, полученных в процессе лавинной рассылки и не должна применяться к собственным LSA маршрутизатора.
3. *Этап 8 в параграфе 13.* Конфликт между маршрутизаторами при определении более свежего экземпляра LSA может приводить к катастрофической лавинной рассылке для протоколов link-state [Ref26]. OSPF решает эту проблему двумя способами: а) возраст LSA используется при лавинной рассылке подобно TTL, что позволяет устранять «петли LSA» в сети (см. параграф 13.3) и б) маршрутизаторы не воспринимают обновлений LSA с периодом меньше MinLSArrival секунд (см. параграф 13). Однако сохраняется одна ситуация в RFC 2178, когда несогласие в определении более свежего экземпляра LSA может порождать излишнюю лавинную рассылку - отклик на старые LSA путем повтора лавинной рассылки копии из базы данных. По этой причине этап 8 в параграфе 13 был исправлен так, чтобы копия из базы данных рассылалась только в тех случаях, когда эта копия не передавалась ни в одном пакете LSU в течение последних MinLSArrival секунд.

### G.2 Смена предпочтений для внешнего пути

Сохраняется возможность возникновения маршрутных петель в RFC 2178 при использовании виртуальных каналов, когда один и тот же внешний путь импортируется несколькими ASBR, относящимися к разным областям. Для решения этой проблемы был переработан параграф 16.4.1. При выборе корректного ASBR/адреса пересылки внутридоменные пути мимо магистрали всегда имеют предпочтение. Однако внутридоменные пути через магистраль (область 0) и междоменные пути имеют одинаковый уровень предпочтения и должны сравниваться с учетом их стоимости.

Изменения обусловлены следующими причинами. При использовании виртуальных каналов внутридоменный путь через магистраль для одного маршрутизатора может совпадать с междоменным путем другого маршрутизатора, лежащего на несколько интервалов (hop) ближе к адресату. Следовательно, внутридоменный магистральный путь и междоменный путь должны иметь одинаковый уровень предпочтений. Мы можем сравнивать стоимость таких путей, выбирая из них лучший (см. параграф 16.3).

Автор благодарит Michael Briggs и Jeremy McCooley из UNH InterOperability Lab, указавших на эту проблему.

### G.3 Неполное разрешение виртуальных next hop

Одной из задач расчета, описанного в параграфе 16.3, является определение следующих интервалов (next hop) для тех адресатов, которые имеют значения next hop, рассчитанные как виртуальный канал (см. параграфы 16.1 и 16.2). После завершения расчетов, описанный в параграфе 16.3, все пути, рассчитанные в 16.1 и 16.2, которые содержат необработанные виртуальные next hop, должны отбрасываться.

### G.4 Просмотр таблицы маршрутизации

Алгоритм поиска в таблице маршрутов (параграф 11.1) был изменен на основе реального опыта. Наиболее соответствующая (best match) запись маршрутной таблицы считается та, которая обеспечивает большее совпадение (наименьшая маска). Предположим для примера, что маршрутизатор пересылает пакеты для адреса 192.9.1.1. Запись в таблице для адресата 192.9.1/24 будет обеспечивать лучшее соответствие по сравнению с записью для 192.9/16, независимо от типов путей в таблице. Отметим, однако, что при наличии в маршрутной записи множества путей расчеты, описанные в 16.1, 16.2 и 16.4, всегда дают более предпочтительные пути (в порядке снижения предпочтений – внутридоменные, междоменные и внешние типа 1 и 2 – см. главу 11).

## Вопросы безопасности

Весь обмен данными протокола OSPF использует аутентификацию. Поддерживается несколько типов аутентификации OSPF и выбор типа задается конфигурационными параметрами для каждого сегмента сети. Предполагается, что криптографическая аутентификация OSPF устойчива к пассивным атакам и обеспечивает достаточно надежную защиту от активных атак. При использовании криптографической аутентификации каждый маршрутизатор добавляет (append) цифровую подпись (message digest) к передаваемому пакету OSPF. На приемной стороне для проверки аутентичности пакетов используется открытый ключ и цифровая подпись из принятого пакета OSPF.

Уровень безопасности при криптографической аутентификации полностью определяется используемым алгоритмом создания цифровой подписи (в настоящее время спецификация включает только алгоритм MD5), качеством используемых ключей и корректностью реализации механизма обеспечения безопасности во всех используемых реализациях OSPF. Требуется также обеспечение безопасности хранения секретных ключей.

Ни один из типов аутентификации OSPF не обеспечивает конфиденциальности и не предотвращает возможность анализа трафика. Управление ключами также не рассматривается в данной спецификации.

Дополнительную информацию можно найти в параграфах 8.1, 8.2 и Приложении D.

## Адреса авторов

John Moy



Ascend Communications, Inc.  
1 Robbins Road  
Westford, MA 01886

Phone: 978-952-1367  
Fax: 978-392-2075  
EMail: [jmoy@casc.com](mailto:jmoy@casc.com)

**Перевод на русский язык**

Николай Малых  
[nmalykh@bilim.com](mailto:nmalykh@bilim.com)

***Полное заявление авторских прав***

**Copyright (C) The Internet Society (1998). All Rights Reserved.**

This document and translations of it may be copied and furnished to others, and derivative works that comment on or otherwise explain it or assist in its implementation may be prepared, copied, published and distributed, in whole or in part, without restriction of any kind, provided that the above copyright notice and this paragraph are included on all such copies and derivative works. However, this document itself may not be modified in any way, such as by removing the copyright notice or references to the Internet Society or other Internet organizations, except as needed for the purpose of developing Internet standards in which case the procedures for copyrights defined in the Internet Standards process must be followed, or as required to translate it into languages other than English.

The limited permissions granted above are perpetual and will not be revoked by the Internet Society or its successors or assigns.

This document and the information contained herein is provided on an "AS IS" basis and THE INTERNET SOCIETY AND THE INTERNET ENGINEERING TASK FORCE DISCLAIMS ALL WARRANTIES, EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY WARRANTY THAT THE USE OF THE INFORMATION HEREIN WILL NOT INFRINGE ANY RIGHTS OR ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE.