Лабораторная работа № 1
Введение в сетевые технологии. Знакомство с CLI.

1.1. Цель работы

Лабораторная работа №1 предназначена для изучения основ функционирования современных сетей передачи данных. Она представляет собой тренировочный сценарий для симулятора Cisco Packet Tracer. Для успешного выполнения лабораторной работы студентам необходимо выполнить индивидуальное задание, подготовить отчет (по своему варианту) и защитить его в форме собеседования.

1.2. Теоретическая часть

1.2.1. IPv4 – адрес

IP-адрес (ай-пи адрес, сокращение от англ. Internet Protocol Address) — сетевой адрес узла в компьютерной сети, построенной по протоколу IP. При связи через сеть Интернет требуется глобальная уникальность адреса, в случае работы в локальной сети требуется уникальность адреса в пределах сети.

В зависимости от назначения IP-адреса делятся на индивидуальные (unicast – служат для адресации одного конкретного узла сети), групповые (multicast – идентифицируют группу узлов, объединенную общим признаком, например, использующую общий протокол) и широковещательные (broadcast – идентифицирующие все узлы сети).

IP-адрес представляет собой 32-битное двоичное число, разделенное на 4 байта (октета). В структуре индивидуальных адресов выделяют две составные части: номер сети (N – network, характеризует адрес сети в целом и совпадает для всех узлов сети) и номер узла в сети (H – host). Сетевая часть служит для определения адреса сети, в которую необходимо доставить пакет, а узловая часть определяет тот конкретный узел, которому это сообщение предназначено.

Примером сетевой и узловой части в повседневной жизни можно назвать отправление письма по адресу: 123456 Москва, Сетевая улица, дом 128, квартира 255, где 123456 Москва, Сетевая улица, дом 128 – это сетевая часть, общая для всех квартир данного дома, а квартира 255 – узловая часть.

Таким образом, IP-адрес состоит из двух частей: номера сети (левые ***n*** бит) и номера узла (правые (32-***n***) бит). Чем больше значение ***n***, тем меньше

размер сети и количество узлов в ней, но тем больше таких сетей можно создать. Верно и обратное.

В IP адресе каждый из четырех октетов состоит из 8 бит, каждый бит имеет значение 0 или 1. Четыре группы из 8 бит имеют одну и ту же серию допустимых десятичных значений – от 0 до 255 включительно. Значения каждого размещения бита соответствует степени двойки от 7 до 0: 128, 64, 32, 16, 8, 4, 2 и 1.

Если все 8 бит имеют значение 0 (00000000), значение октета равно 0.

Если все 8 бит имеют значение 1 (11111111), значение октета равно 255 (128 + 64 + 32 + 16 + 8 + 4 + 2 + 1).

Если среди 8 бит есть и единицы, и нули, то степени двойки, соответствующие единицам, складываются друг с другом. Например, значение октета 0010 0111 составляет $39 = 2^5 + 2^2 + 2^1 + 2^0 = (32 + 4 + 2 + 1)$, т.к. в данном случае единицы стоят на (считать начинаем справа налево с нуля) нулевом, первом, втором и пятом месте.

Таким образом, значение каждого из четырёх октетов находится в диапазоне от 0 до 255.

Пример: 192.168.1.12 перевести в двоичную систему.

192 $= 1\cdot2^7+1\cdot2^6+0\cdot2^5+0\cdot2^4+0\cdot2^3+0\cdot2^2+0\cdot2^1+0\cdot2^0$ *= 11000000*

168 $= 1\cdot2^7+0\cdot2^6+1\cdot2^5+0\cdot2^3+1\cdot2^3+0\cdot2^2+0\cdot2^1+0\cdot2^0$ *= 10101000*

1 $= 0\cdot2^7+0\cdot2^6+0\cdot2^5+0\cdot2^3+0\cdot2^3+0\cdot2^2+0\cdot2^1+1\cdot2^0$ *= 00000001*

12 $= 0\cdot2^7+0\cdot2^6+0\cdot2^5+0\cdot2^3+1\cdot2^3+1\cdot2^2+0\cdot2^1+0\cdot2^0$ *= 00001100*

192.168.1.12 = 1100000. 1010100. 0000001. 0001100

Есть два способа определения того, сколько бит IP-адреса отводится на номер сети, а сколько на номер узла. Так, существует т.н. классовая адресация (INET) и бесклассовая адресация (CIDR). При классовой адресации значение ***n*** определяется классом сети, который в свою очередь зависит от старших бит IP-адреса.

В настоящее время чаще применяется бесклассовая адресация, при которой значение n определяется сетевой маской (netmask). Как и сам адрес, маска состоит из 32-х бит, разделенных на четыре октета. IP-адрес и маска подсети записываются друг под другом, как показано на рисунке 1.1.

Важнейшей особенностью сетевой маски является строгое правило группировки всех бит-единиц слева. Если в IP-адресе единицы и нули могут чередоваться друг с другом в любом октете и в любом порядке (например, 01101001), то в маске всегда и все единицы располагаются в старших битах, строго подряд, а затем следуют нули. Тот бит, в котором в маске заканчиваются единицы и начинаются нули, и является границей между

сетевой и узловой частями соответствующего IP-адреса. Таким образом, единицы в маске подсети определяют сетевую часть, а нули – узловую.

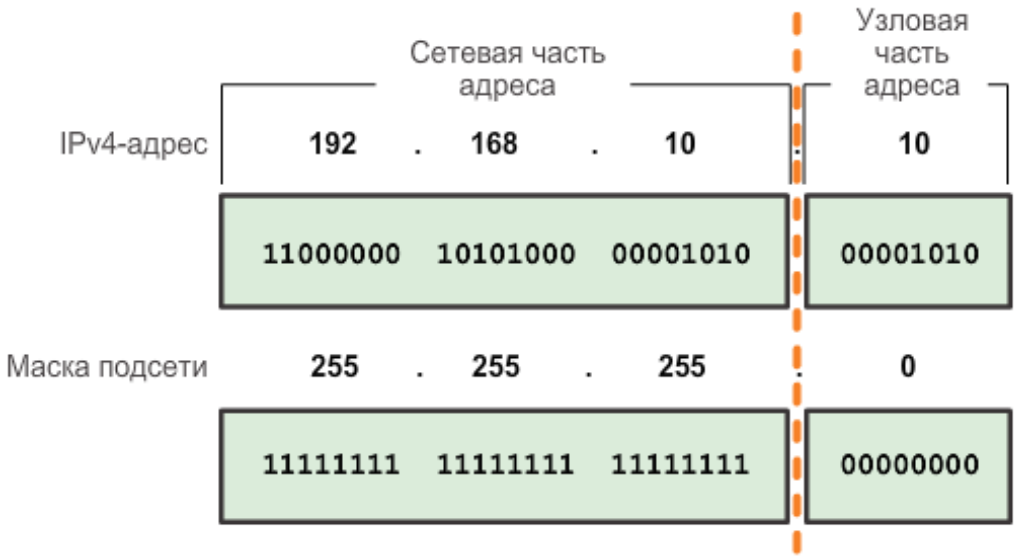

Рисунок 1.1. IP-адрес и маска подсети

Каждому биту в октете присваивается конкретное значение: у старшего бита, идущего первым, это значение равно 128, у младшего бита в октете это значение равно 1. Сумма октета, состоящего из всех единиц, равняется 255, октет, состоящий только из нулей, равен 0.

Маска подсети необходима для установления ограничения по количеству узлов в одной сети. Чем больше сетевая часть (количество единичных разрядов) в маске подсети, тем меньше количество узлов можно разместить в этой сети. Общее количество адресов в подсети вычисляется как 2 в степени, равной количеству нулей в сетевой маске.

Пример. Посчитаем количество адресов в сети с маской 255.255.255.0. Переводим маску в двоичный вид: 11111111 11111111 11111111 00000000 – 24 единицы, 8 нулей. Значит, в сети возможно 2^8 адресов, т.е. 256.

Кроме десятичной и двоичной форм записи маски, существует еще один способ обозначить ее размер. Этот тип записи называется префикс и имеет вид косой черты с числом, например, /19 и записывается в паре с IP-адресом – 10.96.47.0/19. Значение префикса соответствует количеству единиц в маске, а значит, и количеству бит сетевой части адреса. Так как маска подсети состоит из 32-х битов, а первые 19 бит единицы, то двоичная запись такой маски будет выглядеть следующим образом: 11111111 11111111 11100000 00000000.

Пример. Десятичный вид маски подсети, записанный префиксом, для адреса 10.96.47.0/19, имеет следующий вид: 10.96.47.0 255.255.224.0.

Отметим, что особенность взаимного расположения бит в маске задает конечный диапазон допустимых значений каждого октета маски, приведенный в таблице 1.1.

2^7 128	2^6 64	2^5 32	2^4 16	2^3 8	2^2 4	2^1 2	2^0 1	Байт маски
1	**1**	**1**	**1**	**1**	**1**	**1**	**1**	255
1	**1**	**1**	**1**	**1**	**1**	**1**	**0**	254
1	**1**	**1**	**1**	**1**	**1**	**0**	**0**	252
1	**1**	**1**	**1**	**1**	**0**	**0**	**0**	248
1	**1**	**1**	**1**	**0**	**0**	**0**	**0**	240
1	**1**	**1**	**0**	**0**	**0**	**0**	**0**	224
1	**1**	**0**	**0**	**0**	**0**	**0**	**0**	192
1	**0**	**0**	**0**	**0**	**0**	**0**	**0**	128
0	**0**	**0**	**0**	**0**	**0**	**0**	**0**	0

Таблица 1.1. Возможные значения октетов маски

Наложенная на конкретный IP-адрес, маска задает верхнюю и нижнюю границы подсети. Границами любой подсети служат минимальный адрес в сети, называемый также адресом сети (subnet, SN) и максимальный адрес в сети, который является широковещательным адресом по всем узлам, входящим в сеть (broadcast, BC). Для определения значений subnet и broadcast необходимо проверить, в каком октете маски подсети последовательность единиц заканчивается, и маска становится меньше 255. В нашем случае это третий октет, который имеет значение 224. Следующим действием необходимо расписать в двоичном виде третьи октеты маски подсети и IP-адреса один под одним. Третий октет маски под сети со значением 224, набирается суммированием первых трех старших бит октета, имеющих значение 128 + 64 + 32. Третий октет IP-адреса, равный 47, набирается суммированием значений 32 + 8 + 4 +2 + 1, во всех этих местах устанавливается значение 1, битам, которые не участвуют в суммировании, устанавливается значение 0. Затем по границе единиц и нулей в маске делим адрес на фиксированную часть (сеть) и свободную (узел).

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
	128	64	32	16	8	4	2	1
47	**0**	**0**	**1**	**0**	**1**	**1**	**1**	**1**
224	**1**	**1**	**1**	**0**	**0**	**0**	**0**	**0**

Маска нужна только для нахождения границы сетевой и узловой части. Граница найдена, маска более не понадобится. Для нахождения адресов SN и BC необходимо заполнить узловую часть всеми нулями (SN) и всеми единицами (BC) соответственно.

	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0
	128	64	32	16	8	4	2	1
47	**0**	**0**	**1**	**0**	**1**	**1**	**1**	**1**
subnet	**0**	**0**	**1**	**0**	**0**	**0**	**0**	**0**
broadcast	**0**	**0**	**1**	**1**	**1**	**1**	**1**	**1**

Переводим полученные двоичные числа в десятичную форму. Получаем для subnet $2^5=32$, для broadcast $2^5+2^4+2^3+2^2+2^1+2^0=63$. Вспомним, что это третий октет IP-адреса, и есть еще четвертый, который, так же, как и младшие биты третьего (после границы) в адресах SN и BC заполнен всеми нулями и всеми единицами соответственно. Следовательно, значение четвертого октета для SN равно нулю, а для BC – 255.

Получаем ответ: subnet: 10.96.32.0, broadcast: 10.96.63.255.

Результатом нахождения адресов subnet и broadcast является диапазон адресов, входящих в подсеть, которые можно использовать для присвоения их узлам в сети – больше первого и меньше второго. **Адреса со значением subnet и broadcast присваивать узлам запрещено.** Первый и последний из доступных для адресации узлов адреса называются hostmin и hostmax соответственно.

В протоколе IP существует несколько соглашений об особой интерпретации IP-адресов. Так, если весь IP-адрес состоит только из двоичных нулей, то он обозначает адрес того узла, который сгенерировал этот пакет; этот режим используется только в некоторых сообщениях ICMP. Если все двоичные разряды IP-адреса равны 1, то пакет с таким адресом назначения должен рассылаться всем узлам, находящимся в той же сети, что и источник этого пакета. Такая рассылка называется ограниченным широковещательным сообщением (limited broadcast). Если в поле номера узла назначения стоят только единицы, то пакет имеющий такой адрес, рассылается всем узлам сети с заданным номером сети. Например, в сети 192.190.21.0 с маской 255.255.255.0 пакет с адресом 192.190.21.255 доставляется всем узлам этой

сети. Такая рассылка называется широковещательным сообщением (broadcast).

Для изолированной сети её адрес может быть выбран администратором из специально зарезервированных для таких сетей блоков адресов (192.168.0.0/16, 172.16.0.0/12 или 10.0.0.0/8). Если же сеть должна работать как составная часть Интернета, то адрес сети выдаётся провайдером либо региональным интернет-регистратором (Regional Internet Registry, RIR). Региональные регистраторы получают номера автономных систем и большие блоки адресов у IANA, а затем выдают номера автономных систем и блоки адресов меньшего размера локальным интернет-регистраторам (Local Internet Registries, LIR), обычно являющимся крупными провайдерами.

IANA (от англ. Internet Assigned Numbers Authority — «Администрация адресного пространства Интернет») — американская некоммерческая организация, управляющая пространствами IP-адресов, доменов верхнего уровня.

1.2.2. Инкапсуляция и декапсуляция

Инкапсуляция – это процесс передачи данных с верхнего уровня приложений вниз (по стеку протоколов) к физическому уровню, чтобы быть переданными по сетевой физической среде (витая пара, оптическое волокно, Wi-Fi, и др.). Причём на каждом уровне различные протоколы добавляют к передающимся данным свою информацию.

Пример инкапсуляции данных (рисунок 1.2). На компьютере PCA ввели адрес веб-страницы в строке браузера. После этого браузер должен отправить запрос на сервер, на котором хранится эта страница. Введенный адрес страницы и является данными, которые должны передаться на сервер в виде запроса.

Данные с прикладного уровня спускаются на уровень представления. На том уровне данные преобразуются в формат удобный для передачи на нижний уровень.

Транспортный уровень получает данные и определяет, что дальше они должны быть переданы, используя протокол TCP. Перед передачей транспортный уровень разбивает данные на кусочки данных и добавляет к каждому кусочку заголовок, в котором содержится информация о логических портах компьютеров (с какого данные были посланы) и для какого предназначаются. На транспортном уровне эти кусочки данных с заголовком называются **блоком данных**. Блок данных передаётся дальше вниз к сетевому уровню.

Сетевой уровень, получая каждый блок данных, разделяет его на еще более маленькие части и к каждой части добавляет свой заголовок. В заголовке сетевого уровня указываются логические сетевые адреса отправителя и получателя. Эти части на сетевом уровне называются **пакетами.**

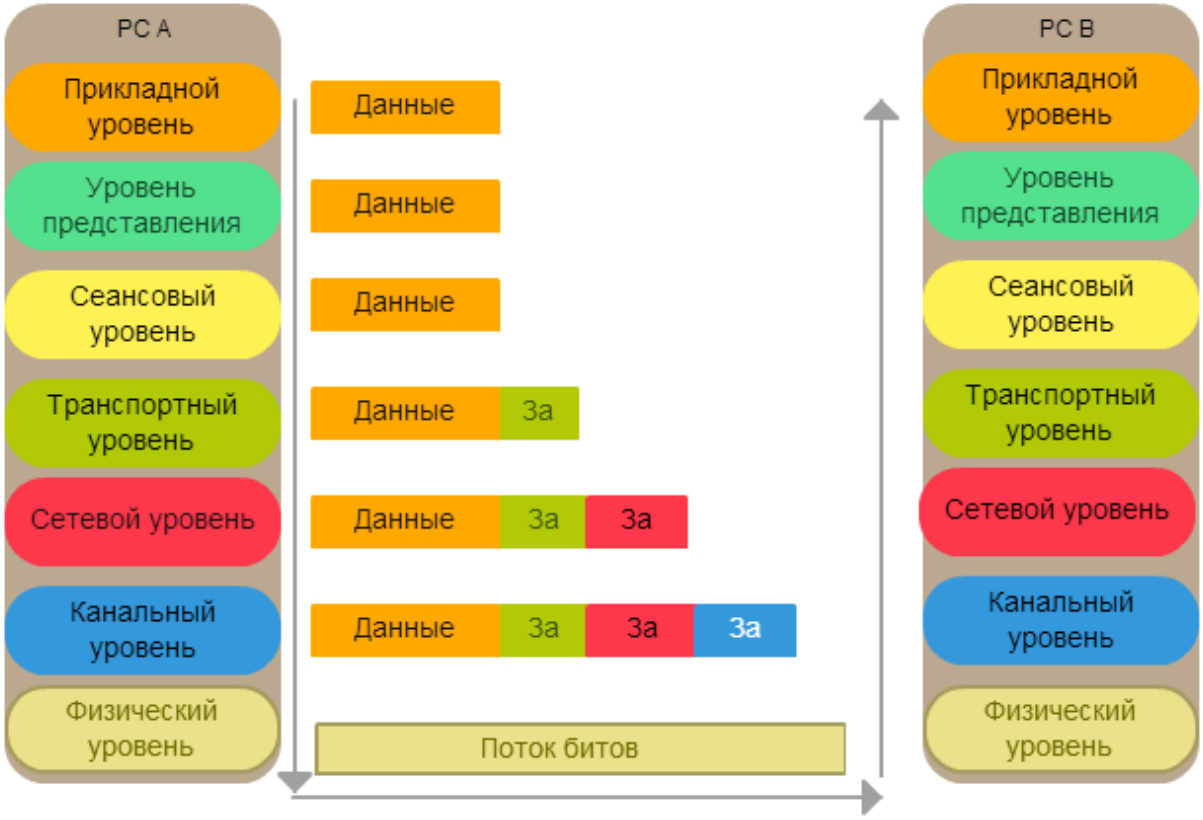

Рисунок 1.2. Инкапсуляция

На канальном уровне пакеты разделяются на еще более маленькие кусочки данных, и к ним помимо опять добавляемого заголовка, только уже канального уровня, добавляется еще FCS. На этом уровне в заголовках содержатся физические адреса устройств – передающего и для кого они предназначаются, а вFCSнаходится вычисленная контрольная сумма, код, который используется для определения целостности данных. Кусочки на канальном уровне называются **кадры**.

На физический уровень кадры передаются уже в виде сигналов битов и следуют через другие сетевые устройства в пункт назначения.

Весь процесс преобразования данных (с верхнего уровня) в сигналы (на нижний уровень) называется **инкапсуляцией.**

Когда сигнал битов на физическом уровне доходит до получателя его сетевая карта принимает биты (на физическом уровне) и преобразует их в кадры (для канального уровня, рисунок 1.3). Канальный уровень в обратной последовательности должен преобразовать кадры в пакеты (для сетевого

уровня), только перед преобразованием уровень сначала смотрит на поле, содержащее MAC-адрес (физический адрес) получателя, он должен совпадать с MAC-адресом сетевой карты, иначе кадр будет уничтожен. Затем канальный уровень (в случае совпадения MAC-адреса) высчитывает сумму полученных данных и сравнивает полученное значение со значением FCS. Значение FCS высчитывалось на исходящем компьютере, а теперь оно, после передачи по физической среде, сравнивается с полученным значением на сервере и если они совпадают, кадр преобразуется в пакет.

На сетевом уровне происходит проверка логического адреса (IP-адреса), в случае успешной проверки пакет преобразуется в сегмент, попадая на транспортный уровень.

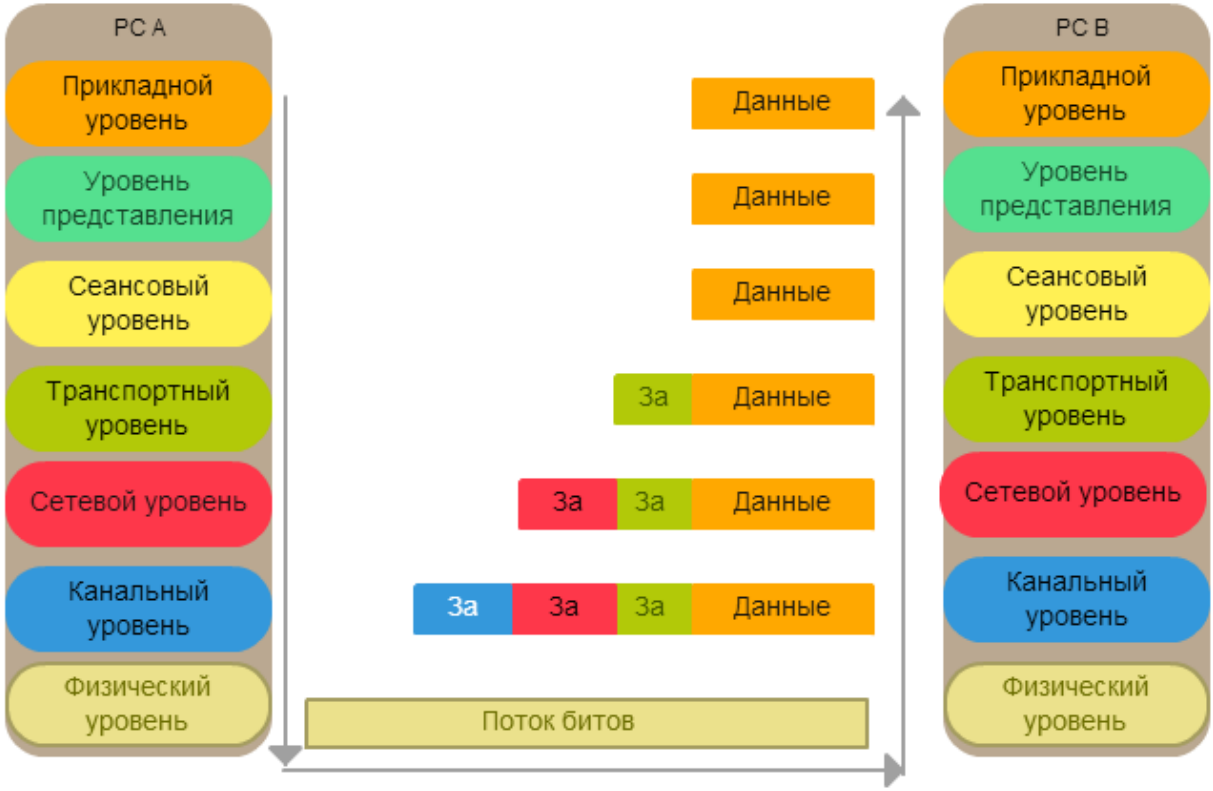

Рисунок 1.3. Декапсуляция

На транспортном уровне проверяется информация из заголовка, что это за сегмент, какой используется протокол, для какого логического порта предназначается и т.п. Протокол использовался TCP, поэтому назад на исходящий компьютер посылается уведомление о прибытии сегмента. Как говорилось выше (когда данные упаковывали в сегмент) в том случае использовался 80 порт назначения. Т.к. на веб-сервере как раз открыт этот порт, данные передаются дальше на верхний уровень.

На верхних уровнях запрос (введенный адрес сайта) обрабатывается веб-сервером (проверяется, доступна-ли запрашиваемая веб-страница).

Этот процесс преобразования сигналов из физической среды в данные называется процессом **декапсуляции.**

1.2.3. Симулятор Cisco Packet Tracer

Данный программный продукт разработан компанией Cisco и рекомендован использоваться при изучении телекоммуникационных сетей и сетевого оборудования. Программа позволяет эмулировать работу различных сетевых устройств: маршрутизаторов, коммутаторов, точек беспроводного доступа, персональных компьютеров, сетевых принтеров, IP-телефонов, настраивать серверы с различными сетевыми услугами и создавать сети практически любой сложности.

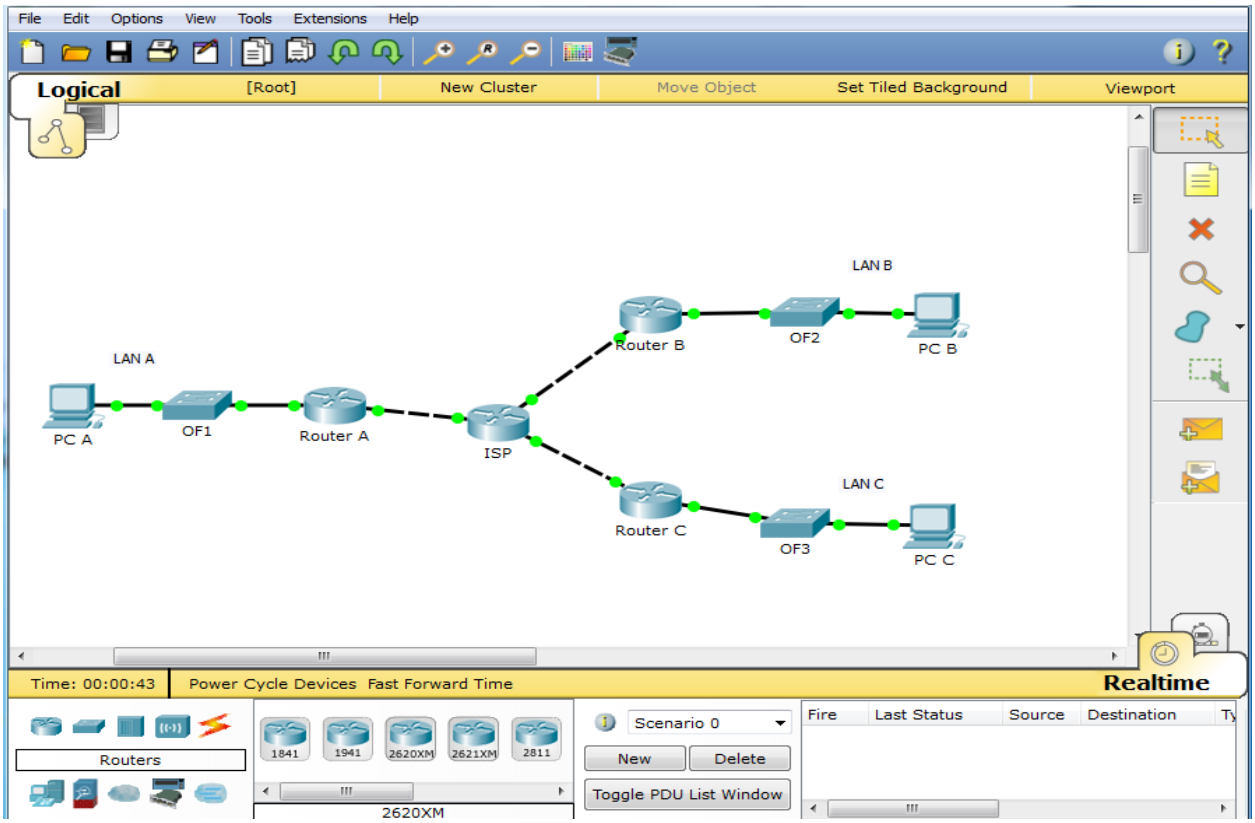

Рисунок 1.4. Интерфейс CPT

Интерфейс Cisco Packet Tracer (рисунок 1.4) достаточно прост и интуитивно понятен. Центральную и большую часть интерфейса занимает рабочая область, предназначенная для построения модели сети, представленной логической топологией.

Рисунок 1.5. Меню выбора устройства

Графическое меню выбора устройств (рисунок 1.5) позволяет добавлять устройства к модели. Все устройства сгруппированы по типам: конечные устройства, маршрутизаторы, коммутаторы, кабели и др.

Симулятор имеет два режима работы: режим реального времени (realtime) и режим симуляции (simulation). Переключение режимов выполняется в правом нижнем углу рабочей области. Режим simulation даёт возможность визуально посмотреть перемещение пакетов от различных протоколов (рисунок 1.6).

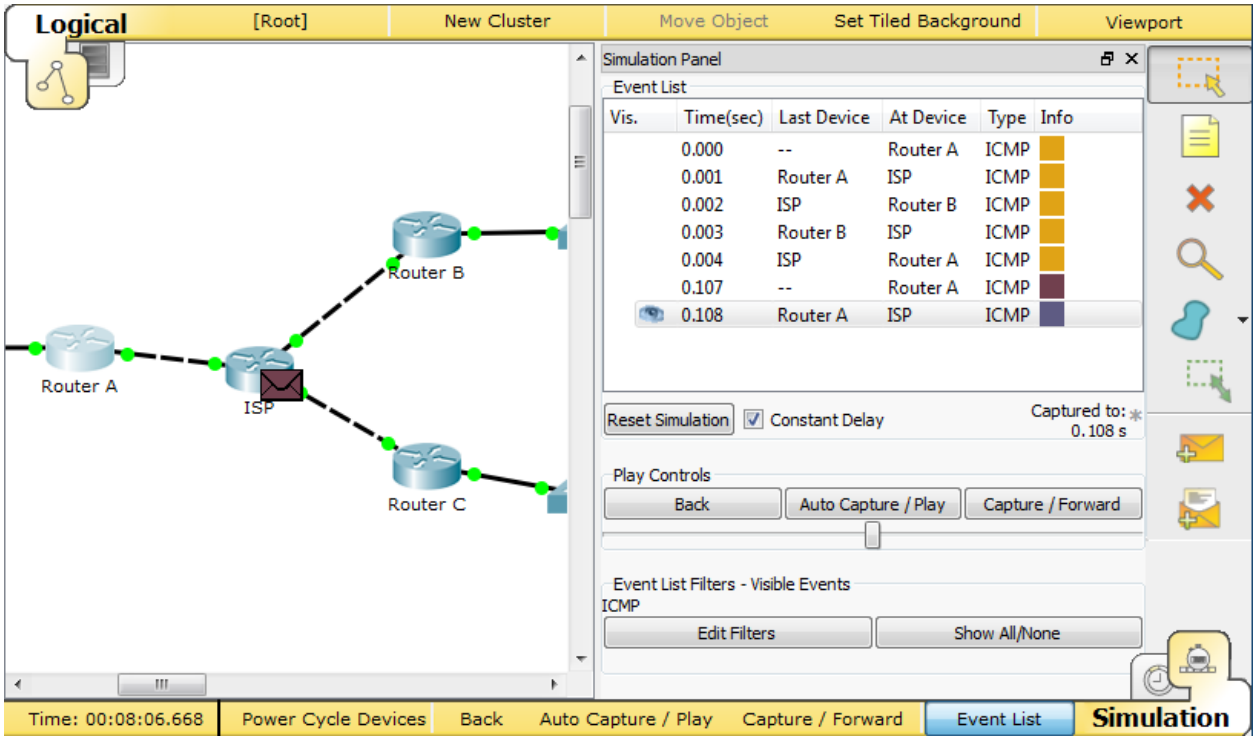

Рисунок 1.6. Режим симуляции

Пакет, который перемещается по сети, можно раскрыть (рисунок 1.7) и посмотреть сведения о текущем состоянии пакета в момент его пребывания на указанном сетевом устройстве. Таблица In Layers показывает сведения о пакете, поступившем на устройство (в данном случае на интерфейс f0/0 маршрутизатора ISP). Таблица Out Layers показывает информацию пакета, который будет отправлен с данного устройства, интерфейс f1/0.

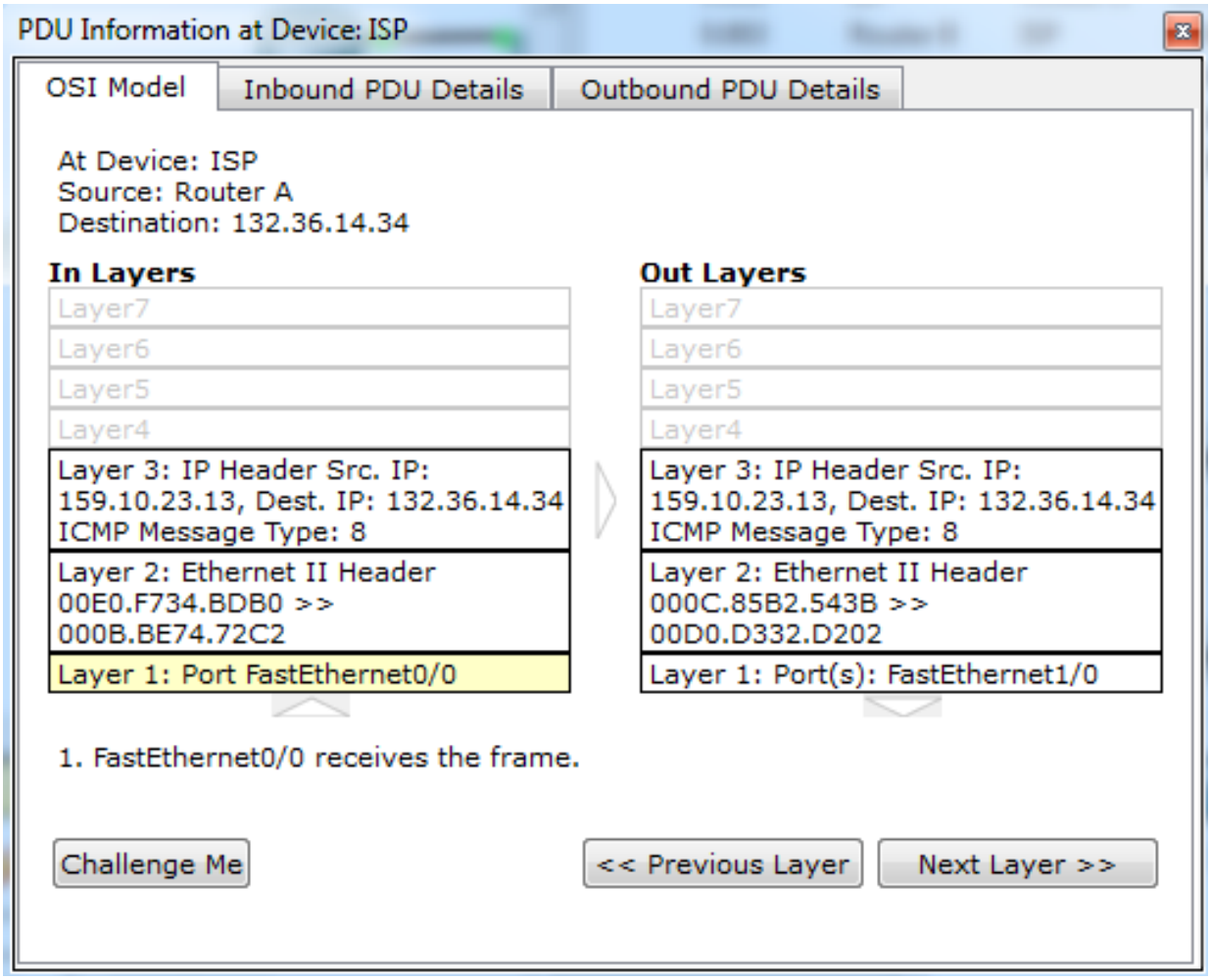

Рисунок 1.7. Содержание пакета

1.2.4. Операционная система Cisco IOS

Cisco IOS — программное обеспечение, используемое в маршрутизаторах Cisco, и некоторых сетевых коммутаторах. Cisco IOS — многозадачная операционная система, выполняющая функции сетевой организации, маршрутизации, коммутации и передачи данных.

Cisco IOS имеет специфичный интерфейс командной строки (command line interface, CLI), который был скопирован многими другими сетевыми продуктами. Интерфейс IOS имеет набор многословных команд, доступные команды определены «режимом» и уровнем привилегий данного пользователя.

Для настройки маршрутизаторов и коммутаторов используется Command line interface, CLI — разновидность текстового интерфейса (CUI) между человеком и компьютером, в котором инструкции компьютеру даются в основном путём ввода с клавиатуры текстовых строк (команд). Интерфейс командной строки использует **иерархическую** структуру для режимов (рис. 1.8).

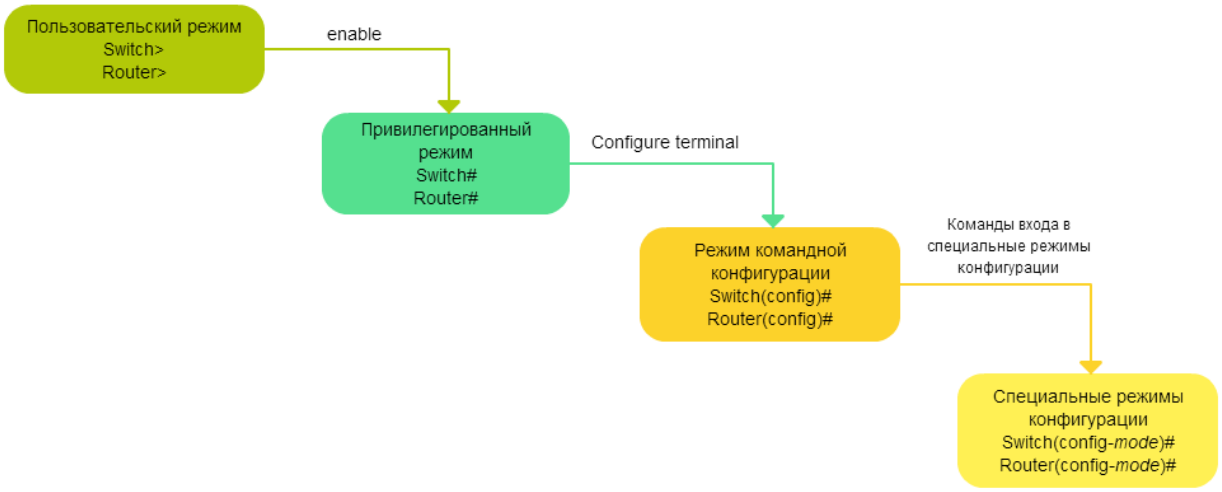

Рисунок 1.8. Иерархия режимов

Пользовательский режим. Это первый режим, в котором пользователь начинает работу при входе в интерфейс командной строки (CLI). Пользовательский режим позволяет выполнять ограниченное количество базовых команд и в нем запрещается выполнять команды, которые могут изменить параметры устройства. Обозначается символом “>”.

```
Router1>
```

Привилегированный режим. В этом режиме в основном доступны команды для мониторинга сети и просмотра конфигурации оборудования. Обозначается символом “#”.

```
Router1#
```

Режим глобальной конфигурации. В этом режиме доступны команды, позволяющие вносить изменения в настройки маршрутизатора или коммутатора.

```
Router1(config)#
```

Другие специальные режимы конфигурации, такие как режим конфигурации интерфейса. В зависимости от режима конфигурации открываются дополнительные команды, такие как настройки интерфейса, динамической маршрутизации, удаленного доступа и другие.

```
Switch(config-if)#
```

Кроме самой IOS – операционной системы – в памяти устройства хранятся файлы конфигурации. Их два: текущий (running-config) и

сохраненный (startup-config). Все изменения в режиме конфигурации вносятся в файл running-config и сразу применяются. При этом файл running-config хранится в энергозависимой оперативной памяти устройства. С одной стороны, это позволяет в любой момент откатиться к последней сохраненной конфигурации обычной перезагрузкой устройства. С другой стороны, незапланированная перезагрузка может сбросить несохраненную конфигурацию устройства. Рекомендуется периодически сохранять рабочую конфигурацию.

1.2.5. Команды IOS

Рассмотрим список команд IOS, необходимый и достаточный для выполнения лабораторной работы № 1. Для следующих лабораторных работ будут добавлены только новые команды, указанные здесь повторяться не будут.

Здесь и далее в <угловых> скобках приводятся обязательные параметры, в [квадратных] скобках – опциональные параметры, в {фигурных} скобках – выбор одного из доступных параметров.

Команды пользовательского режима

`router>`

```
enable
```

Переход в привилегированный режим.

Команды привилегированного режима

`router#`

```
show <параметр>
```

Служит для просмотра различных состояний, элементов и протоколов; что именно просматривается, задается параметром, например:

```
show running-config
```

Выводит текущую конфигурацию оборудования.

```
show interface <интерфейс>
```

Выводит параметры и настройки интерфейса *(например*, `router#show interface fastethernet0/0`*)*.

```
show interfaces
```

Выводит параметры и настройки всех интерфейсов.

```
copy running-config startup-config
```

Сохраняет текущую конфигурацию (перезаписывает сохраненную конфигурацию).

ПРИМЕЧАНИЕ. Текущая конфигурация хранится в энергозависимой памяти. При перезагрузке устройство считывает сохраненную конфигурацию в качестве текущей.

```
configure terminal
```

Переход в режим глобального конфигурирования.

Команды режима глобального конфигурирования
`router(config)#`

```
hostname <имя>
```

Назначает устройству символьное имя.

```
enable password <пароль>
```

Задаёт пароль на привилегированный режим.

```
line console 0
```

Переход в режим конфигурирования консоли управления.

```
line vty 0 15
```

Переход в режим конфигурирования виртуальных терминалов удаленного доступа (vty) с 0 по 15.

ПРИМЕЧАНИЕ. Всего доступно 16 виртуальных терминалов. Допускается настройка и использование меньшего количества, тогда неиспользуемые рекомендуется отключить.

```
interface <интерфейс>
```

Переход в режим конфигурирования указанного интерфейса.

```
service password-encryption
```

Включает шифрование всех паролей в конфигурационном файле.

```
banner motd <баннер>
```

Определяет баннерное сообщение.

Команды режима конфигурирования консоли и vty
`router(config-line)#`

```
password <пароль>
```

Определяет пароль для консольной строки;

```
login
```

Разрешает удаленное подключение и включает аутентификацию по паролю при входе в систему.

Команды режима конфигурирования интерфейса
`router(config-if)#`

```
ip address <IP-адрес> <маска>
```

Задает IP-адрес и маску на интерфейсе.

```
no shutdown
```

Включает интерфейс.

ПРИМЕЧАНИЕ. Интерфейсы коммутаторов по умолчанию включены. Интерфейсы маршрутизаторов по умолчанию выключены.

Команды различных режимов конфигурации

```
exit
```

Возврат на более низкий уровень в иерархии режимов.

```
end
```

Выход их режима конфигурирования в привилегированный режим.

```
no <команда>
```

Отменяет действие выбранной *команды*, удаляет ее из текущей конфигурации устройства.

Cisco IOS имеет дружелюбный пользовательский интерфейс и очень мощную встроенную справочную систему. Символ “?” отображает список команд, доступных в текущем режиме. Можно использовать как в начале строки для отображения всех доступных команд, так и после нескольких символов: без пробела – просмотра списка команд, начинающийся с введенных символов, с пробелом – для просмотра опций введенной команды.

```
Router#sho?
       show

Router#show ?
aaa                 Show AAA values
arp                 Arp table
cdp                 CDP information
...
```

Список команд, доступных в текущем режиме командной строки, конечен. Если существует команда, которая может быть однозначно распознана по начальным символам, ее можно не вводить до конца, ограничившись необходимым количеством символов в начале.

```
Router1>en[ENTER]
Router1#
```

При этом нажатие Tab достраивает неполную команду до полной, если это возможно, без применения этой команды.

```
Router1>en[TAB]
Router1>enable
```

1.3. Задание на лабораторную работу

Лабораторная работа выполняется в среде Cisco Packet Tracer в предложенном Вам файле-сценарии формата pka. Сценарий содержит созданную заранее топологию (рис. 1.9), на которой представлены три локальные сети LANA, LANB и LANC (условные филиалы условной

компании), соединенные между собой через интернет (маршрутизатор ISP). Маршрутизация между сетями уже настроена.

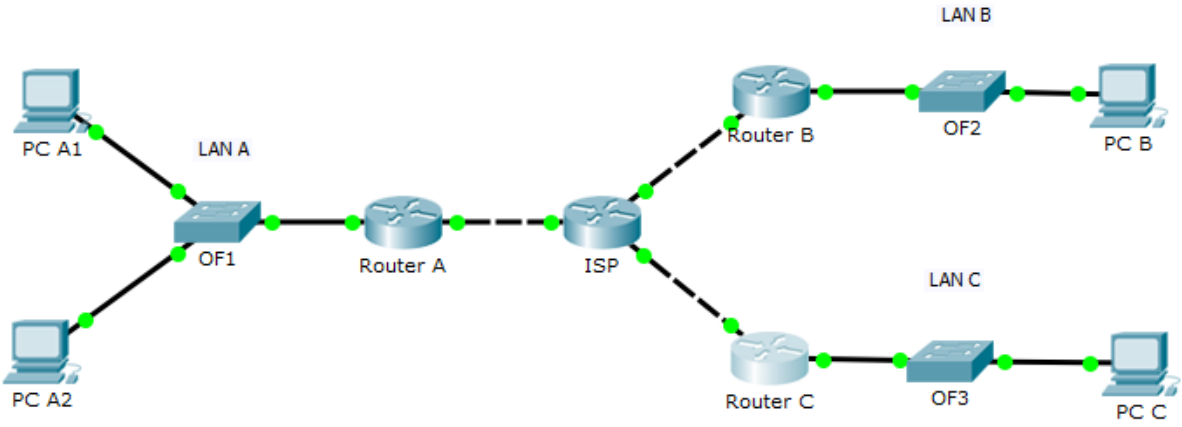

Рисунок 1.9 Топология сети

1.3.1. Расчёт IP-адресов локальных сетей

Рассчитать адреса SUBNET и BROADCAST для сетей LANA, LANB, LANC исходя из известного количества узлов в каждой из них (согласно Вашему варианту), а также известного диапазона адресов для каждой из сетей: (где X – номер Вашего варианта):

- для сети LANA – 10.X.0.0/8;
- для сети LANB – 172.16.X.0/12;
- для сети LANC – 192.168.X.0/16.

Рассчитанные адреса занесите в отчет.

Пример. Рассмотрим вариант 34. В сети LANA должно быть 1300 устройств. Ближайшая сверху к 1300 степень двойки $2048=2^{11}$. Следовательно, 11 бит IP-адреса необходимо отвести под адреса узлов. Значит на адрес сети остается (32-11)=21 бит. Subnet 10.34.0.0/21, broadcast 10.34.7.255

1.3.2. Настройка маршрутизаторов и компьютеров

Выполнить настройку интерфейса Fastethernet0/0 на маршрутизаторе RouterA. Использовать **минимальный** IP адрес из доступных. Пошаговая инструкция по настройке интерфейса:

- подключиться к командной строке маршутизатора;
- зайти в привилегированный режим;
- зайти в режим глобальной конфигурации;
- зайти в режим конфигурации нужного интерфейса;
- включить интерфейс (если необходимо);
- задать на интерфейсе IP-адрес и маску подсети;

– вернуться в привилегированный режим и убедиться, что интерфейс настроен (просмотреть информацию об интерфейсе).

Выполнить настройку компьютеров PCA1 и PCA2 (настроить IP-адрес, маску подсети и шлюз по умолчанию). Использовать **максимальные** IP-адреса из доступных. *Для настройки параметров IP-адресации на компьютере необходимо открыть рабочий стол (вкладка desktop) и запустить приложение IP configuration.*

Пример. Вариант 34, сеть LANA, subnet 10.34.0.0/21, broadcast 10.34.7.255. Выбираем минимальный адрес для маршрутизатора: 10.34.0.1, максимальные для компьютеров: 10.34.7.253, 10.34.7.254. Маска 21 бит: 255.255.248.0. Шлюзом по умолчанию будет интерфейс маршрутизатора в домашней сети: 10.34.0.1.

Повторить описанные действия для сетей LANB и LANC с соответствующими маршрутизаторами и компьютерами. Все адреса устройств занесите в отчет.

1.3.3. Первичная настройка маршрутизаторов

Первичная настройка маршрутизатора выполняется, как правило, при его вводе в эксплуатацию. Оно необходима для обеспечения санкционированного консольного и удаленного (протоколы telnet, secure shell) доступа к командной строке маршрутизатора. Вам необходимо выполнить первоначальную настройку маршрутизаторов, а именно:

– присвоить маршрутизаторам символьные имена (hostname);
– задать пароль для доступа к привилегированному режиму (важно! запишите или запомните этот пароль: в случае его утраты работу придется делать заново);
– настроить пароль доступа к консоли маршрутизатора;
– настроить удаленный доступ (виртуальный терминал);
– включить шифрование всех паролей;
– добавить баннер.

1.3.4. Анализ пакета в режиме симуляции

Сети настроены. Посмотрим, как передаются пакеты между узлами одной сети, а также из одной сети в другую.

Перейти в режим simulation.

Ввести в терминальной строке (приложение command prompt) на компьютере PCA1 команду **ping** с IP-адресом компьютера PCA2 и проследить

за прохождением пакета по сети, изменением адресов в заголовках сетевого и канального уровней.

Повторить анализ для пакета, отправленного с PCA1 на PCB.

Полученные данные и выводы занесите в отчет.

1.4. Контрольные вопросы

1. Что такое протокол и интерфейс?
2. Что представляет собой IP-адрес? Какова его структура?
3. Чем отличаются адреса unicast, multicast, broadcast?
4. В чем отличие классовой адресации от бесклассовой?
5. Что такое маска подсети? Для чего она используется?
6. Каковы структура и размер сети с маской 20 бит?
7. Что такое subnet и broadcast?
8. Что такое шлюз по умолчанию? Для чего он используется? Каков адрес шлюза по умолчанию для сети LANA?
9. Какие режимы работы существуют в Cisco IOS? Для чего используется каждый из них? Как осуществляются переходы между режимами?
10. В каком виде хранится конфигурация маршрутизаторов Cisco?
11. Какой тип кабеля используется при подключении RouterA к RouterB? RouterC к OF3?
12. Как меняются IP и MAC-адреса источника и назначения в заголовках пакета при его прохождении от PCA1 к PCB? От PCA1 к PCA2?
13. Какие устройства могут изменять MAC и IP адреса источника и назначения в заголовках пакетов и кадров? Для чего и в каких случаях?
14. Какие IP и MAC адрес на интерфейсе fa1/0 маршрутизатора RouterA? Какие команды служат для их просмотра и редактирования?

1.5. Варианты индивидуальных заданий

Номер варианта	Количество узлов в сети		
	LAN A	LAN B	LAN C
1	1026	76	12
2	965	23	65
3	247	12	127
4	1654	38	59
5	179	7	89

Номер варианта	Количество узлов в сети		
	LAN A	LAN B	LAN C
6	256	43	67
7	832	15	48
8	653	63	167
9	263	128	250
10	537	72	129
11	1634	138	197
12	456	8	81
13	2350	104	114
14	3087	311	79
15	4211	29	156

1.6. Форма отчета

Отчет о выполнении лабораторной работы оформляется строго в соответствии с индивидуальным вариантом задания и является обязательным требованием для допуска к защите наряду с правильно настроенным сценарием работы в программе Packet Tracer.

Отчет должен включать титульный лист, схему сети, а также заполненные таблицы, приведенные ниже.

1.6.1. Расчет IP-адресов

Параметр	LAN A	LAN B	LAN C
Количество узлов			
Ближайшая сверху степень двойки			
Маска (префиксная)			
Маска (десятичная)			
SUBNET			
HOSTMIN			
HOSTMAX			

Параметр	LAN A	LAN B	LAN C
BROADCAST			

1.6.2. Сведения о конфигурации устройств

Устройство	Интерфейс	IP-адрес (если есть)	Маска подсети	Основной шлюз
PC A1	NIC			
PC A2	NIC			
PC B	NIC			
PC C	NIC			
OF1	Fa0/1			
	Fa0/2			
OF2	Fa0/1			
	Fa0/2			
OF3	Fa0/1			
	Fa0/2			
RouterA	Fa0/0.___			
	Fa0/1			
RouterB	Fa0/0.___			
	Fa0/1			
RouterC	Fa0/0.___			
	Fa0/1			

1.6.3. Анализ заголовков пакета PC A1 – PC A2

На устройстве	IP-адрес		MAC-адрес	
	Source	Destination	Source	Destination
PCA1				
OF1				
PCA2				

1.6.4. Анализ заголовков пакета PCA1 – PC B

На устройстве	IP-адрес		MAC-адрес	
	Source	Destination	Source	Destination
PCA1				
OF1				

На устройстве	IP-адрес		MAC-адрес	
	Source	**Destination**	**Source**	**Destination**
Router A (in)				
Router A (out)				
ISP (in)				
ISP (out)				
Router B (in)				
Router B (out)				
OF2				
PC B				