БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

Факультет прикладной математики и информатики

Конфигурация RIPv2 и ее проверка

Минск 2025

СОДЕРЖАНИЕ

1. Протоколы маршрутизации

1.1. Статическая маршрутизация

Маршрут характеризуется вектором расстояния до места назначения. Предполагается, что каждый маршрутизатор является отправной точкой нескольких маршрутов до сетей, с которыми он связан.

Маршрутизация — это метод, с помощью которого хост или маршрутизатор решает, куда должен быть послан пакет для достижения своего места назначения.

При небольшом количестве подсетей используется **статическая маршрутизация** (см. лабораторная работа №8).

Таблица маршрутизации – это база данных описания маршрутов в составной сети. Статические записи вносятся вручную при конфигурировании маршрутизаторов. Все записи в таблице имеют неизменяемый статический статус, что означает бесконечный срок их жизни. Задача ручного ведения таблиц маршрутизации (база данных описания маршрутов) администратором сети при значительном масштабе сети становится слишком трудоемкой и сопряжена (как показала практика) с большим количеством ошибок, допускаемым при конфигурировании сетевых устройств. При измерении состояния сети администратор обязан *срочно* отразить эти изменения в соответствующих таблицах маршрутизации, иначе это приводит к рассогласованию таблиц на маршрутизаторах. Значит, сеть будет работать некорректно.

Поэтому в настоящее время используются протоколы **адаптивной маршрутизации**. Заметим, что в процессе функционирования сети наблюдается несоответствие маршрутной таблицы реальной ситуации. Это типично не только для RIP, но характерно для всех протоколов, базирующихся на векторе расстояния, где информационные сообщения актуализации несут в себе только пары кодов: адрес места назначение и расстояние до него.

1.2. Динамическая маршрутизация

В общем случае под **алгоритмом маршрутизации** понимается набор правил, регламентирующих процедуры обмена служебной информацией между маршрутизаторами с целью заполнения их таблиц.

Ниже будет рассматриваться так называемая *одношаговая маршрутизация*, при которой маршрутизатор *пересылает* пакет в следующую подсеть и *не заботится* о его дальнейшем продвижении.

В противовес в таблицах маршрутизации при адаптивной маршрутизации имеется информация об интервале времени, в течение которого данный маршрут будет оставаться действительным. Это время называют **временем жизни – TTL маршрута.** Если по истечении времени жизни существование маршрута не подтверждается протоколом маршрутизации, то он считается нерабочим. Пакеты по такому маршруту не посылаются.

Различают следующие два основных вида динамической (адаптивной) маршрутизации: распределенная и централизованная.

1.2.1. Распределенная маршрутизация

При распределенной маршрутизации все маршрутизаторы сети находятся в равных условиях, они находят маршруты и строят собственные таблицы маршрутизации путем обмена служебной информацией друг с другом.

Алгоритмы распределенной маршрутизации можно разделить на **дистанционно-векторные алгоритмы – DVA** и **алгоритмы состояния связей – LSA**.

В **DVA** алгоритмах каждый маршрутизатор *периодически и широковещательно* рассылает по сети вектор расстояний (метрик) от самого себя до известных ему подсетей.

В качестве метрики обычно используется количество промежуточных маршрутизаторов, через которые должен пройти пакет, чтобы достигнуть подсети назначения. Наименьшей метрике соответствует кратчайшее расстояние до сети назначения, маршрут с минимальной метрикой считается оптимальным.

Получив такой вектор от соседа-маршрутизатора, каждый маршрутизатор добавляет свои сведения обо всех известных ему подсетях, и снова рассылает обновленный вектор по сети.

При передаче пакета маршрутизатор выбирает из нескольких альтернативных маршрутов тот маршрут, который имеет наименьшую метку.

Таким образом, в результате такого обмена векторами, каждый маршрутизатор в конце концов получит информацию обо всех подсетях, входящих в составную сеть, а также о расстояниях до них.

Заметим, что дистанционно-векторные алгоритмs- **DVA** хорошо работают только в небольших сетях.

Почему? Периодически засоряют сеть интенсивным графиком, изменения в сети не всегда гарантированно правильно отрабатываются, так как они не работают с информацией о топологии связей в составной сети.

Алгоритмы состояния связей – LSA, напротив, обеспечивают каждый маршрутизатор информацией, достаточной для построения точного графа связей

составной сети. Все маршрутизаторы работают на основании одного и того же графа. Широковещательная рассылка используется здесь только при изменениях состояния связей. А так в обычном режиме маршрутизаторы обмениваются короткими пакетами со своими близкими соседями. То есть, служебный трафик **LSA** менее интенсивен, чем трафик – **DVA**.

1.2.2. Централизованная маршрутизация

При централизованном подходе используется выделенный маршрутизатор, который собирает всю информацию о состоянии сети и о топологии, строит таблицы маршрутизации для всех остальных маршрутизаторов сети, а затем распространяет их по сети. Каждый маршрутизатор, получив свою таблицу, сам самостоятельно принимал решение о продвижении пакета. Пока централизованные алгоритмы маршрутизации не нашли должного применения в современных сетях.

2. Протокол RIP

Протокол RIP является одним из первых внутренних протоколов маршрутизации и относится к дистанционно-векторным протоколам. Существуют две версии **RIP** – первая использует маршрутизацию на основе классов, вторая версия **RIPv2** использует бесклассовую маршрутизацию (позволяет работать с масками подсетей). Кроме того, в дополнение к широковещательному режиму поддерживает мультикастинг.

Протокол RIP не является универсальным протоколом маршрутизации и не может быть использован в IP-сети любого размера и сложности. В частности, протокол накладывает ограничения на **максимальный диаметр сети** (то есть максимальное расстояние, на которое может быть передан пакет, и после превышения, которого пункт назначения считается недостижимым). Для протоколов RIP обеих версий максимальный диаметр сети составляет 15 маршрутизаторов. Поэтому маршрут с метрикой 16 считается недостижимым (бесконечным). Отсюда RIP для больших сетей не годится.

Для сравнения двух маршрутов к одной и той же подсети используется только метрика и не учитываются такие параметры: скорость передачи, надежность, доступная полоса пропускания.

RIP требует много времени для восстановления связи после сбоя в маршрутизаторе (минуты). В процессе установления режима возможны циклы.

В больших сетях чаще возникает проблема цикла. Хотя в RIP предусмотрен механизм распознавания петель, но в больших сетях соответствующие алгоритмы не рациональны (по времени), значительно увеличивают трафик сети.

Маршрут по умолчанию имеет адрес 0.0.0.0 (это верно и для других протоколов маршрутизации). Каждому маршруту ставится в соответствие *таймер тайм-аута* и "*сборщика мусора*". Тайм-аут-таймер сбрасывается каждый раз, когда маршрут инициализируется или корректируется. Если со времени последней коррекции прошло 3 минуты или получено сообщение о том, что вектор расстояния равен 16, маршрут считается закрытым. Но запись о нем не стирается, пока не истечет время "уборки мусора" (2мин). При появлении эквивалентного маршрута переключения на него не происходит, таким образом, блокируется возможность осцилляции между двумя или более равноценными маршрутами.

При реализации RIP можно выделить следующие режимы:

- ***Инициализация,*** определение всех "живых" интерфейсов путем посылки запросов, получение таблиц маршрутизации от других маршрутизаторов. Часто используются широковещательные запросы.
- ***Получен запрос***. В зависимости от типа запроса высылается адресату полная таблица маршрутизации, или проводится индивидуальная обработка.
- ***Получен отклик***. Проводится коррекция таблицы маршрутизации (удаление, исправление, добавление).
- ***Регулярные коррекции***. Каждые 30 секунд вся или часть таблицы маршрутизации посылается всем соседним маршрутизаторам. Могут посылаться и специальные запросы при локальном изменении таблицы.

Форматы сообщений протокола RIP (в одном сообщении может присутствовать информация о 25 маршрутах.) имеют вид (рисунок 1, дано для справки).

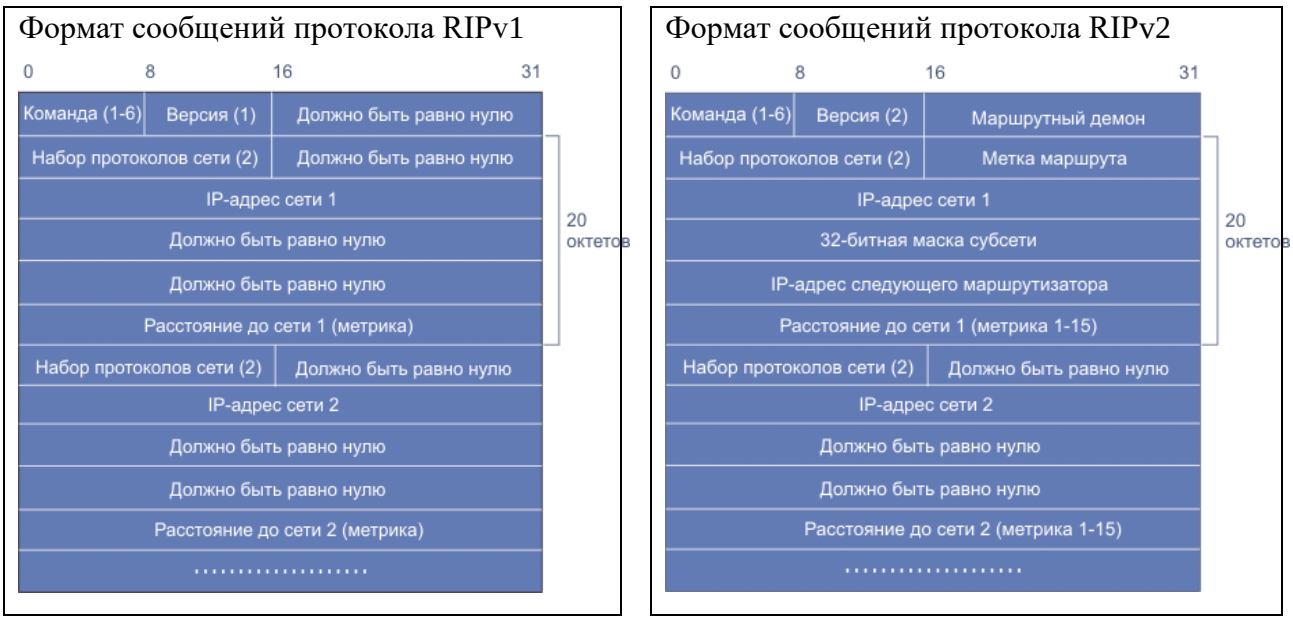


Рисунок 1. Формат сообщений протокола RIP

2.1. Построение таблиц маршрутизации

Так как построение таблиц маршрутизации в обеих версиях не отличается, то для упрощения записей будем ориентироваться на RIPv1.

Процесс построения таблиц маршрутизации рассмотрим на примере составной сети на рисунке 2.

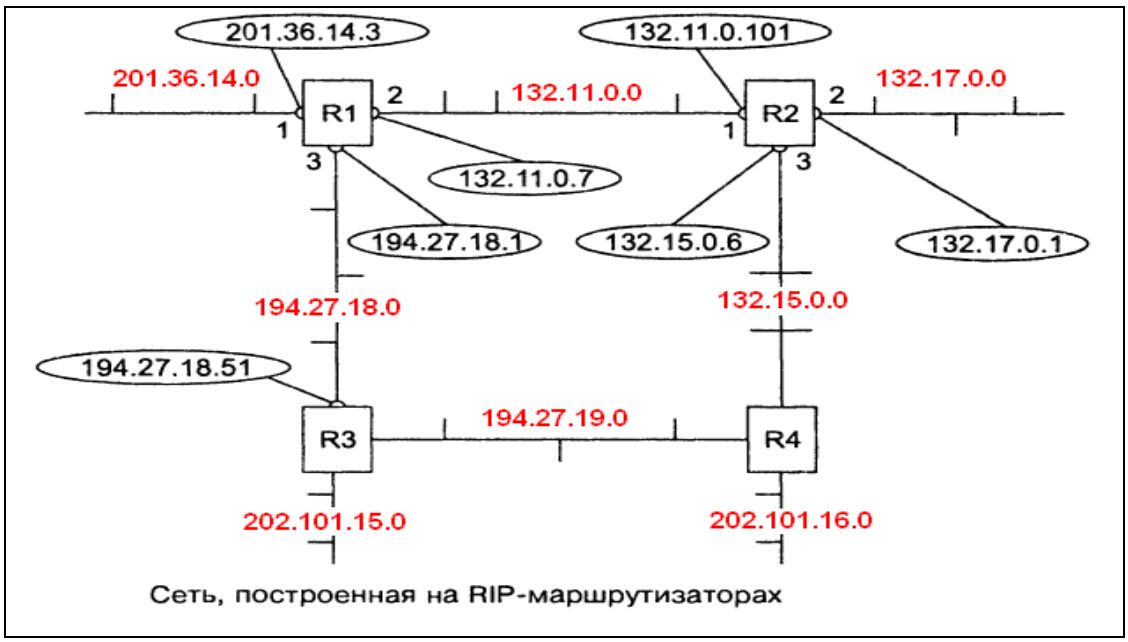


Рисунок 2. Пример составной сети (R1-R4 -роутеры)

Процесс построения таблиц маршрутизации разделим на пять этапов.

2.1.1. Этап 1 – создание минимальной таблицы

В рассматриваемой сети имеем восемь IP-подсетей (адреса сетей заданы красным цветом), связанных четырьмя маршрутизаторами R1, R2, R3 и R4.

В исходном состоянии на каждом маршрутизаторе программное обеспечение стека TCP/IP автоматически создает минимальную таблицу маршрутизации, в которой учитываются только непосредственно подсоединенные подсети. Адреса портов маршрутизаторов помещены в овалы.

Примерный вид минимальной таблицы маршрутизатора R1 представлен на рисунке 3.

Минимальная таблица маршрутизации маршрутизатора R1

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	201.36.14.3	1	1
132.11.0.0	132.11.0.7	2	1
194.27.18.0	194.27.18.1	3	1

Рисунок 3.

Минимальные таблицы маршрутизации в других маршрутизаторах будут выглядеть соответственно, например, таблица маршрутизатора R2 (рисунок 4) состоит из следующих трех записей

Минимальная таблица маршрутизации маршрутизатора R2

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
132.11.0.0	132.11.0.101	1	1
132.17.0.0	132.17.0.1	2	1
132.15.0.0	132.15.0.6	3	1

Рисунок 4.

2.1.2. Этап 2 – рассылка минимальной таблицы соседям

После инициализации каждый маршрутизатор начинает пересылать своим соседям сообщения протокола RIP, в которых содержится его минимальная таблица.

Таким образом, маршрутизатор R1 передает соседям-маршрутизаторам R2 и R3 следующие сообщения:

- сеть 201.36.14.0, расстояние 1;
- сеть 132.11.0.0, расстояние 1;
- сеть 194.27.18.0, расстояние 1;

RIP-сообщения передаются в дейтаграммах протокола UDP

2.1.3. Этап 3 – получение RIP-сообщений и обработка полученной информации

После получения аналогичных сообщений от R2 и R3 маршрутизатор R1 наращивает каждое полученное поле метрики на единицу и запоминает, через какой порт и от какого маршрутизатора получена новая информация. Затем маршрутизатор начинает сравнивать новую информацию с той, которая хранится в его таблице (рисунок 5).

Таблица маршрутизации маршрутизатора R1

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	201.36.14.3	1	1
132.11.0.0	132.11.0.7	2	1
194.27.18.0	194.27.18.1	3	1
132.17.0.0	132.11.0.101	2	2
132.15.0.0	132.11.0.101	2	2
194.27.19.0	194.27.18.51	3	2
202.101.15.0	194.27.18.51	3	2
~~132.11.0.0~~	~~132.11.0.101~~	~~2~~	~~2~~
~~194.27.18.0~~	~~194.27.18.51~~	~~3~~	~~2~~

Рисунок 5.

Записи с четвертой по девятую получены от соседних маршрутизаторов (выделены красным контуром), они претендуют на помещение в таблицу. Однако только записи с четвертой по седьмую попадают в таблицу, а записи восьмая и девятая (выделены синим контуром) – нет (так как они содержат данные, которые есть в таблице, строки 2 и 3, и метрика до них больше, чем в существующих записях).

Аналогичные операции с новой информацией выполняют и все остальные маршрутизаторы.

2.1.4. Этап 4 – рассылка новой таблицы соседям

Каждый маршрутизатор отсылает новое RIP-сообщение обо всех известных ему на данный момент подсетям (как непосредственно подкдюченных, так и удаленных, о которых маршрутизатор узнал от принятых RIP-сообщений).

2.1.5. Этап 5 – Получение RIP-сообщений от соседей и обработка полученной информации.

Пятый этап повторяет этап 3. Маршрутизаторы принимают RIP-сообщения, обрабатывают полученную в них информацию и на ее основании корректируют свои таблицы маршрутизации.

Рассмотрим, как это делает маршрутизатор R1, в результате обработки имеем отредактированную таблицу маршрутизации (рисунок 6).

Таблица маршрутизации маршрутизатора R1

Номер сети	Адрес следующего маршрутизатора	Порт	Расстояние
201.36.14.0	201.36.14.3	1	1
132.11.0.0	132.11.0.7	2	1
194.27.18.0	194.27.18.1	3	1
132.17.0.0	132.11.0.101	2	2
132.15.0.0	132.11.0.101	2	2
~~132.15.0.0~~	~~194.27.18.51~~	~~3~~	~~3~~
194.27.19.0	194.27.18.51	3	2
~~194.27.19.0~~	~~132.11.0.101~~	~~2~~	~~3~~
~~202.101.15.0~~	~~194.27.18.51~~	~~3~~	~~2~~
202.101.16.0	132.11.0.101	2	3
~~202.101.16.0~~	~~194.27.18.51~~	~~3~~	~~3~~

Рисунок 6.

Смотрите, маршрутизатор уже знает сеть 132.15.0.0, причем старая информация (строка 5) имеет лучшую метрику (=2), чем новая с метрикой 3. Поэтому новая информация (строка 6) об этой сети отбрасывается (вычеркивается). О сети 202.101.16.0 маршрутизатор R1 впервые узнал на этом этапе, причем данные пришли от двух соседей - R3 и R4. Метрики у строк 10 и 11 одинаковы, то в таблицу подают данные, пришедшме первыми. В нашей таблице пусть это будет R2.

Если маршрутизаторы периодически повторяют этапы рассылки и обработки RIP-сообщений, то за конечное время в сети установится корректный режим маршрутизации.

Под корректным режимом маршрутизации понимается такое состояние таблицы, когда все подсети достижимы из любой подсети с помощью некоторого маршрута. Говорят, что таблицы согласованы.

Если бы маршрутизаторы, их интерфейсы, их лини связи оставались работоспособными, то вышеописанный процесс можно делать достаточно редко, например, один раз в день, а не 30сек как в реальных условиях.

2.2. Адаптация маршрутизаторов RIP к изменению состояния сети

В сетях постоянно происходят изменения – меняется работоспособность маршрутизаторов и линий связи, кроме того, маршрутизаторы и линии связи могут добавляться в существующую сеть или же выводится из ее состава.

К новым маршрутам маршрутизаторы приспосабливаются очень просто. Они передают новую информацию в очередном RIP-сообщении соседям-маршрутизаторам, и постепенно эта информация становится известной всем маршрутизаторам составной сети.

А вот к изменениям, связанным из-за потери какого-либо маршрута, RIP-маршрутизаторы буксуют. Это, очевидно, объясняется тем, что в формате RIP-сообщения не были предусмотрены поля, которые бы указывали, что путь к данной подсети больше не существует.

Для уведомления того, что некий путь недействителен, используются два механизма:

- Истечение времени жизни;
- Указания специального (бесконечного, в RIP - 16 промежуточных маршрутизаторов) расстояния до сети, которая стала недоступной.

Механизм истечения времени жизни маршрута основан на том, что каждая запись в таблице маршрутизации имеет, полученная по RIP, имеет **время жизни – TTL**. При поступлении очередного RIP-сообщения, которое подтверждает справедливость данной записи, таймер TTL устанавливается в исходное состояние, а затем из него каждую секунду вычитается единица. Если за время тайм-аута не придет новое сообщение об этом маршруте, то он помечается как недействительный.

В протоколе RIP время рассылки выбрано 30сек, а в качестве тайм-аута шестикратное превышение – 180сек. Такой запас выбран для того, чтобы быть уверенным в том, что подсеть стала действительно недоступной, а не просто что RIP-сообщения потеряны (это возможно, так как используется UDP-дейтаграммный транспортный протокол, который ***не гарантирует*** *доставку сообщения*).

Если какой-то маршрутизатор отказал (перестал передавать служебные сообщения о сетях), то через 180сек все записи порожденные этим протоколом станут недействительными у его ближайших соседей, а через следующие 180 сек этот процесс повторится у соседей ближайших соседей, которые вычеркнут подобные записи уже через 360 секунд.

Очевидно, сообщения о неполадках распространяются с сети не очень быстро.

Механизм тайм-аута работает, когда маршрутизатор не может послать сообщение либо когда он сам неисправен или неисправна линия связи.

Если RIP-сообщение послать можно, то RIP-маршрутизатор для сети ставшей недоступной указывает ей бесконечное расстояние - 16. Получив сообщение, в котором расстояние до некоторой сети равно 16 (или 15 +1), маршрутизатор должен проверить , исходит ли это плохая информация о сети того же маршрутизатора, сообщение которого послужило в свое время основанием для записи о данной сети в таблице маршрутизации.

Если это тот же маршрутизатор, то информация считается достоверной и маршрут помечается как недоступный.

Причиной выбора бесконечного расстояния является то, что в некоторых случаях отказы сетей вызывают длительные периоды некорректной работы RIP- маршрутизаторов, выражающейся в зацикливании пакетов в петлях сети. Поэтому чем меньше расстояние, используемое в качестве бесконечного (было выбрано 16), тем такие периоды короче. Заметим, что проблема с петлей, образующейся между соседями-маршрутизаторами, решается с помощью **метода расщепления горизонта**.

Идея метода состоит в том, что маршрутная информация никогда не передается тому маршрутизатору, от которого она получена.

2.3. Пример маршрутной петли

Рассмотрим случай зацикливания пакетов в нашем примере. Пусть маршрутизатор R1 обнаружил, его связь с непосредственно подключенной сетью 201.36.14.0 потеряна (например, отказ интерфейса 201.36.14.3 – повредили кабель, рисунок 7).

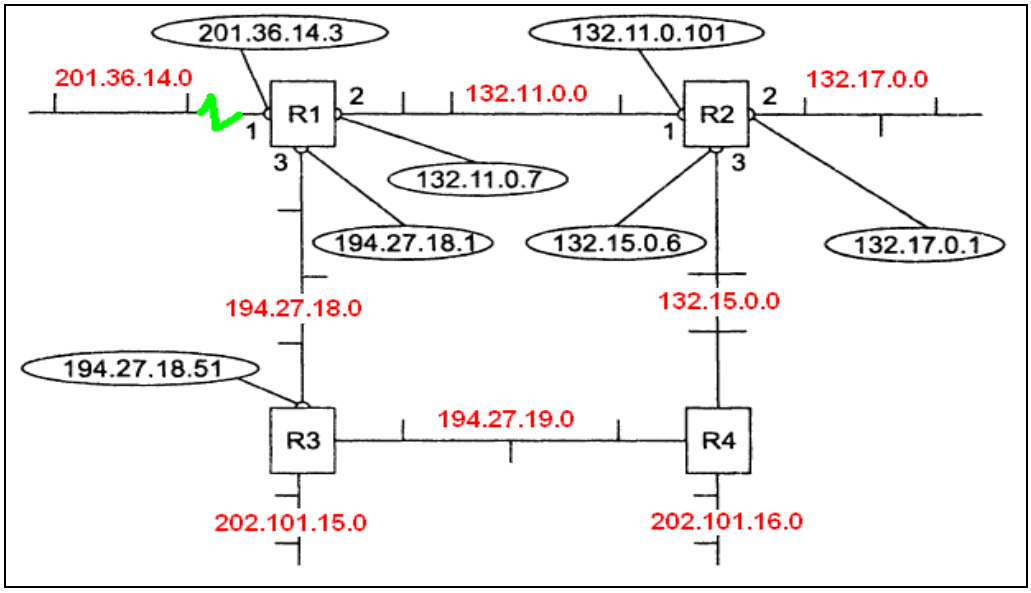


Рисунок 7. Пример составной сети с поврежденной связью

Маршрутизатор отмечает в своей таблице, что сеть 201.36.14.0 недоступна. Через максимум 30сек, когда начнется новый цикл RIP- объявлений соседи получат сообщение, что для сети 201.36.14.0 установлена метрика 16 (бесконечная метрика).

А так как таймеры у маршрутизаторов не синхронизированы, то может оказаться ситуация, R2 опередит R1 и передаст ему свое сообщение раньше, чем R1 передаст, что сеть 201.36.14.0 недоступна.

Заметим, что у маршрутизатора R2 есть запись вида (рисунок 8)

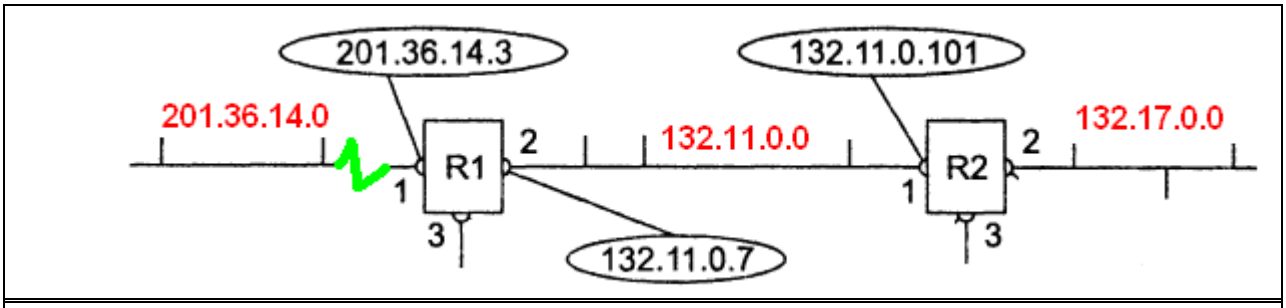
Фрагмент таблицы маршрутизации маршрутизатора R2

Номер сети	Адрес след. маршрутизатора	Порт	Расстояние
201.36.14.0	132.11.0.7	1	2

Рисунок 8. Фрагмент схемы составной сети и соответствующая запись в таблице маршрутизации

Эта запись на рисунке 4 была корректна до отказа интерфейса 201.36.14.3; теперь она уже устарела, но маршрутизатор об этом пока не знает.

Маршщрутизатор R1 получает новую информацию о сети 201.36.14.0 - это сеть достижима через маршрутизатор R2 с метрикой 2. Если раньше R1 получал эту же информацию от R2, но ее игнорировал, так его собственная метрика для 201.36.14.0 была лучше. Теперь все наоборот, у R2 метрика лучше (2 < 16) и R2 заменяет строку о недостижимости сети 201.36.14.0 строкой полученной от R2 (рисунок 9).



Фрагмент таблицы маршрутизации маршрутизатора R1

Номер сети	Адрес след. маршрутизатора	Порт	Расстояние
201.36.14.0	132.11.0.101	2	3

Рисунок 9. Фрагмент схемы составной сети и соответствующая запись в таблице маршрутизации

В результате образуется маршрутная петля: пакеты направляемые узлам сети 201.36.14.0 станут передаваться маршрутизатором R2 маршрутизатору R1, маршрутизатор R1 будет их возвращать маршрутизатору R2. Пакеты будут циркулировать до тех пор пока не истечет время жизни каждого пакета.

Рассмотрим периоды, кратные времени жизни записей в таблицах маршрутизаторов.

2.3.1. Время 0-180сек.

После отказа интерфейса маршрутизатор R2 по-прежнему будет снабжать R1 записью о сети 201.36.14.0 с метрикой 2, так как время ее жизни не истекло. Пакеты зацикливаются.

2.3.2. Время 180-360сек.

В начале этого периода у маршрутизатора R2 истекает время жизни записи о сети 201.36.14.0 с метрикой 2, так маршрутизатор R1 в предыдущий период посылал ему сообщения о сети 201.36.14.0 с худшей метрикой, чем R2. и они не могли подьверждать эту запись. Теперь R2 принимает от R1 запись о сети 201.36.14.0 с метрикой 3 и трансформирует ее в запись с метрикой 4. Маршрутизатор R1 больше не получает сообщений от R2 о сети 201.36.14.0 с метрикой 2, поэтому время его жизни начинает уменьшаться. Пакеты продолжают зацикливаться.

2.3.2. Время 360-540сек.

У маршрутизатора R1 истекает время жизни записи о сети 201.36.14.0 с метрикой 3. Маршрутизаторы опять меняются ролями – R2 снабжает R1 устаревшей информацией о пути к сети 201.36.14.0, но уже с метрикой 4, которую R1 преобразует в метрику 5. Пакеты продолжают зацикливаться.

Если бы в RIP не было выбрано расстояние 16, то процесс зациклся до бесконечности.

Но чтобы дойти до метрики 16 можно подсчитать понадобится 36 минут.

Поэтому ограничение в 16 маршрутизаторов сужает область применения RIP для сетей, у которых число промежуточных маршрутизаторов превышает число 15.

Отметим, если R1 успел раньше передать информацию о недостижимости сети 201.36.14.0 ложной информации маршрутизатора R2, то маршрутная петля не образовалась. Переходное состояние.

2.4. Борьба с ложными маршрутами в RIP

Хотя RIP не в состоянии полностью справиться с переходными процессами в сети, когда некоторые маршрутизаторы пользуются устаревшей информацией о уже несуществующих маршрутах, имеется несколько методов борьбы с этим явлением.

Метод расщепления горизонта

Этот метод заключается в том, маршрутная информация о некоторой сети, хранящаяся в таблице маршрутизации о некоторой сети, никогда не передается тому маршрутизатору, от которого она была получена.

Однако этот метод не работает в тех случаях, когда петли образуются не двумя, а большим числом маршрутизаторов.

Для предотвращения зацикливания пакетов по составным петлям при отказах связей применяют два других приема:

- прием триггерных обновлений,
- замораживание изменений

2.4.1. Прием триггерных обновлений

Идея. Маршрутизатор, получив данные об изменении метрики до какой либо сети, не ждет истечения периода передачи таблицы маршрутизации, а передает данные об изменившимся маршруте немедленно. Это в какой-то степени может предотвратить передачу уже устаревших сведений об отказавшем маршруте. Но он перегружает сеть служебными сообщениями, поэтому триггерные объявления делают с некоторой задержкой. Это приводит к тому, что некоторая устаревшая информация может проскочить.

2.4.2. Прием замораживание изменений

Служит для исключения неприятных ситуаций в приеме триггерных обновлений. Вводится тайм-аут на принятие новых данных о сети, которая только что стала недоступной. Этот тайм-аут предотвращает принятие устаревших данных о некотором маршруте от тех маршрутизаторов, которые находятся на некотором расстоянии от отказавшей связи. Предполагается, что в течении тайм-аута замораживания изменений эти маршрутизаторы вычеркнут данный маршрут из своих таблиц, так как не получат о нем новых записей и значит не будут распространять устаревшие сведения по составной сети.

3. Конфигурация RIPv2 и ее проверка

3.1. Задание 1. Проектирование сети

1. Логическая схема сети показана на рисунке 10. Согласно вашему варианту задания составьте адресную схему сети.

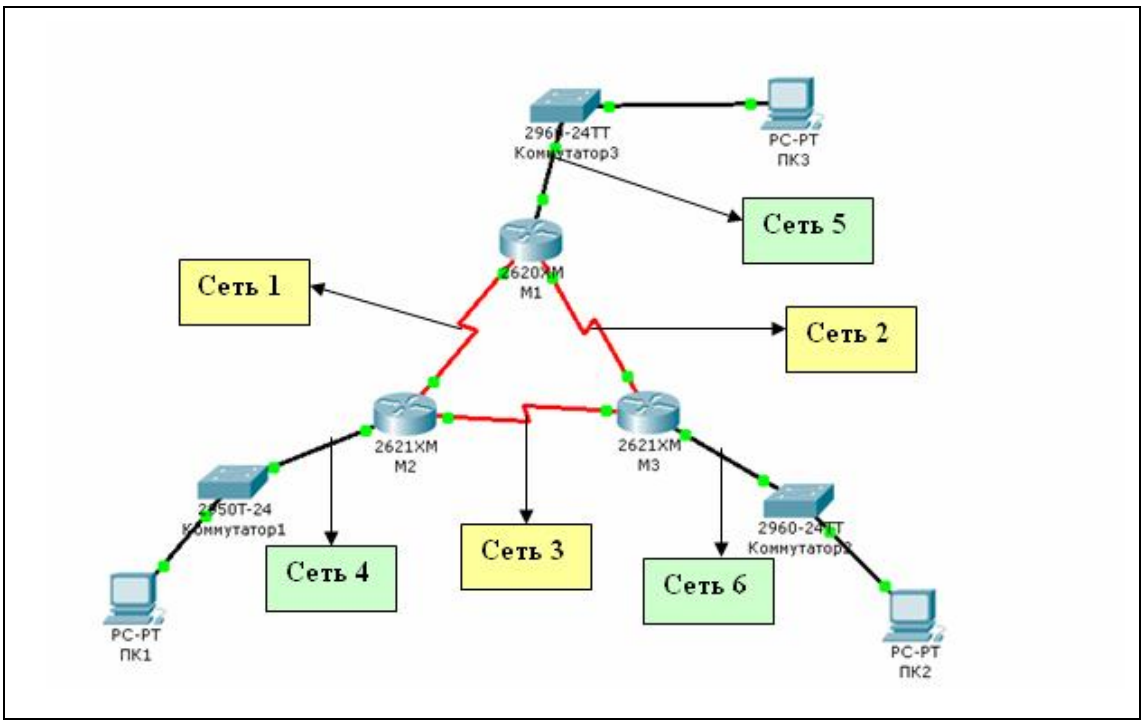


Рисунок 10.

2. **Используя CLI настроить сетевые интерфейсы всех устройств.**
3. Перед настройкой RIP назначьте IP-адреса и маски всем интерфейсам, задействованным в маршрутизации. Задайте при необходимости тактовую частоту для последовательных каналов.
4. **Подсети и интерфейсы маршрутизаторов подписать**
5. **После завершения базовой настройки выдайте таблицы маршрутизации и проанализируйте их содержимое.**
6. Перейдите к настройте протокола RIP.

3.2. Задание 2

1. Согласно вашему варианту задания, настройте RIPv2 на маршрутизаторах.

Команды настройки протокола RIP версии 2 на маршрутизаторах.

Базовая настройка RIP состоит из трех команд:

Router(config)#router rip - Включение протокола маршрутизации.
Router(config)#version 2 - Определение версии.

Router(config-router)#network [сетевой адрес] - Определение всех напрямую подключенных сетей, которым требуется уведомление протоколом RIP.

Протокол RIPv2 распространяет маршрут по умолчанию соседним маршрутизаторам вместе с обновлениями маршрутов.

3.2.1. Пример. *Настройка протокола RIPv1*

Приведем пример настройки протокола RIPv1. для сети (рисунок 11).

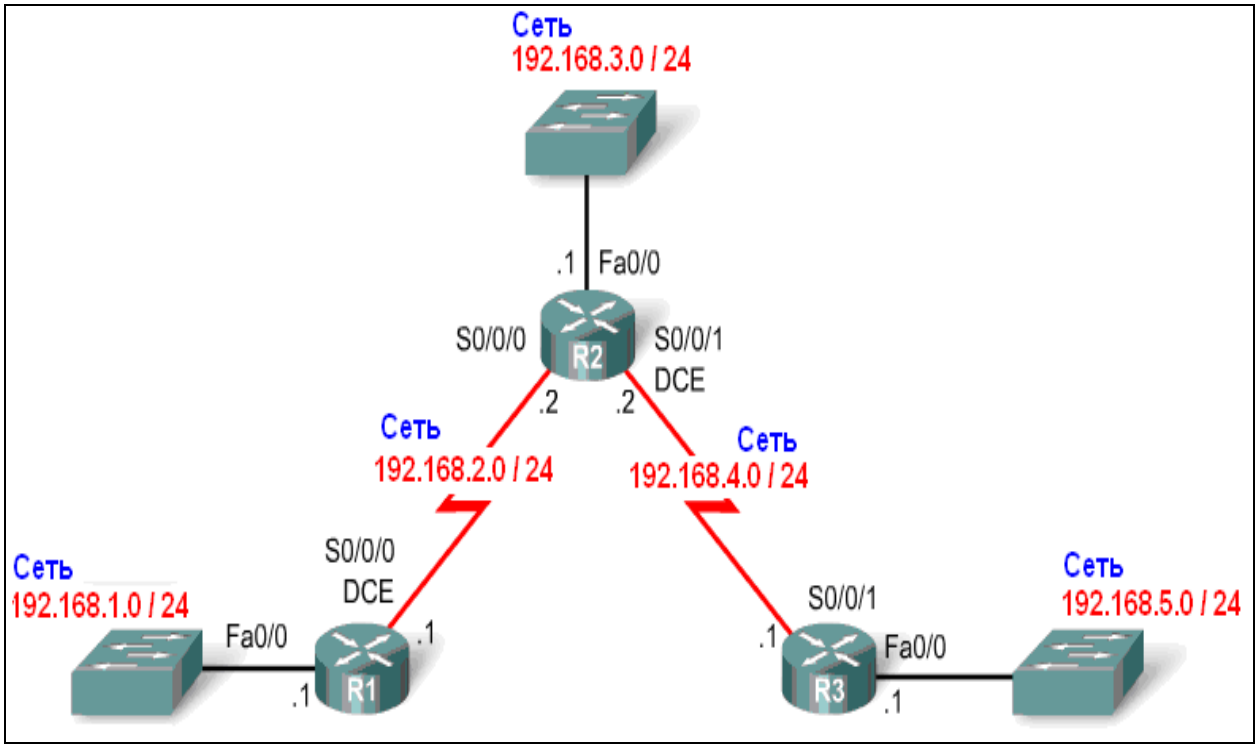


Рисунок 11.

Маршрутизатор **R1**

```
R1(config)#router rip
R1(config-router)#network 192.168.1.0
R1(config-router)#network 192.168.2.0
```

Маршрутизатор **R2**

```
R2(config)#router rip
R2(config-router)#network 192.168.2.0
R2(config-router)#network 192.168.3.0
R2(config-router)#network 192.168.4.0
```

Маршрутизатор **R3**

```
R3(config)#router rip
R3(config-router)#network 192.168.4.0
R3(config-router)#network 192.168.5.0
```

3.3. Задание 3. Тестирование протокола RIP

Использовать команды ***show ip protocols*** для инсталлированных протоколов и команду ***show ip route*** для просмотра таблиц маршрутизации всех маршрутизаторов.

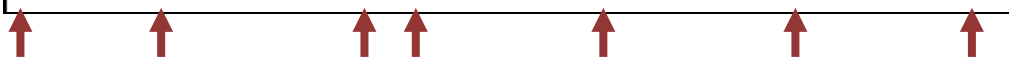
1. **Результаты тестирования представить в отчете.**
2. **Сделать анализ таблиц маршрутизации, полученных в заданиях 1 и 3**

Выходная информация команды ***show ip route:***

```
R1#show ip route
Codes: C - connected, S - static, I - IGRP, R - RIP, M - mobile, B - BGP
<output omitted>

Gateway of last resort is not set

R    192.168.4.0/24 [120/1] via 192.168.2.2, 00:00:02, Serial0/0/0
R    192.168.5.0/24 [120/2] via 192.168.2.2, 00:00:02, Serial0/0/0
C    192.168.1.0/24 is directly connected, FastEthernet0/0
C    192.168.2.0/24 is directly connected, Serial0/0/0
R    192.168.3.0/24 [120/1] via 192.168.2.2, 00:00:02, Serial0/0/0
```



1 2 3 4 5 6 7

1. Источник записи в ТМ (R-из протокола RIP)
2. Сеть назначения.
3. Административное расстояние (у протокола RIP = 120) *
4. Метрика маршрута
5. IP-адрес шлюза
6. Время, прошедшее с последнего обновления
7. IP-адрес интерфейса, через который доступна сеть назначения

Административное расстояние определяет предпочтение маршрута. Каждый источник маршрута, включая статические маршруты, расположен по приоритетам. Маршрутизаторы Cisco используют административное расстояние, чтобы выбрать лучший путь, когда узнают о той же самой сети назначения из двух или больше различных источников.

Административное расстояние – целое число от 0 до 255. Административное расстояние **0** является самым привилегированным. Административное, расстояние 0 только у непосредственно связанной сети и не может быть изменено.

3.4. Задание 4. Конфигурирование пассивных интерфейсов

Протокол ***RIP*** выполняет рассылку обновлений по всем своим интерфейсам. Например, маршрутизатор ***R2*** (из примера) отправляет обновления о маршрутах через интерфейс ***Fa0/0***, но в той подсети нет ***RIP*** устройств. Следовательно, обновления через данный интерфейс не необходимы: это лишний трафик, дополнительная обработка, страдает безопасность – широковещательный трафик легко перехватывается с помощью ***sniffing software***.

Решением проблемы является использование команды пассивного интерфейса, которая предотвращает передачу обновлений через интерфейс маршрутизатора, но позволяет эту сеть анонсировать к другим маршрутизаторам.

Команда пассивного интерфейса имеет вид:

Router(config-router)#passive-interface *interface-type interface-number*

```
R2(config-router)#passive-interface FastEthernet 0/0
```

1. **Для заданной сети для всех маршрутизаторов определить и настроить пассивные интерфейсы.**
2. **Сравнить объем трафика с трафиком в предыдущих заданиях.**

3.5. Задание 5. Тестирование сети

1. Используя команды (какие?) проверить достижимость всех узлов пользователей.
2. **Выдать снова таблицы маршрутизации всех трех маршрутизаторов**.
   Можете воспользоваться любыми допустимыми средствами.
   Проанализируйте ранее выданные и сейчас таблицы маршрутизации
3. Сохраните модель в файле ***Lab9_FIO_01.pkt***.
4. Сделайте копию модели сети в файле ***Lab9_FIO_02.pkt.***
   Далее продолжайте работать только с моделью в файле ***Lab9_FIO_02.pkt***
5. Разорвите канал связи между какой-нибудь парой смежных маршрутизаторов (см , например; рисунок 11) схема должна быть представлена в отчете.
6. **Снова проверить достижимость всех узлов пользователей.**
7. Снова выдать таблицы маршрутизации всех трех маршрутизаторов.
8. **Проанализировать таблицы маршрутизации *до и после* разрыва канала связи. Сделать выводы.**

3.6. Задание 6. Тесты

Дать письменно в отчет аргументированные ответы на следующие вопросы.

1. **Может ли работать маршрутизатор, не имея таблицы маршрутизации?**

   Варианты ответов:

   а) может, если выполняется маршрутизация от источника;

   б) нет, это невозможно;

   в) может, если в маршрутизаторе задан маршрут по умолчанию;

   г) может, если выполняется лавинная маршрутизация

2. **Можно ли обойтись в сети без протоколов маршрутизации?**
3. **По какой причине в протоколе RIP расстояние в 16 хопов между сетями полагается недостижимым?**

   Варианты ответов:

   а) поле, отведенное для хранения значения расстояния,
   имеет длину 4 двоичных разряда;

   б) сети, в которых работает RIP, редко бывают большими;

   в) для получения приемлемого времени сходимости алгоритма.

3.7. Дополнительное задание 7 (только для желающих).

Легенда

1. Сделайте копию модели (***Lab9_ФИО_02.pkt***) сети в файле ***Lab9_ФИО_03.pkt.***
   Далее продолжайте работать только с моделью в файле ***Lab9_ФИО_03.pkt.***
2. В силу экономических, политических или иных причин, а ,больше всего по вине администратора сети **192.168.5.0** был уничтожен канал связи между парой смежных маршрутизаторов **R2** и **R3** (см. например; рисунок 12).

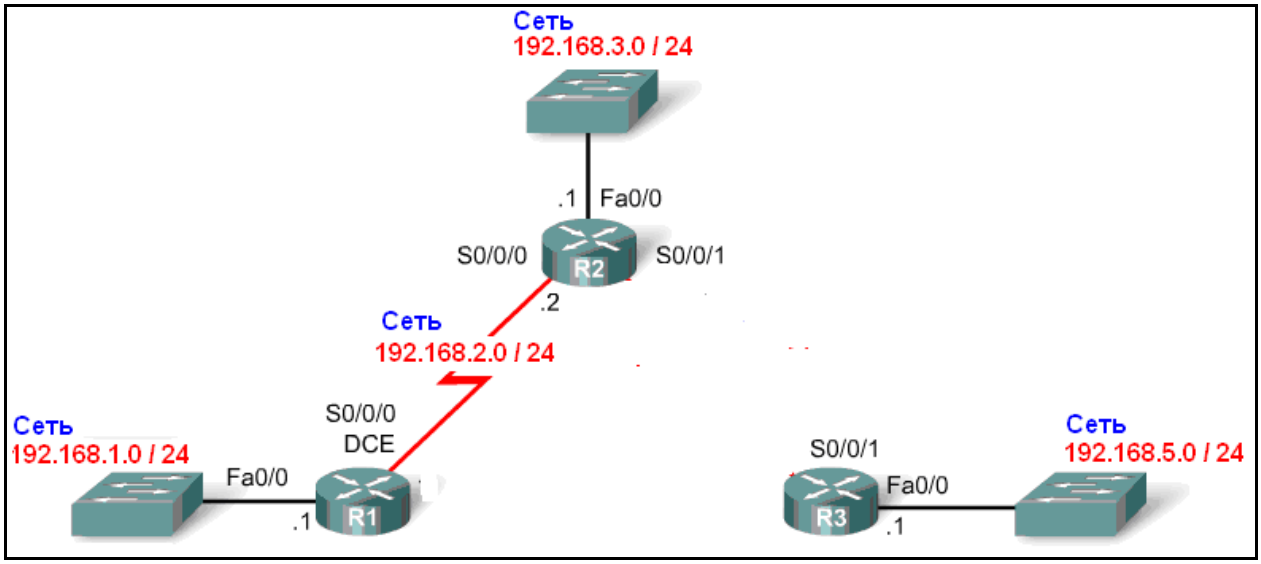


Рисунок 12

3. **Снова проверить достижимость всех узлов пользователей.**

   Чтобы не загромождать схемы на рисунках 11 и 12 не указаны хосты, которые присутствуют в сети (см. рисунок 10)

4. Опять выдать таблицы маршрутизации всех трех маршрутизаторов.
5. **Проанализировать динамику изменения таблиц маршрутизации,** начиная с моделей ***Lab9_FIO_01.pkt, Lab9_ FIO _02.pkt и Lab9_ FIO _03.pkt.***

   Для анализа динамики изменения содержимого таблиц маршрутизации рекомендую использовать инструмент “лупа”; держать одновременно на экране монитора все три таблицы маршрутизации каждого маршрутизатора; следить за их изменением в реальном времени. Будьте внимательны, и вы сможете обнаружить момент изменения таблиц маршрутизации.

6. Сделайте копию модели (***Lab9_ФИО_03.pkt***) сети в файле ***Lab9_ФИО_04.pkt.***
7. **Решение проблемы.**

   Администратор сети **192.168.5.0** , чтобы скрыть эту ситуацию (см. п.2), и свою вину от своего руководства и как то оживить сеть, подпольно подсоединил свой маршрутизатор **R3** к коммутатору-хабу сети **192.168.1.0.** Схему подключения представить в отчете.

8. **Оценить решение студентов.**
   - Снова проверить достижимость всех узлов пользователей.
   - Опять выдать таблицы маршрутизации всех трех маршрутизаторов.
   - Выполнить сравнительный анализ таблиц маршрутизации.

3.8. Задание 8. Подготовка отчетных документов

В отчете (в качестве пунктов) обязательно должны присутствовать пункты заданий, выделенные синим цветом.

Создать на сервере папку **N_Lab9_FIO,** где **N** одно из чисел **3** или **4** в зависимости от номера группы Разработанные модели сетей сохранить в файлах ***N_Lab9_ FIO _01.pkt*** ***(до разрыва канала связи),*** ***N_ Lab9_ FIO _02.pkt*** ***(после разрыва канала связи)***, ***N_Lab9_ FIO _03.pkt*** *и* ***N_Lab9_ FIO _04.pkt*** (***дополнительное задание***), *а* отчет в файле ***N_Lab9_ FIO.doc***. В качестве **FIO** использовать только ФАМИЛИЮ.

4. Варианты заданий

Вариант	*Сеть 1 - 6*	Вариант	*Сеть 1 - 6*	Вариант	*Сеть 1 - 6*
1	11.0.0.0/16 12.0.0.0/16 13.0.0.0/16 14.0.0.0/16 15.0.0.0/16 16.0.0.0/16	**2**	31.0.0.0/16 32.0.0.0/16 33.0.0.0/16 34.0.0.0/16 35.0.0.0/16 36.0.0.0/16	**3**	196.5.1.0/24 196.5.2.0/24 196.5.3.0/24 196.5.4.0/24 196.5.5.0/24 196.5.6.0/24
Вариант	*Сеть 1 - 6*	**Вариант**	*Сеть 1 - 6*	**Вариант**	*Сеть 1 - 6*
4	180.101.0.0/16 180.102.0.0/16 180.103.0.0/16 180.104.0.0/16 180.105.0.0/16 180.106.0.0/16	**5**	203.21.140.0/24 203.21.141.0/24 203.21.142.0/24 203.21.143.0/24 203.21.144.0/24 203.21.145.0/24	**6**	179.11.0.0/16 179.12.0.0/16 179.13.0.0/16 179.14.0.0/16 179.15.0.0/16 179.16.0.0/16
Вариант	Сеть 1 - 6	**Вариант**	Сеть 1 - 6	**Вариант**	Сеть 1 - 6
7	205.100.1.0/24 205.100.2.0/24 205.100.3.0/24 205.100.4.0/24 205.100.5.0/24 205.100.6.0/24	**8**	155.10.0.0/16 155.11.0.0/16 155.12.0.0/16 155.13.0.0/16 155.14.0.0/16 155.15.0.0/16	**9**	200.192.210.0/24 200.192.211.0/24 200.192.212.0/24 200.192.213.0/24 200.192.214.0/24 200.192.215.0/24
Вариант	Сеть 1 - 6	**Вариант**	Сеть 1 - 6	**Вариант**	Сеть 1 - 6
10	187.16.0.0/18 187.17.0.0/18 187.18.0.0/18 187.19.0.0/18 187.20.0.0/18 187.21.0.0/18	**11**	192.168.1.0/24 192.168.2.0/24 192.168.3.0/24 192.168.4.0/24 192.168.5.0/24 192.168.6.0/24	**12**	111.0.0.0/24 112.0.0.0/24 113.0.0.0/24 114.0.0.0/24 115.0.0.0/24 116.0.0.0/24
Вариант	Сеть 1 - 6	**Вариант**	Сеть 1 - 6	**Вариант**	Сеть 1 - 6
13	161.11.0.0/24 161.12.0.0/24 161.13.0.0/24 161.14.0.0/24 161.15.0.0/24 161.16.0.0/24	**14**	54.0.0.0/24 55.0.0.0/24 56.0.0.0/24 57.0.0.0/24 58.0.0.0/24 59.0.0.0/24	**15**	81.0.0.0/16 82.0.0.0/16 83.0.0.0/16 84.0.0.0/16 85.0.0.0/16 86.0.0.0/16

Вариант	Сеть 1 - 6	Вариант	Сеть 1 - 6	Вариант	Сеть 1 - 6
16	101.0.0.0/24 102.0.0.0/24 103.0.0.0/24 104.0.0.0/24 105.0.0.0/24 106.0.0.0/24	**17**	181.79.0.0/24 181.80.0.0/24 181.81.0.0/24 181.82.0.0/24 181.83.0.0/24 181.84.0.0/24	**18**	171.123.0.0/24 171.124.0.0/24 171.125.0.0/24 171.126.0.0/24 171.127.0.0/24 171.128.0.0/24
Вариант	Сеть 1 - 6	**Вариант**	Сеть 1 - 6	**Вариант**	Сеть 1 - 6
19	206.208.101.0/24 206.208.102.0/24 206.208.103.0/24 206.208.104.0/24 206.208.105.0/24 206.208.106.0/24	**20**	128.100.0.0/16 128.101.0.0/16 128.102.0.0/16 128.103.0.0/16 128.104.0.0/16 128.105.0.0/16	**21**	137.42.0.0/24 137.43.0.0/24 137.44.0.0/24 137.45.0.0/24 137.46.0.0/24 137.47.0.0/24
Вариант	Сеть 1 - 6	**Вариант**	Сеть 1 - 6	**Вариант**	Сеть 1 - 6
22	221.56.1.0/24 221.56.2.0/24 221.56.3.0/24 221.56.4.0/24 221.56.5.0/24 221.56.6.0/24	**23**	91.0.0.0/16 92.0.0.0/16 93.0.0.0/16 94.0.0.0/16 95.0.0.0/16 96.0.0.0/16	**24**	121.0.0.0/24 122.0.0.0/24 123.0.0.0/24 124.0.0.0/24 125.0.0.0/24 126.0.0.0/24
Вариант	Сеть 1 - 6	**Вариант**	Сеть 1 - 6	**Вариант**	Сеть 1 - 6
25	201.16.1.0/24 201.16.2.0/24 201.16.3.0/24 201.16.4.0/24 201.16.5.0/24 201.16.6.0/24	**26**	211.16.1.0/24 211.16.2.0/24 211.16.3.0/24 211.16.4.0/24 211.16.5.0/24 211.16.6.0/24	**27**	100.10.0.0/16 100.20.0.0/16 100.30.0.0/16 100.40.0.0/16 100.50.0.0/16 100.60.0.0/16
Вариант	Сеть 1 - 6	**Вариант**	Сеть 1 - 6	**Вариант**	Сеть 1 - 6
28	223.19.1.0/24 223.19.2.0/24 223.19.3.0/24 223.19.4.0/24 223.19.5.0/24 223.19.6.0/24	**29**	192.19.1.0/24 192.19.2.0/24 192.19.3.0/24 192.19.4.0/24 192.19.5.0/24 192.19.6.0/24	**30**	11010.0.0/16 11020.0.0/16 11030.0.0/16 11040.0.0/16 11050.0.0/16 11060.0.0/16