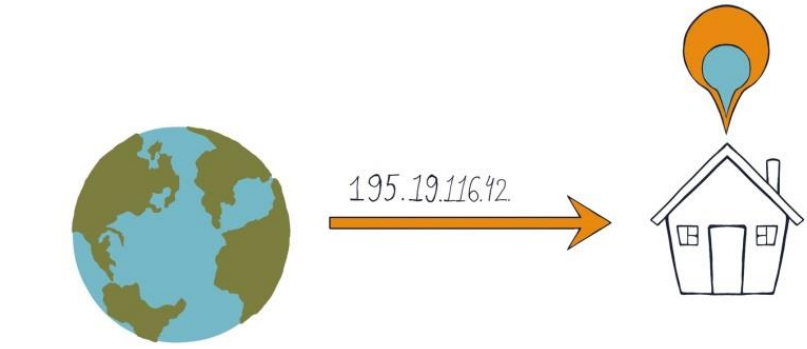
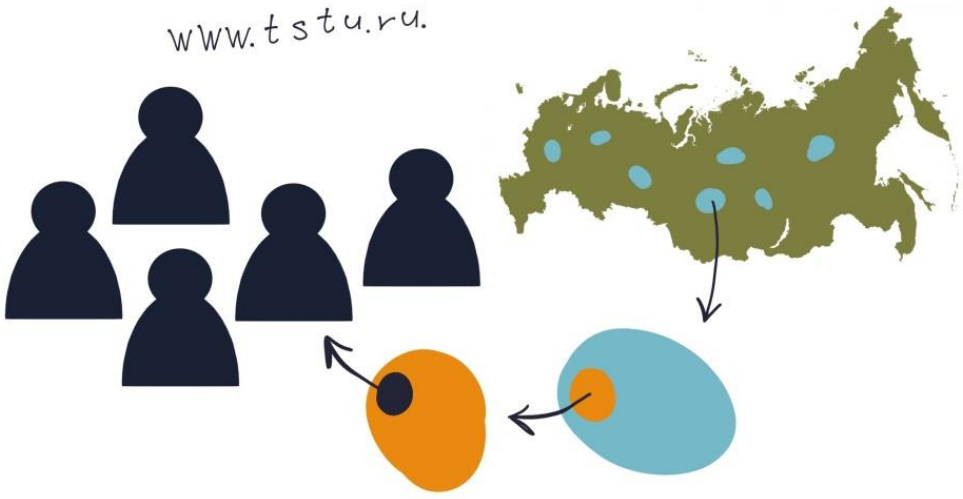
В. В. КОНКИНА, А. Б. БОРИСЕНКО, И. Л. КОРОБОВА

СЕТИ И ТЕЛЕКОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ





Тамбов
Издательский центр ФГБОУ ВО «ТГТУ»
2023

Министерство науки и высшего образования Российской Федерации

**Федеральное государственное бюджетное
образовательное учреждение высшего образования
«Тамбовский государственный технический университет»**

В. В. КОНКИНА, А. Б. БОРИСЕНКО, И. Л. КОРОБОВА

СЕТИ И ТЕЛЕКОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ

Утверждено Ученым советом университета в качестве учебного пособия
для студентов 3 курса направления подготовки
09.03.01 «Информатика и вычислительная техника»,
изучающих дисциплину «Сети и телекоммуникации»,
очной и заочной форм обучения

Учебное электронное издание

**Тамбов
Издательский центр ФГБОУ ВО «ТГТУ»
2023**

УДК 004.73;004.057.4
ББК 32.971.3
К64

Рецензенты:

Кандидат технических наук, доцент, доцент кафедры
«Математическое моделирование и информационные технологии»
ФГБОУ ВО «ТГУ им. Г. Р. Державина»
Д. С. Соловьев

Кандидат технических наук, доцент, доцент кафедры
«Информационные процессы и управление»
ФГБОУ ВО «ТГТУ»
А. А. Третьяков

К64 **Конкина, В. В.**
Сети и телекоммуникационные технологии [Электронный ресурс] : учебное пособие / В. В. Конкина, А. Б. Борисенко, И. Л. Коробова. – Тамбов : Издательский центр ФГБОУ ВО «ТГТУ», 2023. – 1 электрон. опт. диск (CD-ROM). – Системные требования : ПК не ниже Pentium IV ; CD-ROM-дисковод ; 3,2 Мб ; RAM ; Windows 95/98/XP ; мышь. – Загл. с экрана.
ISBN 978-5-8265-2632-3

Рассмотрена техническая структура и архитектура компьютерной сети. Подробно описываются разные уровни сетевой модели. Приведена система доменных имен.

Предназначено для студентов 3 курса направления подготовки 09.03.01 «Информатика и вычислительная техника», изучающих дисциплину «Сети и телекоммуникации», очной и заочной форм обучения.

УДК 004.73;004.057.4
ББК 32.971.3

ISBN 978-5-8265-2632-3

Первопроходцы перевозили тяжести на быках.
И если бык не мог сдвинуть бревно, они не пытались
вырастить быка побольше. Мы должны стремиться
не к повышению мощности отдельных компьютеров,
а к повышению численности компьютерных систем

Грейс Хоппер[*]

[*] Грейс Хоппер (9 декабря 1906 – 1 января 1992) – американская ученая и коммодор флота США. Будучи первооткрывательницей в своей области, она была одной из первых, кто писал программы для гарвардского компьютера Марк I. Она разработала первый компилятор для компьютерного языка программирования, развила концепцию машинно-независимых языков программирования, что привело к созданию COBOL, одного из первых высокоуровневых языков программирования. Ей приписывается популяризация термина debugging для устранения сбоев в работе компьютера.

ВВЕДЕНИЕ

На первый взгляд использование сети Интернет выглядит крайне просто. Мы переходим на какой-либо веб-адрес, и открывается нужная страница. Или же мы заходим в нашу любимую социальную сеть и просматриваем фотографии друзей, семейные фото или снимки с домашними животными. Однако за кажущейся простотой скрывается большое количество сложного программного и аппаратного обеспечения. Разработка технологий, на основе которых работает современный Интернет, началась в 1960-х годах. С тех пор благодаря исследованиям и разработкам в области сетевых технологий сети стали больше, быстрее и теперь могут связывать миллиарды устройств по всему миру. С целью лучше понять, как работает Интернет сегодня, рассмотрим, с помощью каких технологий осуществлялась коммуникация людей и устройств на протяжении многих лет.

1. ОБЩИЕ СВЕДЕНИЯ О СЕТЕВЫХ ТЕХНОЛОГИЯХ

1.1. ОСНОВНЫЕ ОПРЕДЕЛЕНИЯ

Сеть передачи данных – совокупность трех и более оконечных устройств связи, объединенных каналами передачи данных и коммутирующими устройствами (узлами сети), обеспечивающими обмен сообщениями между всеми оконечными устройствами.

Передача данных – физический перенос данных в виде сигналов от точки к точке или от точки к нескольким точкам средствами связи по каналу передачи данных. Примерами подобных каналов могут служить медные провода, волокно-оптические линии связи, беспроводные каналы передачи.

Сетевая инфраструктура включает в себя три категории компонентов сети:

1. Устройства.
2. Среда передачи.
3. Сервисы.

Устройства и среды передачи – это физические элементы или аппаратное обеспечение сети. Аппаратное обеспечение зачастую является видимой частью сетевой платформы: ноутбук, ПК, коммутатор, маршрутизатор, беспроводная точка доступа или кабели, используемые для соединения устройств.

Оконечное устройство является либо отправителем (источником), либо получателем (адресатом) сообщения. Каждому оконечному устройству в сети назначается адрес, чтобы устройства можно было отличить от других. Если оконечное устройство инициирует обмен данными, то в качестве получателя сообщения оно использует адрес оконечного устройства назначения.

Примерами оконечных устройств могут служить:

1) настольные персональные компьютеры;
2) ноутбук;

3) принтер;

4) Ip-телефон;

5) беспроводной планшетный компьютер;

6) смартфон.

Промежуточные устройства соединяют отдельные оконечные устройства с сетью и могут соединять несколько отдельных сетей для создания глобальных сетей. Такие устройства обеспечивают подключение и прохождение потоков данных по сети. Для определения пути передачи сообщения промежуточные устройства используют адрес оконечного устройства назначения в сочетании с информацией о связях в сети.

Примерами промежуточных устройств могут служить:

1) коммутатор;

2) маршрутизатор;

3) межсетевой экран;

4) репитер.

Среда передачи данных – физический канал, по которому сообщение передается от источника к адресату.

Типы физических сред передачи данных:

1) медный кабель;

2) оптоволоконный кабель;

3) беспроводная связь.

Сетевая топология – граф, вершинами которого являются оконечные и промежуточные устройства, а ребрами – физические и информационные связи между вершинами. Схема обеспечивает наглядный способ понимания, каким образом устройства в большой сети связаны между собой.

Подразделяется на несколько типов:

1. **Физическая топология** – отображает физическое расположение промежуточных устройств и кабельных линий.

2. **Логическая топология** – отображает устройства, порты и схемы адресации. Изображения топологий приведены на рис. 1.1, 1.2.

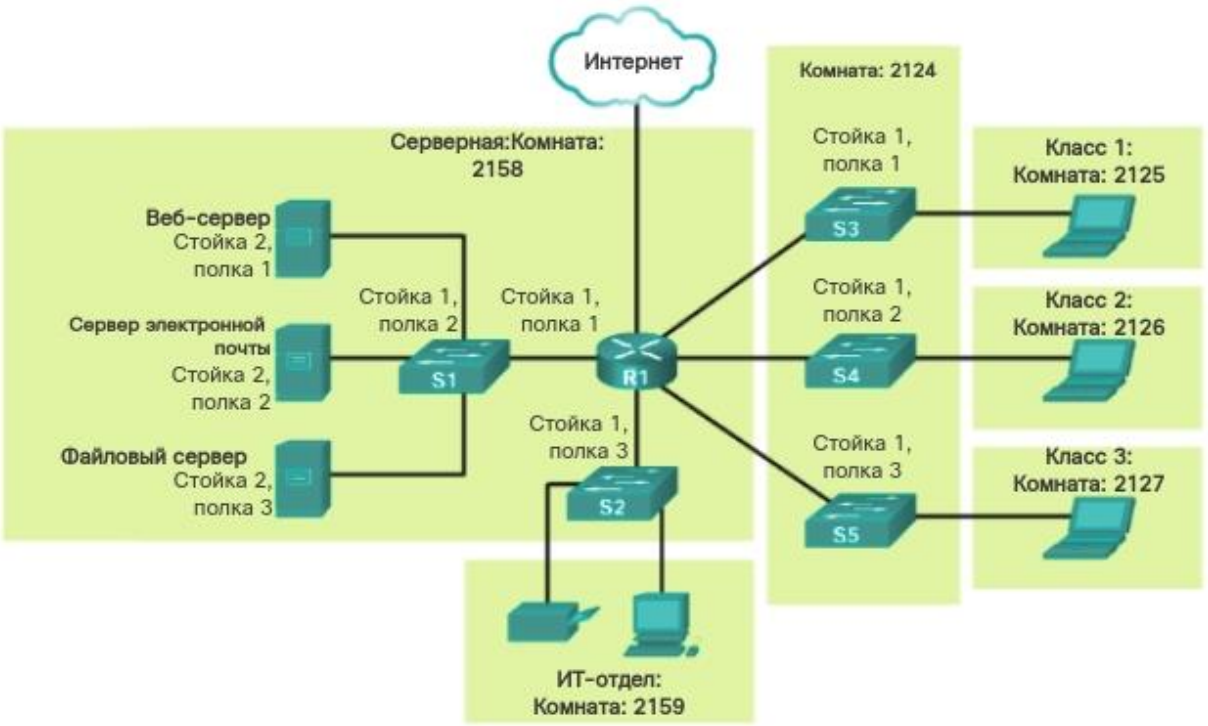


Рис. 1.1. Пример физической топологии

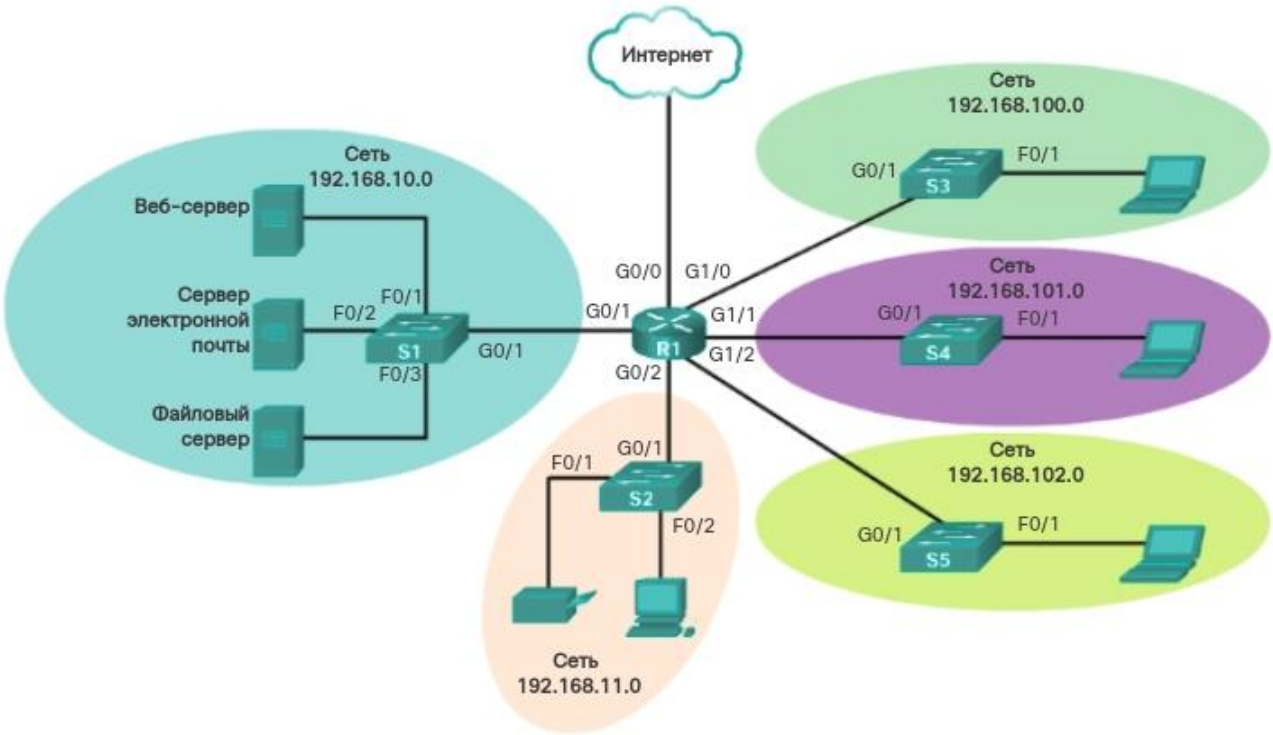


Рис. 1.2. Пример логической топологии

Сетевая карта – устройство, позволяющее взаимодействовать с другими устройствами в сети.

Физический порт – разъем на сетевом устройстве, через который кабели подключены к компьютеру или другому сетевому устройству.

Интерфейс – специализированные порты в сетевом устройстве, которые подключаются к отдельным сетям. Поскольку маршрутизаторы соединяют между собой сети, порты маршрутизатора называются сетевыми интерфейсами.

Часто на практике слова «Порт» и «Интерфейс» являются взаимозаменяемыми.

1.2. ВИДЫ СЕТЕЙ

Сети сильно отличаются по площади покрытия, количеству пользователей, типу и объему предоставляемых услуг пользователям. Наиболее распространенными типами сетевых инфраструктур являются локальные сети LAN и глобальные сети WAN.

Локальная сеть (LAN) – сетевая инфраструктура, предоставляющая высокоскоростной доступ пользователям и оконечным устройствам на небольшой территории. Обычно является домашней сетью, сетью малого или крупного предприятия. Управляется одним квалифицированным лицом или отдельным IT-отделом на предприятии.

Глобальная сеть (WAN) – сетевая инфраструктура, предоставляющая доступ к другим сетям на большой территории. Принадлежит провайдерам телекоммуникационных услуг и находится под их управлением. Интернет – всемирное объединение взаимосвязанных сетей для хранения и передачи информации.

Экстранет – защищенная от несанкционированного доступа корпоративная сеть, использующая интернет-технологии для внутрикорпоративных целей, а также для предоставления части корпоративной информации и корпоративных приложений деловым партнерам компании.

Интранет – частные сети LAN и WAN, которые принадлежат организации и доступны только ее членам, сотрудникам и прочим авторизованным лицам.

Для сети Экстранет особенно важны аутентификация пользователя (который может и не являться сотрудником компании) и, особенно, защита

от несанкционированного доступа, тогда как для приложений Интернет они играют гораздо менее существенную роль, поскольку доступ к этой сети ограничен физическими рамками компании.

Для доступа к Интернет существует множество способов подключения. Домашние пользователи, удаленные сотрудники компаний и малые офисы, как правило, для доступа в Интернет нуждаются в подключении к поставщикам услуг Интернета. Варианты подключения существенно меняются в зависимости от провайдера, географического местоположения и развития инфраструктуры. Популярные варианты включают в себя широкополосную кабельную сеть, широкополосную цифровую абонентскую линию (DSL), беспроводные глобальные сети и мобильные сервисы.

1.3. НАДЕЖНОСТЬ СЕТЕЙ

Для поддержания работоспособности и надежности сети требуется, чтобы она соответствовала четырем основным требованиям:

1. Отказоустойчивость.
2. Масштабируемость.
3. Качество обслуживания.
4. Безопасность.

Отказоустойчивость – свойство сети сохранять свою работоспособность после отказа одного или нескольких составных компонентов. Для этого сети используют несколько путей передачи данных от источника к месту назначения. Если один путь недоступен, сообщения можно немедленно отправить по другой линии связи. Наличие нескольких путей к месту назначения называется резервированием.

Масштабируемость – свойство сети, позволяющая быстро расширить, обеспечив поддержку новых пользователей и приложений без снижения эффективности обслуживания существующих.

Качество обслуживания (QoS – Quality of Service) – технология предоставления различным классам трафика различных приоритетов в обслуживании во избежание перегрузки сети.

Обеспечение **безопасности** инфраструктуры сети включает в себя физическую защиту всех устройств, которые необходимы для сетевых подключений, и предотвращение несанкционированного доступа к установленному на них ПО управления.

Безопасность информации означает защиту пакетов данных, передаваемых по сети, а также информации, хранящейся на подключенных к сети устройствах.

Критерии безопасности:

1. Конфиденциальность – только указанные и авторизованные получатели могут иметь доступ к данным.
2. Целостность – гарантия того, что информация не была изменена в процессе передачи от исходного пункта к месту назначения.
3. Доступность – своевременный и надежный доступ к данным для авторизованных пользователей.

1.4. КОММУНИКАЦИЯ И ПРОТОКОЛЫ

Коммуникация – тип взаимодействия между объектами, который подразумевает обмен информацией между этими объектами. Все способы коммуникаций имеют три общих элемента. Первый – это источник сообщения или отправитель. Второй элемент – это адресат или получатель сообщения. Адресат получает и интерпретирует сообщение. Третий элемент, называемый каналом, представляет собой среду передачи данных, по которой сообщение передается от источника к получателю.

В сетях существует несколько способов передачи данных:

1. Индивидуальная (Unicast).
2. Групповая (Multicast).
3. Широковещательная (Broadcast).

Unicast подразумевает собой передачу данных одному единственному адресату в сети. При передаче данных способом **Multicast** данные получают одновременно несколько адресатов в сети. **Broadcast** означает, что данные получат все узлы в сети за исключением того, кто информацию и передает.

Сетевые протоколы определяют общий формат и набор правил для обмена сообщениями между устройствами. Набор протоколов представляет собой множество протоколов, которые используются вместе для предоставления комплексных сетевых сервисов. Набор протоколов может быть определен организацией по стандартизации или разработан производителем сетевого оборудования.

К примеру, набор протоколов TCP/IP является открытым стандартом. Данные протоколы находятся в свободном доступе, и любой разработчик может использовать эти протоколы в аппаратном или программном обеспечении. Каждый стандартный протокол принят отраслевыми компаниями и утвержден организацией по стандартизации. Использование стандартов в разработке и реализации протоколов гарантирует, что продукты от разных производителей будут успешно взаимодействовать между собой.

Открытые стандарты способствуют совместимости, конкуренции и инновациям. Кроме того, они гарантируют, что продукт отдельной компании не сможет монополизировать рынок или получить несправедливое преимущество по сравнению с конкурентами. Пример – покупка беспроводного маршрутизатора для дома. Существует множество вариантов маршрутизаторов различных производителей, каждый из которых включает стандартные протоколы, такие как IPv4, DHCP, 802.3 (Ethernet) и 802.11 (беспроводная сеть LAN). Открытые стандарты также позволяют клиенту с операционной системой OS X от компании Apple загрузить веб-страницу с веб-сервера под управлением GNU/Linux. Это связано с тем, что обе операционные системы используют протоколы открытых стандартов, например из набора протоколов TCP/IP.

Организации по стандартизации обычно являются независимыми от поставщиков некоммерческими организациями, созданными для разработки и продвижения концепции открытых стандартов. Некоторые протоколы

являются **проприетарными**. Это означает, что описание протокола и принципы его работы определяются одной конкретной компанией или поставщиком. Примерами частных протоколов являются устаревшие наборы протоколов AppleTalk и Novell Netware. Нередко поставщик (или группа поставщиков) разрабатывает частный протокол для удовлетворения потребностей своих заказчиков, а затем способствует принятию этого частного протокола в качестве открытого стандарта.

1.5. ЭТАЛОННАЯ МОДЕЛЬ СЕТИ OSI

Чтобы представить взаимодействие между различными протоколами, принято использовать многоуровневые модели. Многоуровневая модель изображает работу протоколов, происходящую внутри каждого уровня, а также взаимодействие с уровнями выше и ниже.

Есть ряд преимуществ в использовании многоуровневой модели для описания сетевых протоколов и операций.

Преимущества в использовании многоуровневой моделью:

1. Упрощение разработки протоколов, поскольку протоколы, работающие на определенном уровне, определяют формат обрабатываемых данных и интерфейс верхних и нижних уровней.
2. Стимулирование конкуренции, так как продукты разных поставщиков могут взаимодействовать друг с другом.
3. Предотвращение влияния изменений технологий или функций одного уровня на другие уровни (верхние и нижние).
4. Общий язык для описания функций сетевого взаимодействия.

Эталонная модель OSI определяет широкий список функций и сервисов, реализуемых на каждом уровне, с ее помощью можно описать работу любой системы связей. Для обслуживания абонентов любой системы связей необходимо последовательно решить семь задач. В таблице 1.1 представлены уровни стека модели OSI с указанием единицы данных, с которым работает каждый из уровней.

1.1. Эталонная модель стека OSI

Уровень модели OSI	Единица данных
7. Прикладной	Данные
6. Представления	
5. Сеансовый	
4. Транспортный	Сегменты
3. Сетевой	Пакеты
2. Канальный	Кадры
1. Физический	Биты

Описание каждого уровня:

7. *Прикладной уровень* содержит протоколы для обмена данными между приложениями. Уровень решает задачи предоставления услуг, под услугами понимают передачу голоса, видео, электронной почты.

6. *Уровень представления* обеспечивает общее представление данных, передаваемых между службами прикладного уровня. Уровень решает задачи перевода информации из вида, понятного человеку, в вид, понятный машине и наоборот. То есть с одной стороны находится информация в виде, например, голоса, а с другой стороны – некая закодированная комбинация, которая в дальнейшем будет упаковываться в пакеты и передаваться по сети.

5. *Сеансовый уровень* передает сервисы на уровень представления для организации его диалога и управления обмена данными. Здесь решается задача установления соединения между абонентом А и абонентом В, т.е., например, задача соединения доменного имени с его IP-адресом.

4. *Транспортный уровень* определяет сервисы для сегментации, передачи и сборки данных для отдельных коммуникаций между оконечными устройствами. Здесь решается задача обеспечения надежной среды передачи данных. Сети состоят из беспроводных каналов связи на одних участках, на других участках используется оптика, на третьих участках находится медный канал связи. Естественно, каждая из этих линий связей обладает своей надежностью,

с одной стороны, а с другой стороны, сервисы предъявляют определенные требования по надежности трафика, т.е. по задержке, по вероятностным потерям пакета и т.д. Для того, чтобы из этого набора каналов сделать надежный тоннель передачи информации, используются протоколы транспортного уровня.

3. *Сетевой уровень* представляет функции для обмена отдельными частями данных по сети между указанными оконечными устройствами. Сети состоят из большого количества узлов, между которыми находятся различные каналы связи, которые обладают различными характеристиками и параметрами. На сетевом уровне стоит задача выбора наиболее оптимального, целесообразного маршрута передачи информации между абонентами А и В по некоторому критерию. В качестве критерия используются протяженность канала, т.е. расстояние между узлами, пропускная способность либо стоимость доставки информации.

2. К*анальный уровень* описывает способы обмена кадрами данных при обмене данными между устройствами по общей среде передачи данных. Здесь решаются задачи распределения канального ресурса, предотвращения коллизий. Под канальным ресурсом понимается частотный ресурс, временной, кодовый. Различают детерминированный доступ к середе передачи, т.е. доступ к ресурсу заранее распределен, и вероятностный доступ.

1. *Физический уровень* описывает электрические, механические, функциональные и процедурные средства для активации, поддержки, деактивации физического соединения, обеспечивающего передачу битов из одного сетевого устройства в другое. Здесь решается задача непосредственной передачи сигналов, т.е. в процессе передачи сигнал может модулироваться, могут применяться средства помехозащищенности. На этом уровне происходят физические процессы передачи информации.

1.6. МОДЕЛЬ СТЕКА ПРОТОКОЛОВ TCP/IP

Протокольная модель сетевого взаимодействия TCP/IP была создана в начале 1970-х годов и иногда называется моделью сети Интернет. Как показано в табл. 1.2, такая модель определяет четыре категории функций, необходи-

мых для успешного взаимодействия. Архитектура набора протоколов TCP/IP построена на основе этой модели. TCP/IP представляет собой открытый стандарт, ни одна компания не вправе контролировать ее определение.

Описание каждого уровня:

1. *Уровень приложений* отображает данные для пользователя, а также обеспечивает кодирование и управление сеансами связи. К протоколам этого уровня относятся протокол передачи гипертекста HTTP, почтовые протоколы SMTP, POP3, IMAP, протокол передачи файлов FTP.

2. *Транспортный уровень* поддерживает связь между различными устройствами в разных сетях. Наибольшее распространение на этом уровне получили протоколы UDP и TCP. Протокол UDP обеспечивает большую скорость передачи трафика, но не гарантирует доставку, а протокол TCP гарантирует доставку, но скорость передачи будет ниже. Поэтому для передачи трафика реального времени, например видео, аудио, используют UDP.

3. *Уровень межсетевого взаимодействия* определяет наилучший путь через сеть. Для назначения адресов узлам сети используются протоколы IPv4, IPv6. Для построения маршрута применяются протоколы RIP, OSPF, BGP.

4. *Уровень канальный* управляет устройствами и средами передачи данных, из которых состоит сеть. Здесь применяются такие протоколы, как IEEE: 802.3 (Ethernet), 802.11 (Wi-Fi), 802.15 (Bluetooth). На этот уровень можно отнести физическое подключение к сети с помощью различных интерфейсов, таких как RJ-45, SFP, стандарты кабелей UTP/STP (витая пара).

В таблице 1.3 показано сопоставление моделей OSI и TCP/IP.

1.2. Модель стека протоколов TCP/IP

Уровень модели TCP/IP	Единица данных
4. Приложений	Данные
3. Транспортный	Сегменты
2. Межсетевого взаимодействия	Пакеты
1. Канальный	Биты/Кадры

1.3. Сопоставление моделей OSI и TCP/IP

Уровень модели OSI	Уровень модели TCP/IP
7. Прикладной	Приложений
6. Представления	
5. Сеансовый	
4. Транспортный	Транспортный
3. Сетевой	Межсетевого взаимодействия
2. Канальный	Канальный
1. Физический	

На уровне доступа к сети набор протоколов TCP/IP не определяет список протоколов, используемых при работе со средой передачи данных, он описывает только передачу информации с сетевого уровня физическим сетевым протоколам. Уровни 1 и 2 модели OSI описывают процедуры доступа к среде передачи и физическим способам отправки данных по сети.

Уровень 3 модели OSI, или сетевой уровень, соответствует сетевому уровню модели TCP/ IP. Этот уровень описывает протоколы, определяющие пути передачи данных в сети.

Уровень 4 модели OSI, или транспортный уровень, соответствует транспортному уровню модели TCP/IP. Этот уровень описывает общие сервисы и функции, которые обеспечивают упорядоченную и надежную доставку данных от источника до места назначения.

Уровень приложений TCP/IP включает в себя ряд протоколов, которые поддерживают определенные функции для работы разнообразных приложений конечных пользователей. Уровни 5, 6 и 7 модели OSI используются в качестве

образцов разработчиками и поставщиками прикладного программного обеспечения для производства продуктов, предназначенных для работы в сети.

Обе модели (TCP/IP и OSI) широко применяются в отношении протоколов различных уровней. Так как модель OSI разделяет канальный и физический уровни, именно она используется для этих уровней.

1.7. СТРУКТУРА IP-АДРЕСА

В качестве примера рассмотрим IP-адрес вида: 192.168.0.1 с маской 255.255.255.0. Маска нужна для того, чтобы отделить часть, которая отвечает за адрес сети от той части, которая отвечает за номер хоста. Хост – это любое сетевое устройство, это компьютер, IP-камера, сервер. В рассматриваемом примере за номер сети отвечает часть «192.168.0», за номер хоста отвечает та часть, где находится «.1» (рис. 1.3).

Во время распределения пула IP-адресов между узлами-участниками сети необходимо соблюдать следующие правила:

1. у всех хостов сети должна быть одинаковой та часть IP-адреса, которая относится к номеру сети;
2. у всех хостов та часть, которая относится к номеру хоста, должна быть разной;
3. у всех хостов должна быть одинаковая маска по сети.

Максимальное количество хостов в сети определяется маской. В рассматриваемом примере максимальное количество хостов равняется 254.

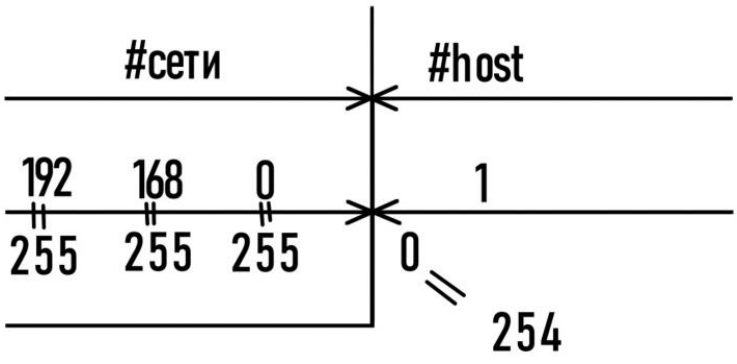


Рис. 1.3. Структура IP-адреса

1.8. ТИПЫ IP-АДРЕСОВ

Рассмотрим типы IP-адресов, которые встречаются в одной сети. Маска 255.255.255.0 показывает, что в данной сети может быть максимум 256 IP-адресов. В любой сети самый первый IP-адрес является номером сети, используется для работы маршрутизатора, последний IP-адрес – это адрес широковещательной (broadcast) рассылки, т.е. пакет получают все хосты сети, broadcast рассылка используется для работы различных протоколов. Со второго по предпоследнего IP-адрес используется для нумерации хостов.

Для IP-адреса

192.168.0.1

с маской

255.255.255.0

первым IP-адресом будет номер сети

192.168.0.0,

последний IP-адрес будет адресом широковещательной рассылки

192.168.0.255.

Таким образом, в качестве IP-адресов хостов будет выступать пул

192.168.0.1 – 192.168.0. 254.

Как правило, первый или последний из хостовых IP-адресов используется для назначения его маршрутизатору или шлюзу по умолчанию в данной локальной сети.

1.9. КЛАССЫ IP-СЕТЕЙ

В настоящее время используется как классовая (табл. 1.4), так и бесклассовая IP-адресация. Использование бесклассовой IP-адресации более технически грамотно, поскольку повышает производительность сети и ее безопасность.

На практике наибольшее распространение нашли сети классов B и C. Как правило, из этих диапазонов чаще всего используются IP-адреса для нумерации хостов в пределах локальных сетей, так называемые внутренние IP-адреса. Для адресации в сети Интернет используются внешние IP-адреса, и адресное пространство распределяется с помощью бесклассовых сетей.

1.4. Классы IP-адресов

Класс	Маска	Максимальное количество хостов	Диапазон IP-адресов
A	255.0.0.0	256^3 > 15,5 млн	1.0.0.0 – 126.0.0.0
B	255.255.0.0	256^2 > 65 тыс.	128.0.0.0 – 191.255.0.0
C	255.255.255.0	256 – 2 = 254	192.0.0.0 – 223.255.255.0
D	Групповые адреса (multicast address)		224.0.0.0 – 239.255.255.255
E	Зарезервированы для будущих применений		240.0.0.0 – 247.255.255.255

В TCP/IP существуют ограничения при назначении IP-адресов, а именно ни номера сетей, ни номера узлов не могут состоять из одних двоичных нулей или единиц.

Дело в том, что:

• заполнение нулями всех битов адреса сети означает, что узел относится к несуществующей сети с адресом 0.0.0.0;

• заполнение нулями всех битов адреса узла приводит к получению адреса сети, например: 19.0.0.0 (сеть класса A), 141.85.0.0 (сеть класса B), 192.16.2.0 (сеть класса C);

• заполнение единицами всех битов адреса сети не предусмотрено выделением классов A, B, C, D, E;

• заполнение единицами всех битов адреса узла дает особые адреса – широковещательные (англ. Limited Broadcast), применяемые при рассылке ограниченных широковещательных сообщений всем клиентам данной сети (например, адрес 192.168.1.255 – является широковещательным для сети 192.168.1.0 (класс C) и не может быть использован ни одним узлом сети).

Особый смысл имеет IP-адрес, первый байт которого равен 127. Этот адрес является *внутренним адресом стека протоколов компьютера* (или

маршрутизатора). Он используется для тестирования программ, а также для организации работы клиентской и серверной частей приложения, установленных на одном компьютере. Обе программные части данного приложения спроектированы в расчете на то, что они будут обмениваться сообщениями по сети. Когда программа посылает данные по IP-адресу 127.x.x.x, то данные не передаются в сеть, а возвращаются модулям верхнего уровня того же компьютера как только что принятые. Маршрут перемещения данных образует «петлю», поэтому этот адрес называется адресом обратной петли (англ. Loopback – петля возврата).

Групповые адреса (англ. Multicast Address), относящиеся к классу D, предназначены для экономичного распространения в Интернете или большой корпоративной сети аудио- или видеопрограмм, адресованных сразу большой аудитории слушателей или зрителей. Если групповой адрес помещен в поле адреса назначения IP-пакета, то данный пакет должен быть доставлен сразу нескольким узлам, которые образуют группу. Один и тот же узел может входить в несколько групп. В общем случае члены группы могут распределяться по различным сетям, находящимся друг от друга на произвольно большом расстоянии.

Групповые адреса не имеют в своем составе ни адреса сети, ни адреса узла и обрабатываются маршрутизаторами особым образом. Основное назначение групповых адресов – распространение информации по схеме «один ко многим».

1.10. ВЫДЕЛЕННЫЕ ДИАПАЗОНЫ АДРЕСОВ IP v4 ДЛЯ ЛОКАЛЬНЫХ СЕТЕЙ

Проблема экономного использования адресного пространства для версии IP v4 решается несколькими взаимно дополняющими друг друга способами.

Так, например, поскольку большинство пользователей глобальных сетей подключается в составе локальных сетей (например, локальных сетей местного

провайдера), то нет смысла каждому пользовательскому компьютеру присваивать постоянный полноценный IP-адрес. При таком подходе множество адресов IP v4 было бы давным-давно исчерпано.

Пакеты из локальной сети подлежат маршрутизации при их передаче в глобальные сети, а, поэтому для пользователей локальных сетей во всем мире можно применять одни и те же диапазоны сетевых адресов. Это позволяет в несколько раз сократить расход адресов IP v4.

В таком случае маршрутизатор должен обеспечивать поддержку NAT (англ. Network Address Translation – преобразование сетевых адресов).

В каждом из рассмотренных нами классов A, B, и C имеется специально выделенный для использования в локальных сетях диапазон адресов:

– класс A: 10.0.0.0 – 10.255.255.255 (только сеть 10.0.0.0);

– класс B: 172.16.0.0 – 172.31.255.255 (сети с 172.16.0.0 по 172.31.0.0);

– класс C: 192.168.0.0 – 192.168.255.255 (сети с 192.168.0.0 по 192.168.255.0).

Адреса из этих диапазонов в глобальных сетях просто игнорируются.

При проектировании локальной сети следует выбрать диапазон адресов из состава перечисленных в списке, учитывая при этом число ЭВМ в сети.

Так, например, для достаточно простой сети небольшой организации, число ЭВМ в которой вряд ли превысит несколько десятков, не стоит выбирать адрес сети классов A или B.

Достаточно выбрать любую сеть класса C:

– 192.168.0.0,

– 92.168.1.0,

– 192.168.2.0,

– и т.д. до 192.168.255.0.

Для сетей, объединяющих большое число компьютеров (несколько сотен), можно использовать адреса из диапазонов класса A или класса B. Также для больших сетей возможно использование нескольких адресов

класса С (организация подсетей с последующим объединением с помощью маршрутизаторов).

Напомним, что пакеты, в заголовках которых указаны IP-адреса другой сети, игнорируются всеми узлами данной сети. На этом основано разделение сетей на подсети.

Разделить сеть на несколько подсетей можно с помощью маски сети.

1.11. МАСКА СЕТИ (ПОДСЕТИ)

На практике задаваемый или присваиваемый сетевому узлу IP-адрес всегда дополняется так называемой маской подсети, которая имеет точно такой же формат, как и сам адрес, т.е. состоит из 4 байт, разделяемых при записи точками.

Маска – это число, применяемое в паре с IP-адресом, причем двоичная запись маски содержит непрерывную последовательность единиц в тех разрядах, которые должны в IP-адресе интерпретироваться как номер сети.

Граница между последовательностями единиц и нулей в маске соответствует границе между номером сети и номером узла в IP-адресе.

Так, например, для адреса класса А (в двоичной записи):

адрес: 0xxxxxxx.xxxxxxxx.xxxxxxxx.xxxxxxxx

маска: 11111111.00000000.00000000.00000000,

т.е. адрес сети – первые 8 бит (первые восемь единиц маски), а адрес узла остальные 24 бита (24 нуля в маске).

Маска сети, записанная в двоичном виде, всегда начинается с непрерывной последовательности единиц. Если какой-либо бит маски оказался нулем, то и все последующие биты должны быть нулями.

Таким образом, для адресов из классов А, В и С, применяемых к реальным узлам сетей, маски принимают стандартную запись:

- маска для класса А: 255.0.0.0 (8 единиц в маске);
- маска для класса В: 255.255.0.0 (16 единиц в маске);
- маска для класса С: 255.255.255.0 (24 единицы в маске);

Маска может указываться как явно, так и в сокращенном виде. Например, адрес сети может быть указан одним из двух способов:

1) 151.24.0.0 (255.255.0.0) – маска указана явно;

2) 151.24.0.0/16 – указано число первых бит «1» в составе маски.

1.12. РАЗДЕЛЕНИЕ СЕТИ НА ПОДСЕТИ

Снабжая каждый IP-адрес собственной маской, администратору локальной сети можно отказаться от понятий классов адресов и сделать более гибкой систему адресации узлов своей сети.

В частности, использование маски позволяет выполнить логическую структуризацию сети, т.е. разбить имеющийся диапазон адресов сети на несколько логически несвязанных подсетей (их можно будет в дальнейшем связать при помощи маршрутизаторов или на основе сервера).

Рассмотрим подобную ситуацию более подробно.

Пусть в некоторой локальной сети для адресации узлов планируется использование диапазона адресов 192.168.0.0. Однако при использовании стандартной маски 255.255.255.0 (/24) мы получим одну сеть, максимальное число ЭВМ в которой: 28 – 2 узла (адрес узла из 8 бит, исключая байты 00000000 и 11111111).

Рассмотрим в качестве примера применение маски 255.255.255.128 (или/25) на множестве адресов 192.168.0.0 (табл. 1.5).

Таким образом, назначив к использованию диапазон 192.168.0.0 с маской 255.255.255.128 (или/25), мы получили две подсети с максимальным числом узлов до 126 в каждой:

– 192.168.0.0/25 с диапазоном рабочих адресов 192.168.0.1 – 192.168.0.126 (адрес 192.168.0.127 – широковещательный);

– 192.168.0.128/25 с диапазоном рабочих адресов 192.168.0.129 – 192.168.0.254 (адрес 192.168.0.255 – широковещательный).

1.5. Расчет подсетей для 192.168.0.0/25

	Двоичные октеты				Десятичный адрес	Подсети
	1 байт	2 байта	3 байта	4 байта		
Маска	11111111	11111111	11111111	**1**0000000		
Адреса	11000000	10101000	00000000	0<u>0000000</u>	192.168.0.0	
	11000000	10101000	00000000	0<u>0000001</u>	192.168.0.1	Первая подсеть из 126 узлов
	11000000	10101000	00000000	0<u>0000010</u>	192.168.0.2	
	…				…	
	11000000	10101000	00000000	0<u>1111111</u>	192.168.0.127	broadcast
	11000000	10101000	00000000	1<u>0000000</u>	192.168.0.128	
	11000000	10101000	00000000	1<u>0000001</u>	192.168.0.129	Вторая подсеть из 126 узлов
	11000000	10101000	00000000	1<u>0000010</u>	192.168.0.130	
	…	…				
	11000000	10101000	00000000	1<u>1111111</u>	192.168.0.255	broadcast

Поскольку маска в четвертом байте имеет один бит «1», то узлы с адресами от 0.0.0.1 до 0.0.0.126 считают себя относящимися к одной подсети, а узлы с адресами от 0.0.0.129 до 0.0.0.254 – к другой (при использовании одной и той же маски). Пакеты этих двух подсетей могут передаваться в одной общей сети, объединенной коммутаторами, но считаются пакетами разных сетей. Непосредственное взаимодействие (обмен пакетами) узлов из разных подсетей будет невозможно без маршрутизации.

Подобными рассуждениями можно показать, что при использовании диапазона 192.168.0.0 с маской 255.255.255.192 (или /26) мы получим четыре подсети по 62 (2^6 – 2) узла в каждой. Первые биты четвертого байта IP-адреса для этих подсетей: 00, 01, 10, 11.

То есть, назначив к использованию диапазон 192.168.0.0 с маской 255.255.255.192 (или /26), мы получим четыре подсети с максимальным числом узлов до 62:

– 192.168.0.0/26 с диапазоном рабочих адресов 192.168.0.1 – 192.168.0.62 (адрес 192.168.0.63 – широковещательный);

– 192.168.0.64/26 с диапазоном рабочих адресов 192.168.0.65 – 192.168.0.126 (адрес 192.168.0.127 – широковещательный);

– 192.168.0.128/26 с диапазоном рабочих адресов 192.168.0.129 – 192.168.0.190 (адрес 192.168.0.191 – широковещательный);

– 192.168.0.192/26 с диапазоном рабочих адресов 192.168.0.193 – 192.168.0.254 (адрес 192.168.0.255 – широковещательный).

При использовании диапазона 192.168.0.0 с маской /27 возможна эксплуатация восьми подсетей из 30 (2^5 – 2) узлов каждая. Первые биты четвертого байта в этих подсетях будут: 000, 001, 010, 011, 100, 101, 110, 111.

В случае необходимости построения более крупной сети не используют адреса класса С, а используют адреса класса В (172.16.0.0 – 172.31.0.0) или даже класса А (диапазон 10.0.0.0) в сочетании с соответствующими заданным размерам сети масками.

Например:

– 172.16.0.0 с масками /17, /18, /19 (обычная маска /16),

– 10.0.0.0 с масками /9, /10, /11 (обычная маска /8) и т.д.

1.13. НАСТРОЙКА IP-АДРЕСОВ В ЛОКАЛЬНЫХ СЕТЯХ

Локальная сеть предприятия, квартиры (рис. 1.4) начинается с входящего нитевого кабеля *1*, от коммутатора провайдера *2*, который подключен к маршрутизатору провайдера *3*. В свою очередь маршрутизатор провайдера подключен к маршрутизатору более высокого ранга *4*, который, собственно, и продает Интернет провайдеру, а провайдер перепродает его уже конечному пользователю.

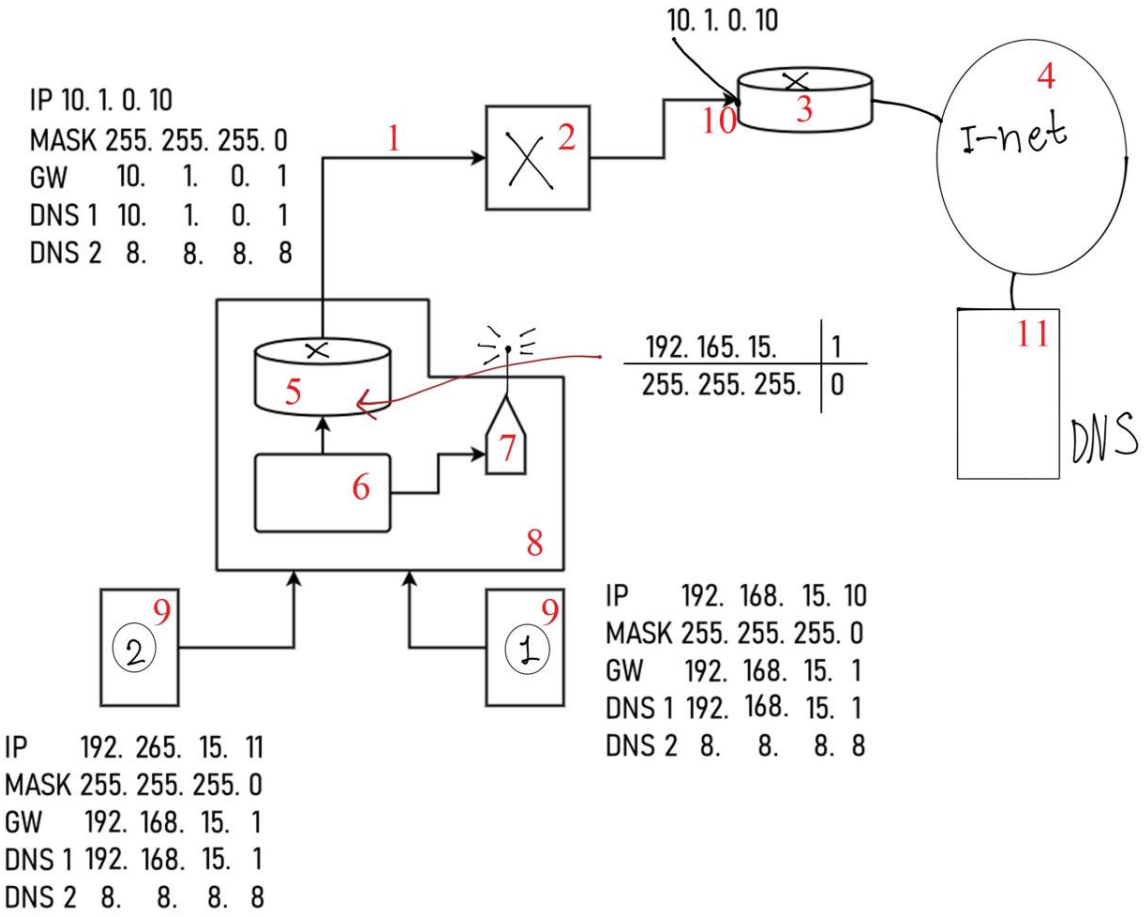


Рис. 1.4. Настройка IP-адресов в локальных сетях

Кабель провайдера подключается к маршрутизатору локальной сети *5*. В состав маршрутизатора входит коммутатор *6* и Wi-Fi – точка доступа *7*. Все это заключено в пластиковый корпус *8*, к которому через интерфейсы подключаются компьютеры, IP-камеры и другие сетевые устройства *9*.

Для подключения к сети Интернет провайдер выдает настройки, которые надо указать на маршрутизаторе локальной сети. В рассматриваемом примере провайдер выдал внешний IP-адрес 10.1.0.10 с маской 255.255.255.0, в качестве шлюза по умолчанию GW провайдер указывает IP-адрес *10* своего маршрутизатора 3, 10.1.0.1. В качестве DNS-сервера крупный провайдер может иметь свой выделенный сервер, на котором хранится база данных сопоставления IP-адресов сети Интернет с их доменными именами. Небольшой провайдер может использовать свой маршрутизатор, 10.1.0.1, либо удаленный

DNS-сервер *11*, например DNS-сервер компании Google с IP-адресом 8.8.8.8. На этом настройка внешнего интерфейса заканчивается.

Основной объем работ при настройке локальной сети представляет собой распределение внутренней IP-адресации. Внутренний IP-адрес маршрутизатора *5* назначается из пула IP-адресов, например 192.168.15.1, сеть класса С, с маской 255.255.255.0. Далее необходимо присвоить IP-адреса хостам сети *9*. IP-адрес хоста (1) 192.168.15.10, маска 255.255.255.0. Шлюз по умолчанию нужен компьютеру для того, чтобы знать, куда отправить запрос в случае, если вводится IP-адрес внешней сети. В качестве шлюза по умолчанию используется маршрутизатор 192.168.15.1. В качестве первичного DNS-сервера чаще всего используется маршрутизатор 192.168.15.1, в качестве вторичного DNS-сервера можно использовать тот, который выдал провайдер, 10.1.0.1 либо какой-то внешний, например компании Google, 8.8.8.8. На этом настройка первого хоста закончена.

Настройка других компьютеров локальной сети (2) аналогична, однако номер хоста должен быть отличным от других, уже распределенных адресов.

1.14. ШЛЮЗ ПО УМОЛЧАНИЮ

На рисунке 1.5 представлена схема, состоящая из маршрутизатора *1*, установленного в малом офисе или квартире, компьютера *2*, который подключен к маршрутизатору *1*. Маршрутизатор малого офиса подключен к сети интернет-провайдера *3*, в свою очередь сеть провайдера подключена к сети Интернет *4*. Где-то в сети Интернет есть удаленный сервер *5*, на который нужно получить доступ. У этого сервера *5* есть внешний белый IP-адрес 217.69.139.201. Так же в сети Интернет располагается удаленный DNS-сервер *6*.

Провайдер выдал маршрутизатору локальной сети IP-адрес 10.1.0.2, маску своей сети 255.255.255.248 и IP-адрес шлюза по умолчанию 10.1.0.1, т.е. внутренний адрес своего маршрутизатора, а также адрес своего DNS-сервера (DNS1) и адрес внешнего DNS-сервера (DNS2). Распределение адресного пространства внутри сети было показано в разделе 1.11.

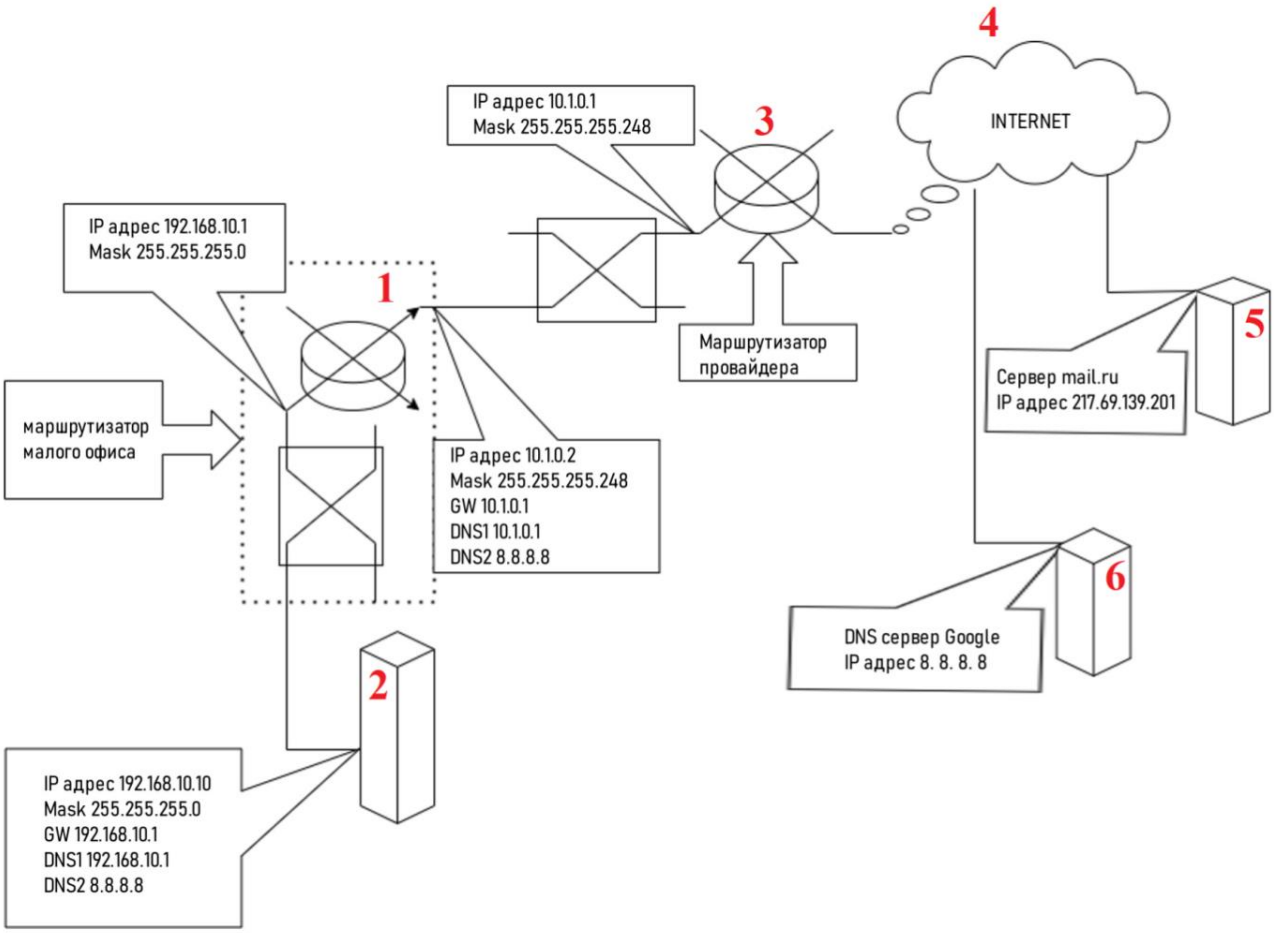


Рис. 1.5. Демонстрация доступа к удаленному серверу

Для того, что бы получить доступ к удаленному серверу, в адресной строке браузера необходимо ввести запрос вида http://IP-адрес_удаленного_сервиса. Например, http://217.69.139.201, это IP-адрес компании mail.ru. После введения подобного запроса формируется информационный пакет, в котором есть поля: IP-адрес отправителя, IP-адрес получателя. Если номера сетей IP-адрес отправителя и IP-адрес получателя будут одинаковыми, то пакет отправится компьютеру, установленному в локальной сети. Если номера сетей разные, как в примере, то компьютер должен знать, что подобный пакет нужно отправить на маршрутизатор сети или на шлюз по умолчанию 192.168.10.1. Для этого в настройке компьютера указывается в качестве шлюза по умолчанию IP-адрес маршрутизатора локальной сети. После получения пакета маршрутизатором идет проверка номера сети запрашиваемого адреса и номера сети внешнего интерфейса, если они не совпадают, то пакет отправляется на маршрутизатор

провайдера, который прописан как шлюз по умолчанию 10.1.0.1. Маршрутизатор провайдера имеет большую гибкость настроек, на нем могут быть запущены несколько протоколов маршрутизации, созданы статические маршруты и указаны маршруты по умолчанию. Маршрутизатор провайдера таким образом находит другие маршрутизаторы сети Интернет, которые последовательно приведут к серверу назначения и удаленный сервер отправит ответ на запрашиваемый сервис, и в браузере начнется загрузка запрашиваемой страницы.

Максимальное количество маршрутизаторов, которые может пройти информационный пакет, определяется временем его жизни, TTL (Time To Live). При прохождении каждого маршрутизатора время жизни пакета сокращается на 1. Если в маршрутизатор пришел пакет с временем жизни TTL = 0, то такой пакет уничтожается, а в браузер придет ответ о недоступности данного узла.

1.15. СЛУЖБА DNS

DNS (Domain Name System – система доменных имен) – компьютерная распределенная система для получения информации о доменах. Чаще всего используется для получения IP-адреса по имени хоста (компьютера или устройства). DNS важна для работы Интернета, так как для соединения с узлом необходима информация о его IP-адресе, а для людей проще запоминать буквенные (обычно осмысленные) адреса, чем последовательность цифр IP-адреса.

Рассмотрим работу службы DNS на примере рис. 1.6. Если в браузере ввести строку вида http://mail.ru. Если в кеш компьютера *1* есть сопоставление данного доменного имени с IP-адресом, то компьютер возьмет этот адрес и через шлюз по умолчанию 192.169.10.1 отправит запрос на удаленный сервер *2*. Если в кеш компьютера такой информации нет, компьютер отправляет запрос на узел, IP-адрес которого указан в поле первичного DNS (DNS1: 192.168.10.1). Если DNS1 находится в пределах локальной сети, то запрос сразу уходит на него, если нет, то запрос через шлюз по умолчанию *3* отправляется во внешнюю сеть.

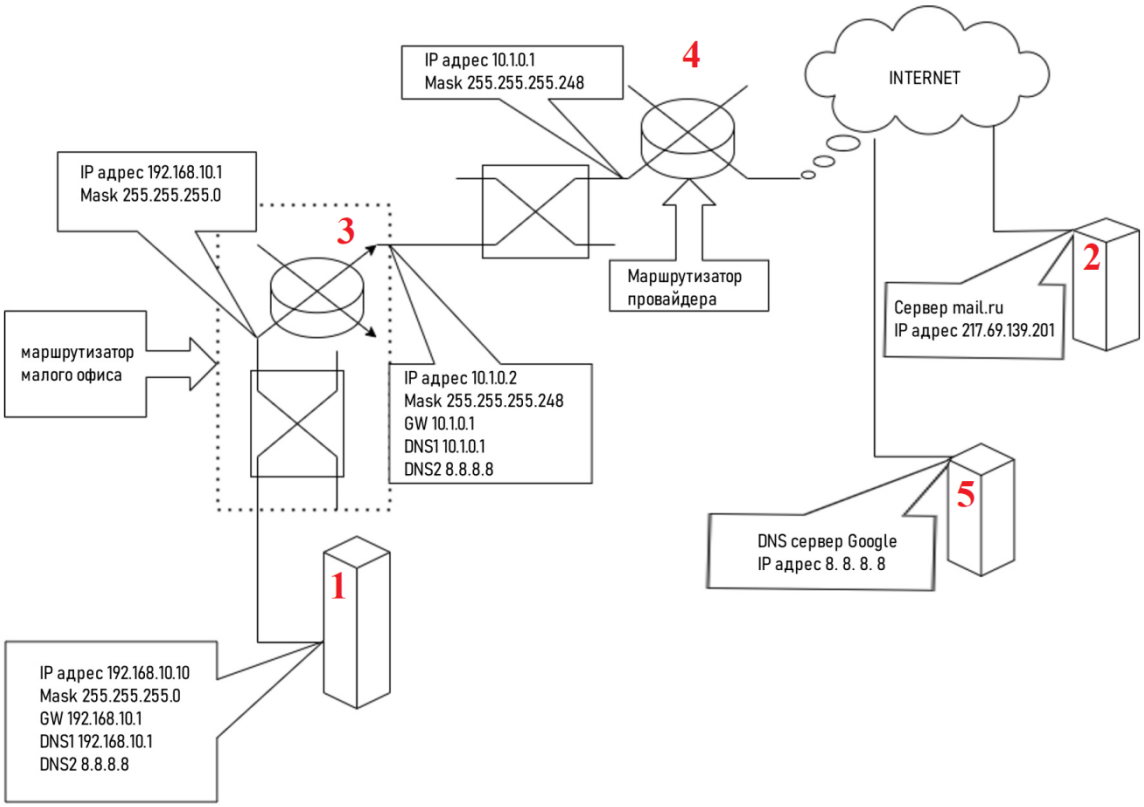


Рис. 1.6. Работа службы DNS

В небольших сетях шлюз по умолчанию является DNS1 и запрос на сопоставление доменного имени IP-адресу отправляется непосредственно на него. Если в кеш DNS1 есть такое сопоставление, он выдает данное сопоставление компьютеру, если такого сопоставления нет, то он запрашивает его у DNS сервера провайдера *4*. Если в сервере провайдера есть такое сопоставление, то ответ отправится DNS1 маршрутизатора *3*, если нет, то запрос отправляется уровнем выше и так до тех пор, пока не будут достигнуты корневые сервера *5*. В корневых серверах есть информация обо всех доменных именах. Если корневой сервер обнаруживает в своей базе сопоставление, например вида http://mail.ru = 217.69.139.201, то он выдаст ответ на запрос, содержащий IP-адрес, который сохранят все промежуточные DNS-сервера в цепочке между корневым сервером и компьютером сети. Если соответствия нет, то выдается ответ о недоступности подобного домена. Сохранение информации в промежуточных серверах необходимо для того, чтобы не перегружать сеть лишним трафиком, а DNS-сервера более высокого уровня – запросами. Для повышения

надежности системы в настройках хоста *1* и в настройках маршрутизатора *3* рекомендуется, кроме IP-адреса первичного DNS, назначить вторичный DNS-сервер.

1.16. ПРОТОКОЛ ПРИКЛАДНОГО УРОВНЯ DHCP

DHCP (Dynamic Host Configuration Protocol – протокол динамической настройки узла) – сетевой протокол, позволяющий компьютерам автоматически получать IP-адрес и другие параметры, необходимые для работы в сети TCP/IP. Данный протокол работает по модели «клиент–сервер». Для автоматической конфигурации компьютер-клиент на этапе конфигурации сетевого устройства обращается к так называемому серверу DHCP и получает от него нужные параметры. Сетевой администратор может задать диапазон адресов, распределяемых сервером среди компьютеров. Это позволяет избежать ручной настройки компьютеров сети и уменьшает количество ошибок. Протокол DHCP используется в большинстве сетей TCP/IP.

1.17. СХЕМЫ АДРЕСАЦИИ УЗЛОВ В СЕТЯХ

Для однозначной адресации интерфейсов используются локальные (физические, аппаратные, в технологии Ethernet – MAC) адреса.

Пара, состоящая из номера сети и номера узла, отвечает поставленным условиям и может служить в качестве сетевого адреса. Каждый раз, когда пакет направляется адресату через сеть, в его заголовке указывается IP-адрес узла назначения. По номеру сети назначения каждый очередной маршрутизатор находит IP-адрес следующего маршрутизатора. Перед тем как отправить пакет в следующую сеть, маршрутизатор должен определить на основании найденного IP-адреса следующего маршрутизатора его локальный адрес. Для этой цели протокол IP, как показано на рис. 1.7, обращается к протоколу ARP (англ. Address Resolution Protocol – протокол определения адреса).

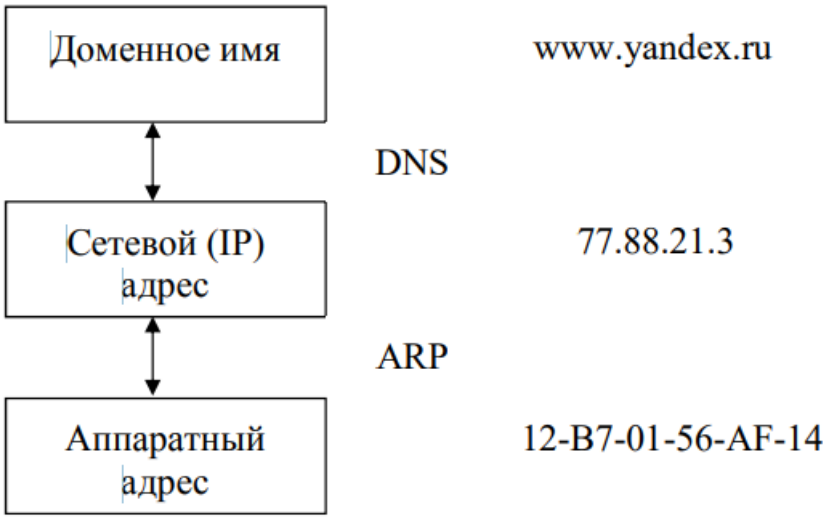


Рис. 1.7. Преобразование адресов

Символьные идентификаторы сетевых интерфейсов в составных сетях строятся по иерархическому признаку. Составляющие полного символьного (доменного) имени в IP-сетях разделяются точкой и перечисляются в следующем порядке: сначала простое имя хоста, затем имя группы хостов (например, имя организации), затем имя более крупной группы (домена) и так до имени домена самого высокого уровня (ru, com, org или др.). Между доменным именем и IP-адресом узла нет никакой функциональной зависимости, поэтому единственный способ установления соответствия – это таблица соответствия, для этого используется служба доменных имен (DNS), которая устанавливает это соответствие на основании создаваемых администраторами сети таблиц соответствия.

2. ЛАБОРАТОРНЫЙ ПРАКТИКУМ

Лабораторная работа № 1

РАБОТА С СЕТЕВЫМИ ПРОТОКОЛАМИ СРЕДСТВАМИ ОС WINDOWS

Целью лабораторной работы является знакомство с утилитами для работы с сетью, встроенными в операционную систему Windows.

Перед тем, как приступить к выполнению работы, прочитать про эталонную сетевую модель OSI (Open System Interconnection):

https://infocisco.ru/network_model_osi.html .

В настоящее время широкое распространение получили сети на базе технологии Ethernet. Ключевыми компонентами при построении таких сетей являются протоколы канального уровня (в рамках спецификации IEEE 802) и сетевого уровня (RFC 791). Несмотря на то, что работа с данными протоколами в основном сосредоточена в специализированных устройствах управления сетью (коммутаторы, маршрутизаторы), нередко требуется получать информацию об их работе на оконечном оборудовании (компьютерах).

Стандарт IEEE 802 разделяет канальный уровень на два подуровня: MAC-подуровень (Media Access Control) и LLC-подуровень (Logical Link Control).

Наиболее часто в настройках сети приходится сталкиваться с MAC-подуровнем, а именно при работе с адресацией.

MAC-адрес (media access control address) – уникальный идентификатор, назначенный сетевому адаптеру, применяется в сетях стандартов IEEE 802, в основном Ethernet, Wi-Fi и Bluetooth. Официально он называется «идентификатором типа EUI-48». Из названия видно что адрес имеет длину в 48 бит, т.е. 6 байт. Общепринятого стандарта на написание адреса нет (в противополож-

ность IPv4 адресу, где октеты всегда разделяют точками). Обычно он записывается как шесть шестнадцатеричных чисел, разделенных двоеточием: 00:AB:CD:EF:11:22, хотя некоторые производители оборудования предпочитают запись вида 00-AB-CD-EF-11-22 и даже 00ab.cdef.1122.

Адрес состоит из части идентификатора производителя, OUI, и идентификатора, присваиваемого производителем. Для обеспечения глобальной уникальности адреса назначением идентификаторов OUI (Organizationally Unique Identifier) занимается одна организация IEEE (IEEE Registration Authority), которая выделяет целые диапазоны адресов производителям сетевого оборудования. Длина может быть не только 3 байта (24 бита), а 28 или 36 бит, из которых формируются блоки (MAC Address Block, MA) адресов типов Large (MA-L), Medium (MA-M) и Small (MA-S) соответственно. Размер выдаваемого блока в таком случае составит 24, 20, 12 бит или 16 млн, 1 млн, 4 тыс. адресов. В настоящий момент распределено порядка 38 тыс. блоков, их можно посмотреть многочисленными онлайн-инструментами, например у IEEE или Wireshark.

https://regauth.standards.ieee.org/standards-ra-web/pub/view.html;

https://www.wireshark.org/tools/oui-lookup.html.

Производитель в свою очередь обеспечивает уникальность адресов в рамках своей линии производства. Вторая тройка (Network Interface Controller) NIC – обозначает конкретное устройство, которым может быть сетевая карта, Wi-Fi или Bluetooth-модуль.

Стандарты IEEE определяют 48-разрядный (6 байт) MAC-адрес, который разделен на две части:

– старшие 3 байта адреса содержат уникальный идентификатор организации (OUI), который производитель получает в IEEE. В старшем байте под OUI используется старшие 6 бит. Два младших бита имеют специальное назначение: нулевой бит указывает для одиночного (0) или группового (1) адресата предназначен кадр; первый бит указывает, является ли MAC-адрес глобально (0) или локально (1) администрируемым;

– младшие 3 байта содержат уникальный адрес контроллера (NIC), который выбираются изготовителем для каждого экземпляра устройства.

***Задание* 1.1.**

Применить утилиту *getmac* для получения MAC-адреса компьютера. Полученное значение использовать на сайте Wireshark и определить производителя оборудования.

getmac[.exe] [/s Computer [/u Domain\User [/p Password]]] [/nh] [/v].

Возвращает список MAC-адресов всех сетевых интерфейсов компьютера. Параметры

/s Computer – задает имя или IP-адрес удаленного компьютера. Если не используется, то выбран локальный компьютер;

/u Domain\User – указывает имя пользователя, под которым делается попытка входа на удаленный компьютер;

/p Password – указывает пароль для входа пользователя;

/nh – вывод информации без строки заголовков;

/v – отображение подробной информации.

Примеры:

getmac /v – получить информацию о MAC-адресах локального компьютера;

getmac /s 192.168.0.1 /u student /p rtf – получить информацию о MAC-адресах компьютера с IP-адресом 192.168.0.1, пытаясь использовать учетную запись пользователя student с паролем rtf.

***Задание* 1.2.**

Применить утилиту *arp* с перечисленными ниже параметрами для вывода и анализа таблицы ARP-кеша.

Утилита ARP (Address Resolution Protocol) – протокол разрешения адресов) позволяет отображать и изменять таблицу трансляции IP-адресов в локальные физические MAC-адреса. Записи в данной таблице ARP-кеша формирует

протокол ARP. В случае отсутствия IP-адреса в таблице ARP-кеша, данный протокол отправляет широковещательный ARP-запрос всем устройствам в подсети, пытаясь определить MAC-адрес, принадлежащий данному IP-адресу.

ARP-таблица содержит в себе два типа записей адресов: динамические и статические адреса. Динамические записи помещаются в таблицу автоматически, путем широковещательных запросов, а также автоматически удаляются по истечении определенного времени. Статические добавляются вручную и хранятся постоянно.

Синтаксис команды:

arp [-s inet_addr eth_addr] [-d inet_addr] [-a].

Параметры команды *arp*:

-s inet_addr eth_addr – добавляет в таблицу ARP-кеша статическую запись с заданными IP-и MAC-адресом;

-d inet_addr – удаляет из ARP-таблицы запись, соответствующую указанному IP-адресу;

-a – отображает содержимое таблицы ARP-кеша для всех адаптеров на данном устройстве.

***Задание* 1.3.**

Применить утилиту *ipconfig* с перечисленными параметрами для проверки правильности конфигурации TPC/IP для операционной системы Microsoft Windows.

Данная команда выводит значения текущего состояния настроек стека TCP/IP: IP и MAC-адрес компьютера, IP-адрес шлюза по умолчанию, маску подсети, IP-адреса серверов DHCP (Dynamic Host Configuration Protocol), WINS (Windows Internet Naming Service) и DNS (Domain Name System).

При попытке устранения неисправностей в сети протокола TCP/IP необходимо в первую очередь проверить конфигурацию сети, используя диагностическую утилиту *ipconfig*.

Синтаксис данной утилиты:

ipconfig [/all] [/renew[adapter]] [/release[adapter]].

В квадратных скобках указаны опциональные параметры команды:

/all – показывается полный список настроек сети. Без этого параметра отображаются только базовые данные: IP-адрес компьютера, маска подсети, адрес шлюза по умолчанию;

/renew [adapter] – применение данного параметра выполняет команду обновления настроек DHCP. В случае использования дополнительного параметра *adapter* обновляются настройки только указанного адаптера;

/release [adapter] – применение данного параметра освобождает используемый адаптером IP-адрес;

/flushdns – очистить содержимое кеша, содержащего соответствие DNS-имен и IP-адресов;

/displaydns – вывести содержимое кеша, содержащего соответствие DNS-имен и IP-адресов;

/registerdns – инициализировать регистрацию записей DNS для всех адаптеров компьютера. Используется при изменении настроек DNS-сервера.

Утилита *ipconfig* позволяет установить, сконфигурирована ли сеть и нет ли проблемы дублирования IP-адресов. В случае если:

- проблем не обнаружено, то отображаются IP-адрес компьютера, маска подсети и адрес шлюза;
- дублируются IP-адреса, маска подсети принимает значение 0.0.0.0;
- есть проблемы получения IP-адреса при использовании DHCP, адрес компьютера примет значение 0.0.0.0.

***Задание* 1.4.**

Применить утилиту *ping* с перечисленными ниже параметрами для проверки правильности конфигурации TPC/IP и диагностики ошибок соединения. Данная утилита позволяет проверить доступность и функционирование хоста.

Применение утилиты *ping* позволяет проверить существование маршрута между компьютером и устройством в сети протокола TCP/IP.

Данная команда посылает эхо-пакеты протокола ICMP (Internet Control Message Protocol) и прослушивает полученные эхо-ответы, таким образом проверяя соединение с целевым адресом. Команда *ping* покажет число принятых и переданных пакетов, а также время (в миллисекундах), за которое посланный пакет доходит до адреса и возвращается обратно. По умолчанию время ожидания равно 1 с, а значит, что максимальное время отклика составляет 1000 мс. В случае превышения этого времени выводится сообщение «Превышен интервал ожидания». В таком случае возможно, что, увеличив время ожидания с помощью параметра *-w*, можно получить отклик от хоста. Некоторые серверы в целях безопасности отключают отправку эхо-ответов.

Команда *ping* может применяться как с IP-адресом, так и с символьным именем, как в примере выше. В случае если команда успешно выполняется с IP-адресом, а при использовании символьного имени — неудачно, то это значит, что проблема заключается в службе распознавания символьных имен DNS, а не в самом соединении.

Синтаксис команды:

ping [-t] [-a] [-n count] [-l length] [-f] [-I ttl] [-v tos] [-r count] [-s count]

[-j host-list] [-k host-list] [-w timeout] destination-list.

Параметры команды *ping*:

-t – продолжает посылать эхо-пакеты до принудительного прерывания. Нажав комбинацию клавиш Ctrl+C, можно прервать выполнение команды. Без этого параметра команда *ping* посылает четыре эхо-пакета;

-a – этот параметр позволяет определить символьное имя хоста по IP-адресу;

-n count – данный параметр позволяет указать количество эхо-пакетов для отправки, где *count* – число пакетов (по умолчанию команда ping отправляет четыре эхо-запроса);

-l length – устанавливает размер пакета длиной в *length* байтов (максимум 8192 байта);

-f – устанавливает на посылаемом пакете флаг «не фрагментировать», отключающий фрагментацию на промежуточных маршрутизаторах;

-i ttl – задает параметр TTL (Time to live) – время жизни пакета. При прохождении каждого маршрутизатора TTL уменьшается на единицу, т.е. время жизни пакета – это счетчик пройденных маршрутизаторов;

-v tos – указывает приоритет обработки пакета;

-r count – запись пути исходящего и возвращающегося пакетов, *count* – значение от 1 до 9 хостов;

-s count – указывает максимальное значение переходов между подсетями (хопов);

-j host-list – задает список хостов для направления пакетов. Максимум девять хостов;

-k host-list – задает жесткую статическую маршрутизацию через список хостов, при этом хосты не могут быть разделены между собой промежуточными маршрутизаторами;

-w timeout – указывает в миллисекундах время ожидания ответа. По умолчанию ожидание составляет 1 с;

-destination-list – список IP-адресов или символьных имен, к которым необходимо выполнить команду ping;

Как правило, в реальной ситуации наиболее используемыми являются два параметра: *-n* и *-t*.

***Задание* 1.5.**

Применить утилиту *tracert* с перечисленными ниже параметрами для выявления последовательности маршрутизаторов, через которые проходят IP-пакеты.

С помощью утилиты *tracert* возможно более эффективно локализовать сегмент сети, в котором возникла проблема. Также необходимо учитывать, что

некоторые маршрутизаторы настроены таким образом, что они просто уничтожают пакеты с истекшим временем жизни, не отправляя обратно сообщение об ошибке.

Синтаксис команды:

tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout] destination-list.

Параметры команды *tracert*:

-d – отключает распознавание адресов для имен хостов;

-h maximum_hops – задает лимит хопов при трассировке маршрута. По умолчанию 30 хопов;

-j host-list – указывает нежесткую статическую маршрутизацию в соответствии со списком хостов;

-w timeout – задает в миллисекундах параметр тайм-аута ожидания ответа на каждый эхо-пакет.

Пример применения команды *tracert* с параметрами: отключено отображение имен хостов (*-d*), задано максимальное число хопов 25 (*-h 25*), время тайм-аута 100 миллисекунд (*-w 100*), параметром -4 задано принудительное использование протокола ipv4:

tracert -d -h 25 -w 100 -4 dzen.ru.

***Задание* 1.6.**

Применить утилиту *netstat* с перечисленными ниже параметрами для вывода и анализа статистики протоколов и текущих TCP/IP соединений.

Синтаксис команды:

netstat [-a][-e][-n] [-s][-p name][-r][interval].

Параметры команды *netstat*:

-a – вывод всех соединений и портов, а также полной информации по каждому из них;

-e – вывод статистики интерфейса подключения к сети;

-n – вывод адресов и портов в числовом формате;

-p name – настройка формата отображения информации о протоколе (в качестве значения name указывается один из трех протоколов: TCP, UDP, IP);

-r – вывод таблицы маршрутизации;

-s – вывод более подробной статистики по заданному протоколу. По умолчанию выводятся данные по трем протоколам TCP, UDP, IP, используя данный параметр одновременно с параметром *-p*, можно указать вывод по конкретному протоколу. В случае использования дополнительного оператора interval команда срабатывает повторно через заданное число секунд до принудительного прерывания через комбинацию клавиш Ctrl+C.

Пример применения команды *netstat* с параметрами *-s* для вывода данных в формате более подробной статистики, *-p tcp* для фильтрации только TCP-соединений:

netstat -s -p tcp.

***Задание* 1.7.**

Применить утилиту *nslookup* с перечисленными ниже параметрами для разрешения символьных имен через запрос к DNS-серверам.

Синтаксис команды:

nslookup [host] [server].

Параметры команды nslookup:

host – символьный адрес, который необходимо преобразовать в IP-адрес;

server – адрес сервера доменных имен (DNS-сервера), к которому будет обращаться утилита. В случае если *server* не указан, будет использоваться сервер, указанный в параметрах настройки TCP/IP, отображающийся в команде *ipconfig*.

Пример использования команды *nslookup* для определение *ip*-адреса хоста www.tstu.ru с использованием публичного Яндекс.DNS (77.88.8.8).

nslookup www.tstu.ru 77.88.8.8.

***Задание* 1.8.**

Применить сервис Whois для определения по символьному имени или IP-адресу контактных данных администратора или юридических данных владельца.

Адреса таких сервисов:

https://whois.ru/;

https://www.whois.com/whois/;

https://www.nic.ru/whois/.

Отчет по первой лабораторной работе

1. Используя утилиту *ipconfig*, необходимо определить следующие сетевые данные Вашего компьютера:
   - имя компьютера и MAC-адреса сетевых устройств компьютера;
   - IP-адреса компьютера;
   - IP-адрес шлюза;
   - IP-адреса основного и дополнительного адреса DNS-серверов;
   - адрес DHCP-сервера (в случае, если он используется).
2. Используя утилиту *nslookup*, определить IP-адрес одного из удаленных серверов, доменные имена которых указаны в табл. 2.1.
3. Используя утилиту *ping*, проверить состояние связи с любым компьютером и шлюзом локальной сети, а также с одним из удаленных серверов, доменные имена которых указаны в табл. 2.1. Определить IP-адрес хоста назначения, среднее время приема-передачи, процент потерянных пакетов.
4. Используя команду *arp*, проверить состояние ARP-кеша. Выполнить команду *ping* какого-либо хоста локальной сети, адрес которого не был отражен в кеше. Повторно открыть ARP-кеш и проконтролировать модификацию его содержимого.

Таблица 2.1

№	Адрес	№	Адрес	№	Адрес
1	dzen.ru	11	spacex.com	21	ebay.org
2	google.com	12	iprbookshop.ru	22	infocisco.ru
3	youtube.com	13	citilink.ru	23	scopus.com
4	mail.ru	14	vk.ru	24	overleaf.com
5	tstu.ru	15	twitch.tv	25	mendeley.com
6	wikipedia.org	16	pochta.ru	26	elibrary.ru
7	whatsapp.com	17	gosuslugi.ru	27	orcid.org
8	duckduckgo.com	18	apteka.ru	28	www.nasa.gov
9	roscosmos.ru	19	rewe.de	29	www.nvidia.com
10	researchgate.net	20	amazon.com	30	pythonawesome.com

5. Выполнить команду *tracert* одного из удаленных хостов, представленных в п. 2. Определить участок сети между двумя соседними маршрутизаторами, в котором происходит самая большая задержка при пересылке пакетов. Для найденных маршрутизаторов с помощью сервиса Whois определить название организаций и контактные данные администратора.

6. Посмотреть активные текущие сетевые соединения и их состояние на текущем компьютере, используя команду *netstat,* для чего:

– загрузить различные страницы с разных веб-сайтов;

– закрыть браузеры и с помощью утилиты *netstat*, проверить изменение списка сетевых подключений. Проконтролировать сетевые соединения в режиме реального времени, для чего:

– закрыть все открытые сетевые приложения;

– с помощью команды *netstat* из командной строки задать числовой формат отображения адресов и номеров портов с повторным выводом с периодом 20…30 с;

– в новом окне командной строки запустить команду *ping* в режиме «до прерывания»;

– наблюдать текущую статистику сетевых приложений и отображение *netstat*;

– последовательно закрыть утилиты *ping* и *netstat*, нажав Ctrl + C.

Контрольные вопросы

1. Дать определение MAC-адреса.
2. Что позволяют определить утилиты *ipconfig*, *nslookup*, *ping*?
3. Для чего используются команды *tracert*, *arp*, *netstat*?
4. Что такое глобальная сеть (WAN)?

РАСЧЕТ IP v4-СЕТЕЙ

Адресация составляет важную функцию протоколов сетевого уровня, поскольку обеспечивает обмен данными между узлами в одной и той же сети или между разными сетями. Целью данной лабораторной работы является изучение IP v4-адресов и компонентов, из которых они состоят – сетевую (*network*) и узловую части (*host*), маску подсети. В число рассматриваемых типов адресов входят общие (*public*) и частные (*private*) адреса, адреса для широкополосной (*broadcast*) и многоадресной (*multicast*) рассылки.

Перед тем, как приступить к выполнению работы, прочитать пример расчета количества хостов и подсетей на основе IP-адреса и маски:

https://help.keenetic.com/hc/ru/articles/213965829-Пример-расчета-количества-хостов-и-подсетей-на-основе-IP-адреса-и-маски.

В процессе выполнения работы можно использовать калькулятор IP-сетей, например:

https://www.networkcenter.info/calcs/subnetcalc.

***Задание* 2.1.**

Проанализируйте приведенную ниже таблицу и определите сетевую и узловую части указанных IPv4-адресов.

Первые две строки содержат примеры заполнения таблицы.

Сокращения, используемые в таблице:

N = все 8 бит для октета содержатся в сетевой части адреса;

n = один бит в сетевой части адреса;

H = все 8 бит для октета содержатся в узловой части адреса;

h = один бит в узловой части адреса.

IP-адрес/префикс IP-address/prefix	Сеть/узел Network/Host N, n = сеть H, h = узел	Маска подсети Subnet Mask	Сетевой адрес Network address
192.168.10.10/24	N.N.N.H	255.255.255.0	192.168.10.0
10.101.99.17/23	N.N.nnnnnnnh.H	255.255.254.0	10.101.98.0

***Задание* 2.2.**

Проанализируйте приведенную ниже таблицу и укажите диапазон адресов узлов и широковещательных адресов в виде пары маски подсети и префикса.

В первой строке приведен пример завершения таблицы.

IP-адрес/префикс IP-address/prefix	Адрес первого узла First Address	Адрес последнего узла Last Address	Широковещательный адрес Broadcast address
192.168.10.10/24	192.168.10.1	192.168.10.254	192.168.10.255
10.101.99.17/23	10.101.98.1	10.101.99.254	10.101.99.255

***Задание* 2.3.**

Проанализируйте приведенную ниже таблицу и определите тип адреса (адрес сети (*Network*), узла (*Host*), многоадресной (*Multicast*) или широковещательной (*Broadcast*) рассылки). Определите тип адреса – общий (*Public*) или частный (*Private*).

IP-адрес IP-address	Маска подсети Subnet Mask	Тип адреса Network/Host/Multicast/Broadcast	Публичный или частный Public/Private
192.168.10.10	192.168.10.254	Host	Private
10.101.98.0	255.255.255.240	Network	Private

Задание 2.4.

Проанализируйте приведенную ниже таблицу и определите, является ли пара адреса и префикса допустимым адресом узла (*Host Address*).

IP-адрес/префикс IP-address/prefix	Допустимый адрес узла? Yes/No	Причина
127.1.0.10/30	Yes	
192.31.7.256/24	No	Четвертый октет больше 255
192.31.7.255/24	No	Broadcast address

Отчет по второй лабораторной работе

Выполнить задания 2.1 – 2.4, взяв задание согласно варианту из табл. 2.2.

Таблица 2.2

№ варианта	Адреса для задания 1, 2	Адреса для задания 3	Адреса для задания 4
1	10.1.8.200/24 192.168.10.10/20 109.108.32.0/24 172.25.12.52/25	10.1.8.200/24 224.10.1.11/24 98.98.136.0/20 192.167.167.167/20	224.10.1.255/24 224.10.1.0/24 224.10.1.11/24 264.10.1.11/24
2	10.1.8.200/28 192.168.10.10/22 109.108.32.0/30 172.25.12.52/23	10.1.8.192/28 98.98.128.0/20 172.31.255.253/16 224.1.1.15/20	172.31.255.253/30 172.31.255.255/30 172.31.255.252/30 172.31.255.252/32
3	10.1.8.200/24 192.168.10.10/25 109.108.32.0/26 172.25.12.52/30	224.1.1.15/20 10.1.8.207/28 98.98.143.255/20 172.31.255.255/16	109.108.32.0/26 109.108.32.0/26 109.108.32.63/26 109.108.32.63/32
4	10.1.8.255/24 192.168.10.10/26 109.108.32.0/30 172.25.12.52/22	10.1.8.255/24 94.242.129.255/23 172.31.253.253/24 224.1.15.255/20	172.25.12.0/22 172.25.12.52/22 172.25.15.255/22 172.25.15.270/22
5	10.1.8.0/24 192.168.10.10/25 109.108.32.0/28 172.25.12.52/30	10.1.8.0/24 172.31.253.0/24 94.242.128.0/23 224.0.0.15/30	224.0.0.15/32 224.0.0.15/30 224.0.0.12/30 224.0.0.14/30
6	10.1.1.101/28 192.168.33.63/26 101.198.200.0/24 172.16.117.77/21	224.0.0.12/30 10.1.1.96/28 104.166.148.0/22 172.31.253.255/24	10.1.255.254/16 10.1.0.0/16 10.1.255.255/16 10.1.256.256/16
7	10.1.1.101/16 192.168.33.63/24 101.198.200.0/30 172.16.117.77/20	10.1.1.111/28 104.166.148.0/22 224.0.99.15/23 172.16.1.1/24	101.255.255.254/10 101.255.255.255/10 101.192.0.0/10 0.198.200.0/10

Продолжение табл. 2.2

№ варианта	Адреса для задания 1, 2	Адреса для задания 3	Адреса для задания 4
8	10.1.1.101/28 192.168.33.63/25 101.198.200.0/27 172.16.117.77/24	10.1.1.101/28 104.166.151.255/22 172.16.1.0/24 224.0.99.255/23	0.63.255.254/10 0.63.255.255.10 0.0.0.0/10 0.0.0.1/10
9	10.1.1.96/26 192.168.33.63/30 101.198.200.0/28 172.16.117.77/24	224.0.98.0/23 10.1.1.101/20 109.105.129.254/23 172.16.1.255/24	224.0.98.1/23 224.0.99.255/23 224.0.98.0/23 224.0.980.0/23
10	10.1.100.101/24 192.168.33.63/28 101.198.200.0/23 172.16.117.77/30	10.1.0.0/20 109.105.129.255/23 224.0.255.15/25 172.15.10.1/30	255.255.255.0/24 255.255.255.254/24 255.255.255.255/24 255.255.555.255/24
11	10.1.100.0/24 192.168.33.63/16 185.117.48.0/30 172.16.128.48/24	10.1.100.0/24 224.0.255.0/25 109.105.128.0/23 172.15.10.0/30	185.117.48.0/26 185.117.48.63/26 185.117.48.5/26 185.317.48.55/26
12	10.1.100.10/24 192.0.3.6/16 185.117.48.0/23 172.16.128.48/25	10.1.100.0/24 238.2.255.255/16 31.44.48.0/25 172.15.10.3/30	238.2.255.255/16 238.2.0.0/16 238.2.0.254/16 238.2.0.253/34
13	10.1.100.101/28 192.0.3.6/24 185.117.48.0/30 172.16.128.48/28	10.1.100.255/24 224.0.255.127/25 31.44.48.0/25 172.16.0.0/24	31.44.48.0/25 31.44.48.126/25 31.44.48.126/25 31.44.48.127/25
14	10.1.100.101/24 192.0.3.6/26 185.117.48.0/25 172.16.128.48/30	224.255.255.215/25 10.1.100.1/18 31.44.48.127/25 172.16.0.255/24	172.16.128.48/23 172.16.128.0/23 172.256.129.255/23 172.16.129.255/23
15	10.1.100.101/20 192.0.3.6/24 185.117.48.0/22 172.16.128.48/18	10.1.127.255/18 31.6.96.0/28 224.255.255.255/25 172.16.0.10/24	172.16.128.254/18 172.16.128.0/18 185.117.48.0/33 172.16.191.255/18

Продолжение табл. 2.2

№ варианта	Адреса для задания 1, 2	Адреса для задания 3	Адреса для задания 4
16	10.10.15.10/25 192.8.2.151/19 196.199.72.0/30 172.16.14.0/24	10.10.15.10/25 224.0.0.0/24 172.16.16.16/23 31.6.96.0/28	196.199.72.2/30 196.199.72.0/30 196.299.72.3/30 196.199.72.3/30
17	10.10.15.15/26 192.8.2.151/24 196.199.72.0/23 172.16.14.0/30	10.10.15.0/25 31.6.96.15/28 224.0.0.13/24 172.16.16.0/10	10.10.15.60/25 10.10.15.0/26 10.10.15.0/33 10.10.15.63/26
18	10.10.15.10/22 192.8.2.151/30 196.199.72.0/25 172.16.14.0/29	10.10.15.127/25 224.0.0.255/24 62.249.128.0/23 172.16.17.255/23	196.199.72.255/25 196.199.72.0/25 196.199.72.126/25 196.299.72.126/25
19	10.255.0.0/16 192.8.2.151/23 196.199.72.0/24 172.16.14.0/21	10.255.255.255/16 62.249.128.15/23 192.167.0.1/25 238.255.255.255/26	196.199.72.254/16 196.199.0.0/16 196.199.255.255/16 196.199.255.255/32
20	10.255.0.0/30 192.8.2.151/20 196.199.72.0/23 172.16.14.0/24	10.255.0.0/16 238.255.255.192/26 62.249.128.0/23 192.167.0.0/25	62.249.128.0/16 62.249.128.0/32 62.249.129.255/16 62.249.129.255/16
21	10.101.99.17/30 192.20.23.14/20 139.28.16.0/23 172.20.23.0/24	238.255.255.255/26 10.101.99.18/30 62.249.129.255/23 192.167.0.127/25	172.20.23.0/24 172.20.23.255/24 172.20.23.2/24 172.20.23.256/24
22	10.101.99.17/24 192.20.23.14/30 139.28.16.0/22 172.20.23.0/23	10.101.99.16/30 91.220.120.34/26 192.167.168.1/25 238.255.1.255/24	91.220.120.34/15 91.220.0.0/15 91.256.0.0/15 91.221.255.255/15
23	10.101.99.17/16 192.20.23.14/24 139.28.16.0/30 172.20.23.0/25	10.101.99.19/30 238.255.1.0/24 192.167.168.0/25 91.220.120.0/26	139.28.16.0/14 139.28.0.0/14 139.31.255.255/14 139.31.256.255/14

Окончание табл. 2.2

№ варианта	Адреса для задания 1, 2	Адреса для задания 3	Адреса для задания 4
24	10.101.99.17/20 192.20.23.14/25 139.28.16.0/29 172.20.23.0/24	10.101.111.255/20 192.167.168.127/25 91.220.120.63/26 238.255.1.255/24	139.28.16.1/29 139.281.16.5/29 139.28.16.7/29 139.28.16.0/29
25	10.101.99.17/24 192.20.23.14/30 139.28.16.0/23 172.20.23.0/27	238.255.128.255/22 10.101.96.0/20 95.78.0.200/26 192.168.168.168/27	95.79.255.255/15 95.79.255.256/15 95.78.0.0/15 95.79.255.254/15
26	10.255.255.24/24 192.14.20.23/18 188.215.32.0/25 172.16.33.0/30	192.168.168.191/22 10.255.255.250/18 95.78.0.192/26 238.255.128.0/22	192.14.63.254/18 192.14.0.0/18 192.14.63.255/18 192.14.63.257/18
27	10.255.255.24/22 192.14.20.23/21 188.215.32.0/30 172.16.33.0/29	10.255.255.255/18 238.255.131.255/22 192.168.168.160/22 95.78.0.255/26	192.169.255.254/15 192.168.0.0/15 192.169.255.256/15 192.169.255.255/15
28	10.255.255.24/16 192.14.20.23/19 188.215.32.0/24 172.16.33.0/27	10.255.192.0/18 192.167.167.167/20 95.47.170.133/24 238.2.128.255/16	192.14.20.23/19 192.14.0.0/19 192.14.0.256/19 192.14.31.255/19
29	10.255.255.24/20 192.14.20.23/24 188.215.32.0/23 172.16.33.0/28	10.255.240.1/20 95.47.170.255/24 238.2.0.0/16 192.167.160.0/20	188.215.33.254/23 188.215.32.0/23 188.257.33.255/23 188.215.33.255/23
30	10.255.255.24/21 192.14.20.23/24 188.215.32.0/26 172.16.33.0/30	10.255.240.0/20 238.2.255.255/16 95.47.170.0/24 192.167.175.255/20	188.215.32.0/26 188.215.32.63/26 188.256.32.63/26 188.215.32.61/26

Контрольные вопросы

1. Какие устройства считаются оконечными устройствами в сети?

2. Какие изменения в развертывании корпоративных сетей происходят в связи с внедрением концепции BYOD?

СОЗДАНИЕ ЛОКАЛЬНОЙ СЕТИ В CISCO PACKET TRACER

Cisco Packet Tracer представляет собой программный симулятор сетей и устройств фирмы Cisco. Симулятор отображает работу реальных сетевых устройств и их окружения. Программа обладает следующими возможностями: моделирование логической топологии; моделирование физической топологии; моделирование в режиме реального времени; режим симуляции с графическим отображением пересылаемых между устройствами пакетов и пр. Данный симулятор позволяет проектировать свои собственные сети с использованием таких сетевых устройств, как коммутаторы второго и третьего уровней, рабочих станций, определять типы связей между ними и соединять их.

Целью данной лабораторной работы является изучение программы Cisco Packet Tracer, приобретение практических навыков проектирования и моделирования работы сети, а также оценки принятых проектных решений.

Перед тем, как приступить к выполнению работы, прочитать пример расчета количества хостов и подсетей на основе IP-адреса и маски:

https://help.keenetic.com/hc/ru/articles/213965829-Пример-расчета-количества-хостов-и-подсетей-на-основе-IP-адреса-и-маски.

***Задание* 3.1.**

1. В программе Ciso Packet Tracer создать топологию сети согласно заданному варианту (см. табл. 2.3). Топологии сетей представлены в табл. 2.4.

2. Беспроводную сеть защитить по технологии WPA2-PSK на основе шифрования AES. Шифрование Pre-Shared Key (общий ключ) предназначено для домашних сетей и малых офисов, где у всех пользователей один пароль. Всем беспроводным устройствам назначить SSID (*Service Set Identifier* – идентификатор беспроводной сети).

3. По умолчанию у таких устройств, как Принтер, Ноутбук, PC, отсутствует модуль беспроводной связи. В настройках таких устройств на закладке «*Physical*» необходимо заменить проводной модуль на беспроводной. Для этого необходимо отключить устройство кнопкой вкл/выкл питания. Затем вытащить мышкой установленный проводной модуль, например «*PT-HOST-NM-1CFE*», после чего вместо него установить мышкой из перечня «*MODULES*» модуль беспроводной связи, например «*PT-HOST-NM-1W-AC*». Затем включить питание устройства.

4. Назначить в свойствах всем оконечным устройствам (компьютерам, принтерам, планшетам и т.п.) на закладке «*Config*» IP-адреса согласно заданному адресу сети (табл. 2.3).

5. Проверить в свойствах всех устройств на закладке «*Desktop*», значок «*Command Prompt*» настройки каждого оконечного узла (команда *ipconfig*). Необходимо добиться возможности пересылки данных между всеми объектами сети.

6. По запросу преподавателя проверить доступность соединения между двумя оконечными устройствами с помощью протокола *ICMP* (используя команду *ping*).

Отчет по третьей лабораторной работе

Выполнить задание 3.1 согласно варианту, взятому из табл. 2.3. Представить преподавателю:

1. Изображение топологии сети.

2. Результат выполнения команды *ipconfig* оконечного узла.

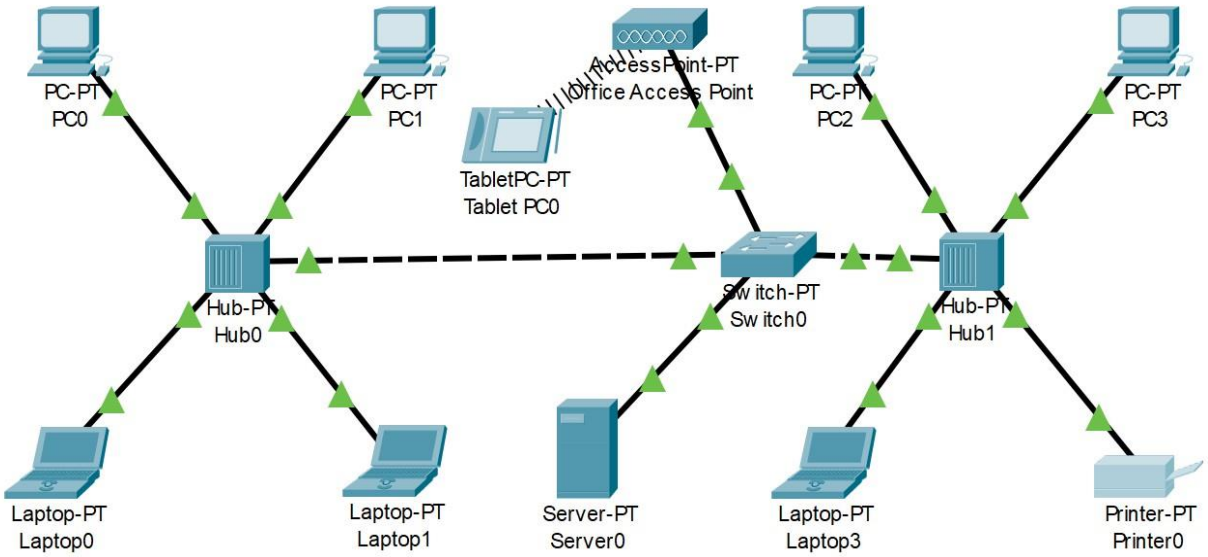
3. Результат выполнения команды *ping* между выбранными оконечными узлами.

Таблица 2.3

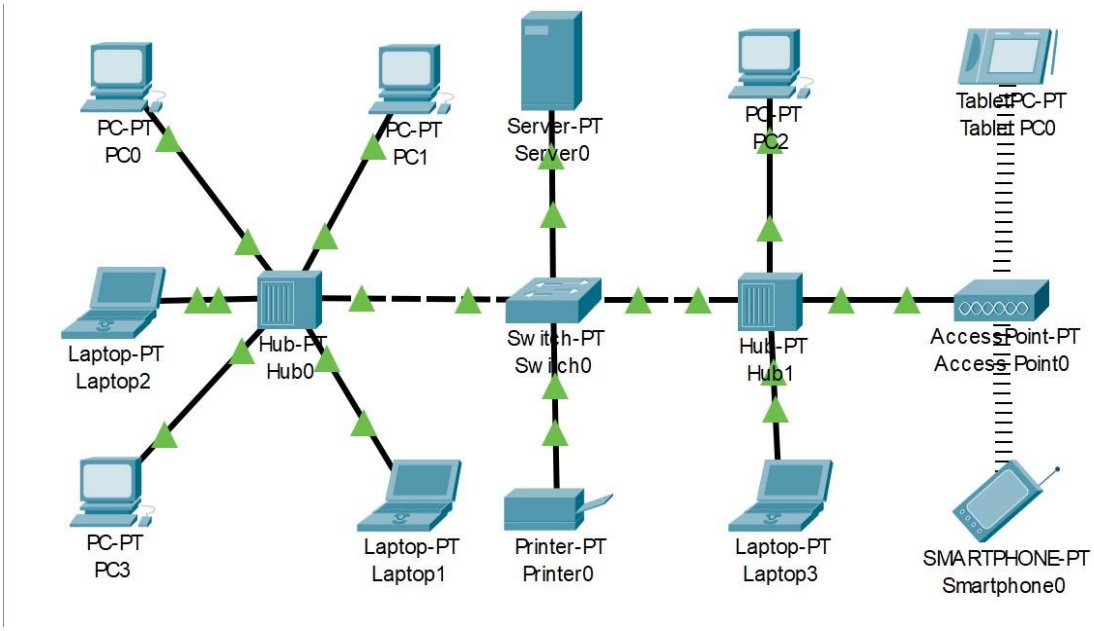
№ варианта	Топология сети (см. табл. 2.4)	Адрес сети
1	Топология 1	10.0.1.0/24
2	Топология 1	172.16.1.0/24
3	Топология 1	192.168.1.0/24
4	Топология 2	10.0.2.0/24
5	Топология 2	172.16.2.0/24
6	Топология 2	192.168.2.0/24
7	Топология 3	10.0.3.0/24
8	Топология 3	172.16.3.0/24
9	Топология 3	192.168.3.0/24
10	Топология 4	10.0.4.0/24
11	Топология 4	172.16.4.0/24
12	Топология 4	192.168.4.0/24
13	Топология 5	10.0.5.0/24
14	Топология 5	172.16.5.0/24
15	Топология 5	192.168.5.0/24
16	Топология 6	10.0.6.0/24
17	Топология 6	172.16.6.0/24
18	Топология 6	192.168.6.0/24
19	Топология 7	10.0.7.0/24
20	Топология 7	172.16.7.0/24
21	Топология 7	192.168.7.0/24
22	Топология 8	10.0.8.0/24
23	Топология 8	172.16.8.0/24
24	Топология 8	192.168.8.0/24
25	Топология 9	10.0.9.0/24
26	Топология 9	172.16.9.0/24
27	Топология 9	192.168.9.0/24
28	Топология 10	10.0.10.0/24
29	Топология 10	172.16.10.0/24
30	Топология 10	192.168.10.0/24

Таблица 2.4

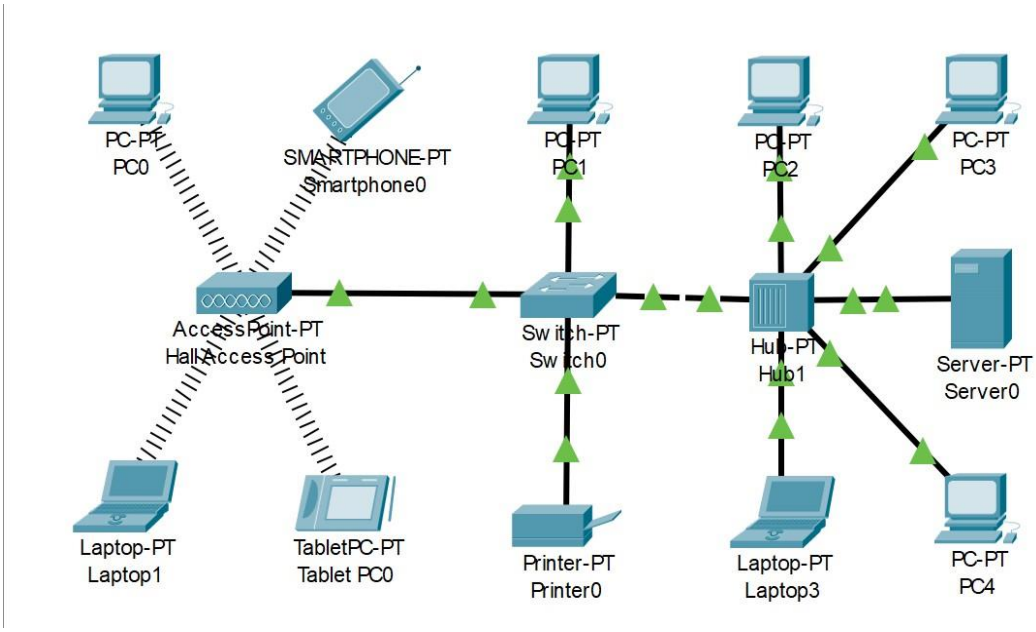
Топология 1



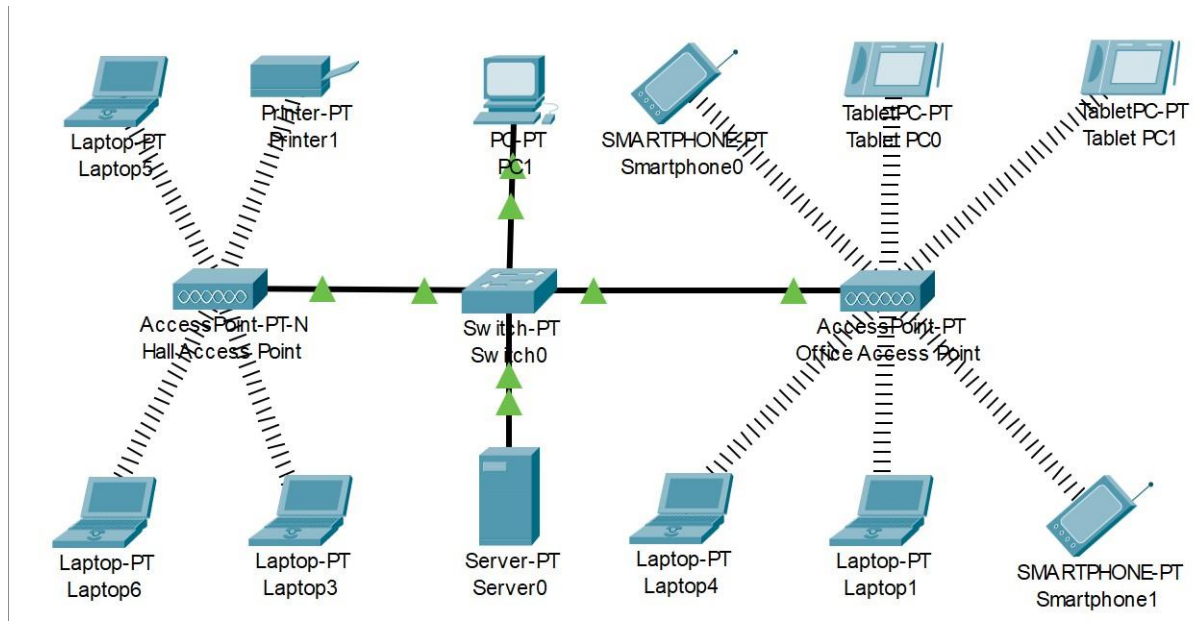
Топология 2



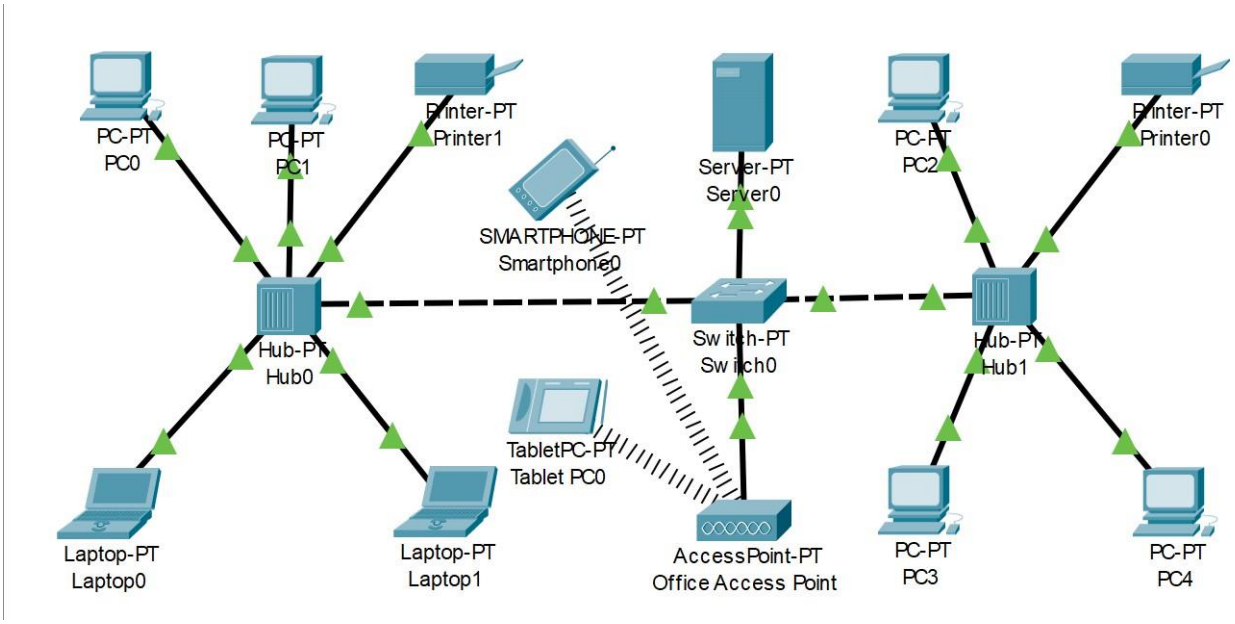
Топология 3



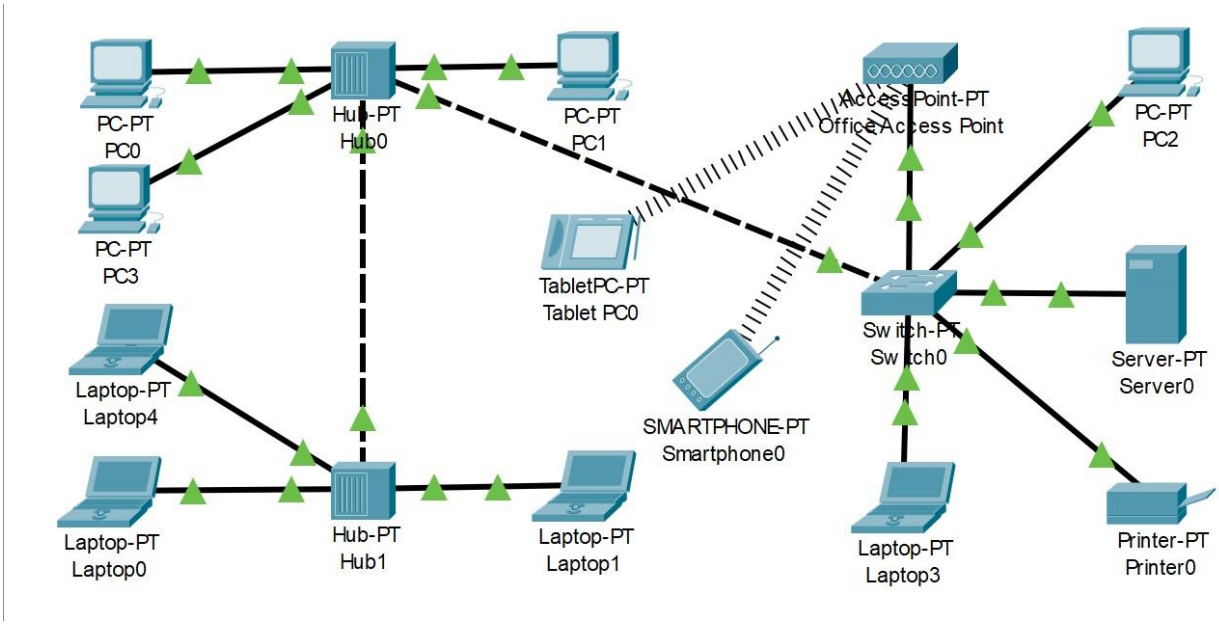
Топология 4



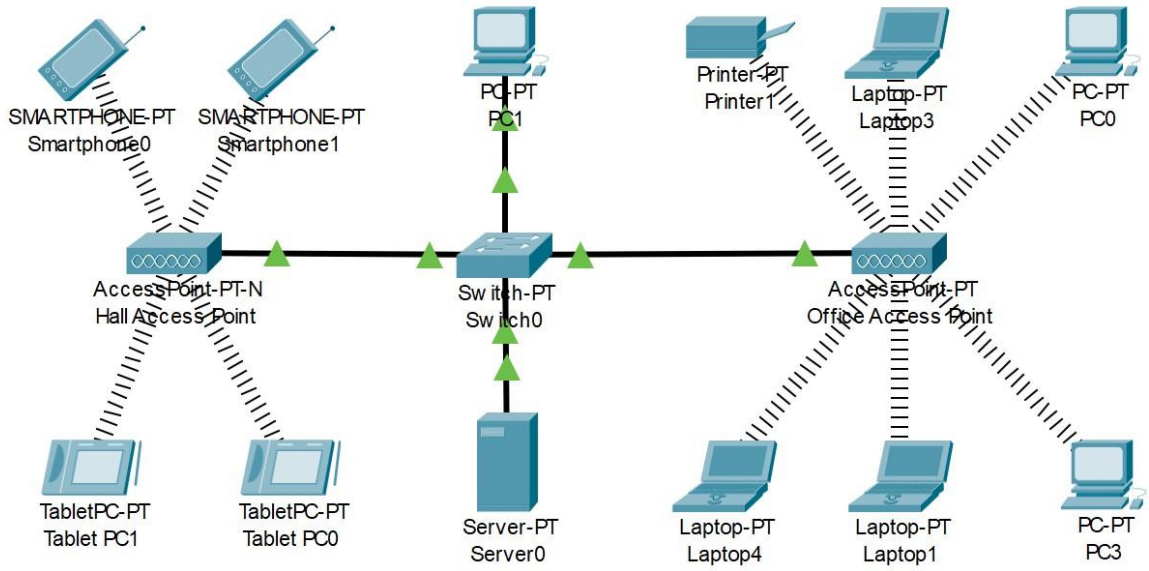
Топология 5



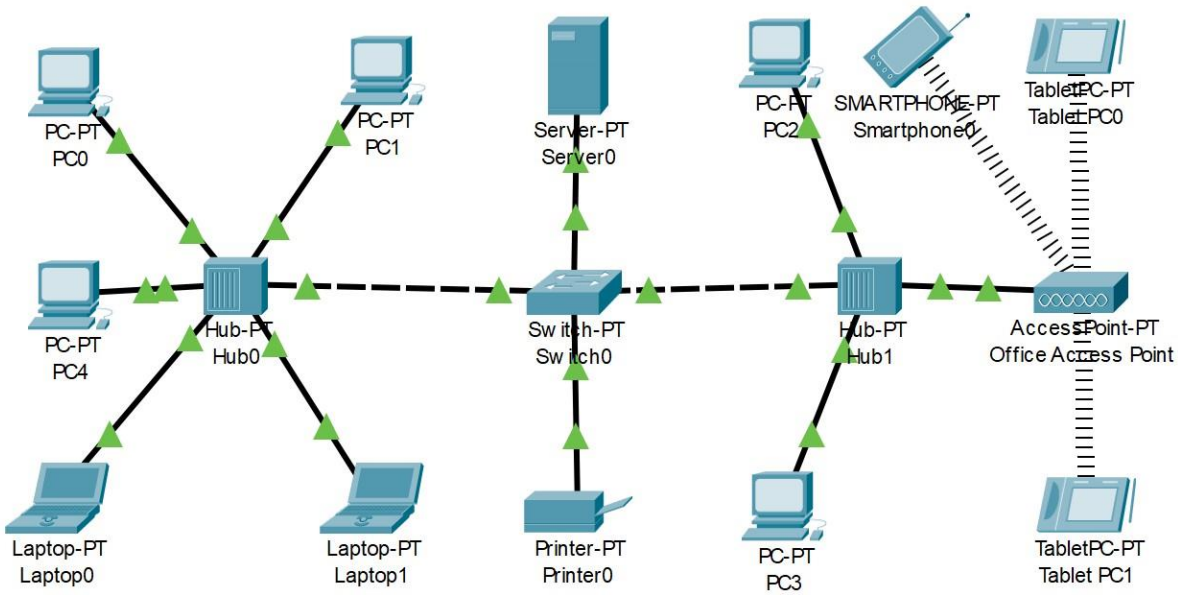
Топология 6



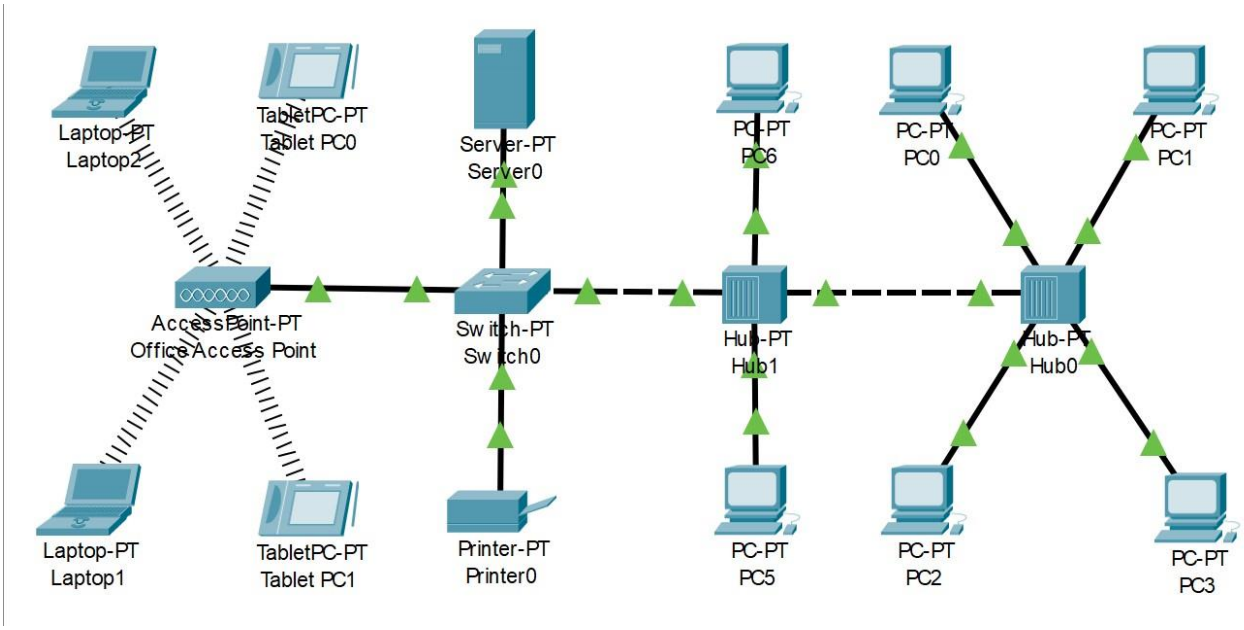
Топология 7



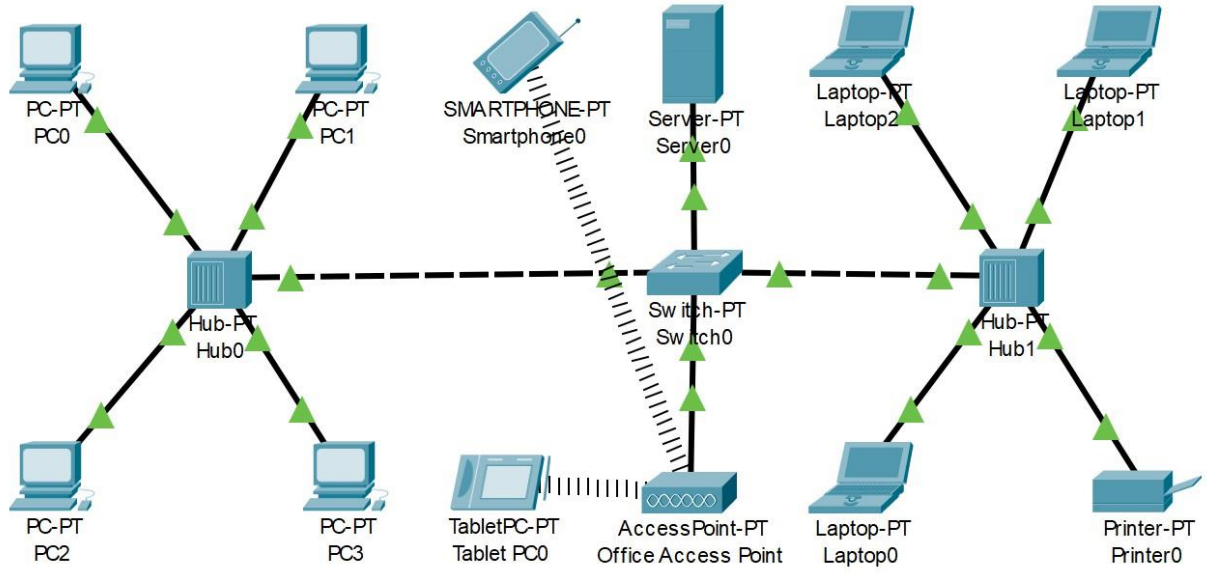
Топология 8



Топология 9



Топология 10



Контрольные вопросы

1. Опишите назначение маски подсети?

ПРОЕКТИРОВАНИЕ ЛОКАЛЬНОЙ СЕТИ С ПРИМЕНЕНИЕМ СТАТИЧЕСКОЙ МАРШРУТИЗАЦИИ В CISCO PACKET TRACER

Маршрутизация (англ. *Routing*) – процесс определения маршрута следования информации в сетях связи. Маршруты могут задаваться административно (статические маршруты) либо вычисляться с помощью алгоритмов маршрутизации, базируясь на информации о топологии и состоянии сети, полученной с помощью протоколов маршрутизации (динамические маршруты).

Статическая маршрутизация – вид маршрутизации, при котором маршруты указываются в явном виде при конфигурировании маршрутизатора. Вся маршрутизация при этом происходит без участия каких-либо протоколов маршрутизации.

Настройка статической маршрутизации заключается в ручном вводе в таблицу маршрутизации всех маршрутов.

Целью данной лабораторной работы является приобретение практических навыков проектирования и моделирования работы сети, состоящей из нескольких подсетей, соединенных с помощью маршрутизаторов, а также настройка статических маршрутов.

Перед тем, как приступить к выполнению работы, прочитать следующий материал:

http://ciscotips.ru/subnetting-reasons

В процессе выполнения работы можно использовать калькулятор IP-сетей, например:

https://www.networkcenter.info/calcs/subnetcalc

Задание 4.1.

1. В программе Ciso Packet Tracer создать топологию сети согласно заданному варианту (см. табл. 2.5). Топологии сетей представлены в табл. 4.2.

2. Беспроводную сеть защитить по технологии WPA2-PSK на основе шифрования AES. Всем беспроводным устройствам назначить SSID.

3. Маршрутизаторы *Router0* и *Router1* соединить с помощью оптоволоконной линии (*Fiber*).

4. Разбить заданную исходную сеть (см табл. 2.5) на две подсети (сеть А и сеть В), рассчитав маску подсети. При необходимости воспользоваться калькулятором IP-сетей.

5. Назначить сетевым интерфейсам маршрутизаторов для сети А и сети В первые допустимые IP-адреса, рассчитанные в п. 4. Для сетевых интерфейсов сети С задать адреса согласно варианту (см. табл. 2.5).

6. Назначить в свойствах всем оконечным устройствам (компьютерам, принтерам, планшетам и т.п.) на закладке «*Config*» соответствующие IP-адреса, полученные в п. 4. В качестве шлюза по-умолчанию (*Default Gateway*) указать IP-адрес соответствующих интерфейсов маршрутизаторов, используемых в п. 5.

7. В свойствах каждого маршрутизатора на вкладке «*Config*» в подменю *ROUTING* выбрать пункт *Static*. Далее добавить статический маршрут. Например, для роутера сети А (*Router0*) в поле *Network* необходимо ввести адрес сети В, в поле *Mask* – маску сети В, а в качестве следующего перехода в поле *Next Hop* ввести IP-адрес роутера *Router1* сети С.

8. Проверить в свойствах всех устройств на закладке «*Desktop*» значок «*Command Prompt*» настройки каждого оконечного узла (команда *ipconfig*). Необходимо добиться возможности пересылки данных между всеми объектами сети.

9. По запросу преподавателя проверить доступность соединения между двумя оконечными устройствами с помощью протокола *ICMP* (используя команды *ping* и *tracert*).

Отчет по четвертой лабораторной работе

Выполнить задание 4.1 согласно варианту, взятому из табл. 2.5. Представить преподавателю:

1. Изображение топологии сети.

2. Результат выполнения команды *ipconfig* оконечного узла.

3. Результат выполнения команды *ping* между выбранными оконечными узлами сетей A и B.

4. Результат выполнения команды *tracert* между выбранными оконечными узлами сетей A и B.

Таблица 2.5

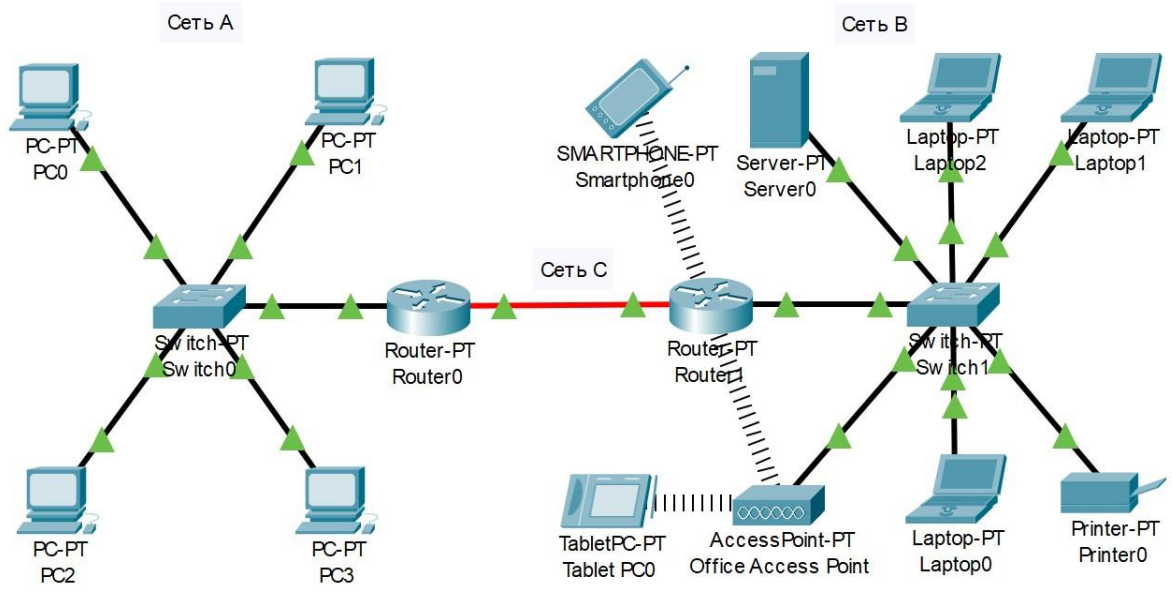
№ варианта	Топология сети (см. табл. 2.6)	Исходная сеть для разбиения на сеть A и сеть B	Адрес сети C
1	Топология 1	10.1.101.0	10.0.1.0/30
2	Топология 1	172.16.1.0	10.0.2.0/30
3	Топология 1	192.168.1.0	10.0.3.0/30
4	Топология 2	10.4.102.0	10.0.4.0/30
5	Топология 2	172.16.2.0	10.0.5.0/30
6	Топология 2	192.168.2.0	10.0.6.0/30
7	Топология 3	10.7.103.0	10.0.7.0/30
8	Топология 3	172.16.3.0	10.0.8.0/30
9	Топология 3	192.168.3.0	10.0.9.0/30
10	Топология 4	10.10.104.0	10.0.10.0/30
11	Топология 4	172.16.4.0	10.0.11.0/30
12	Топология 4	192.168.4.0	10.0.12.0/30
13	Топология 5	10.13.105.0	10.0.13.0/30
14	Топология 5	172.16.5.0	10.0.14.0/30
15	Топология 5	192.168.5.0	10.0.15.0/30
16	Топология 6	10.16.106.0	10.0.16.0/30

Продолжение табл. 2.5

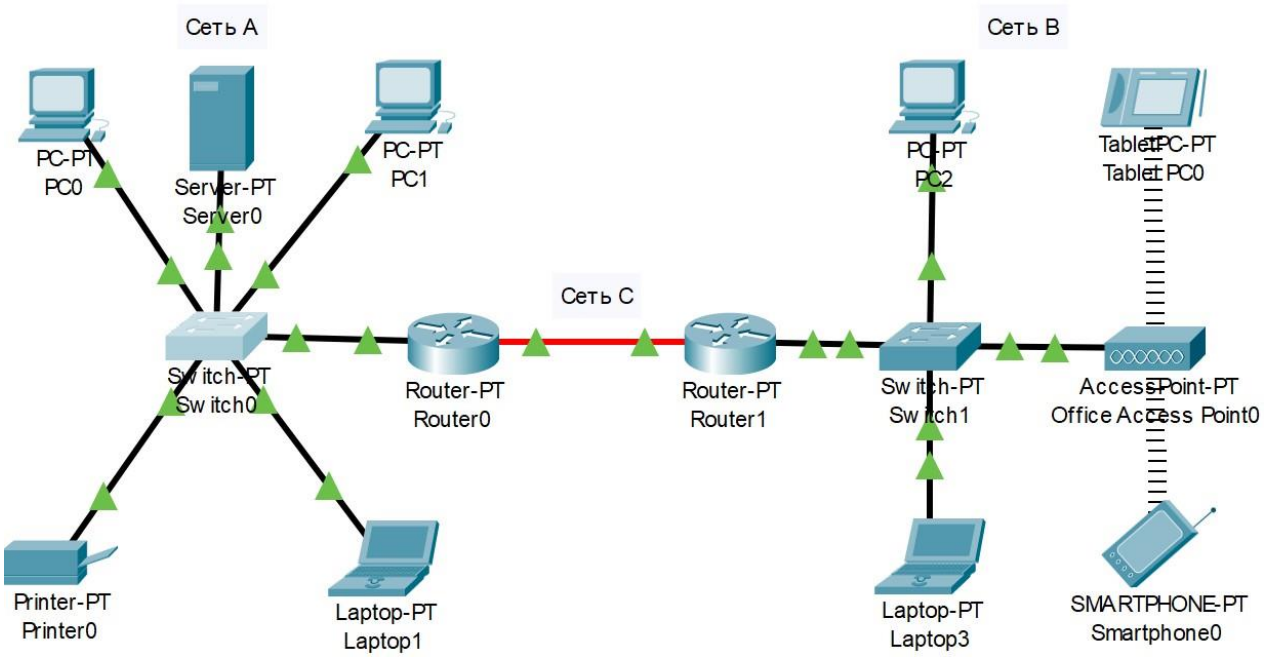
№ варианта	Топология сети (см. табл. 2.6)	Исходная сеть для разбиения на Сеть А и Сеть В	Адрес Сети С
17	Топология 6	172.16.6.0	10.0.17.0/30
18	Топология 6	192.168.6.0	10.0.18.0/30
19	Топология 7	10.19.107.0	10.0.19.0/30
20	Топология 7	172.16.7.0	10.0.20.0/30
21	Топология 7	192.168.7.0	10.0.21.0/30
22	Топология 8	10.22.108.0	10.0.22.0/30
23	Топология 8	172.16.8.0	10.0.23.0/30
24	Топология 8	192.168.8.0	10.0.24.0/30
25	Топология 9	10.25.109.0	10.0.25.0/30
26	Топология 9	172.16.9.0	10.0.26.0/30
27	Топология 9	192.168.9.0	10.0.27.0/30
28	Топология 10	10.28.110.0	10.0.28.0/30
29	Топология 10	172.16.10.0	10.0.29.0/30
30	Топология 10	192.168.10.0	10.0.30.0/30

Таблица 2.6

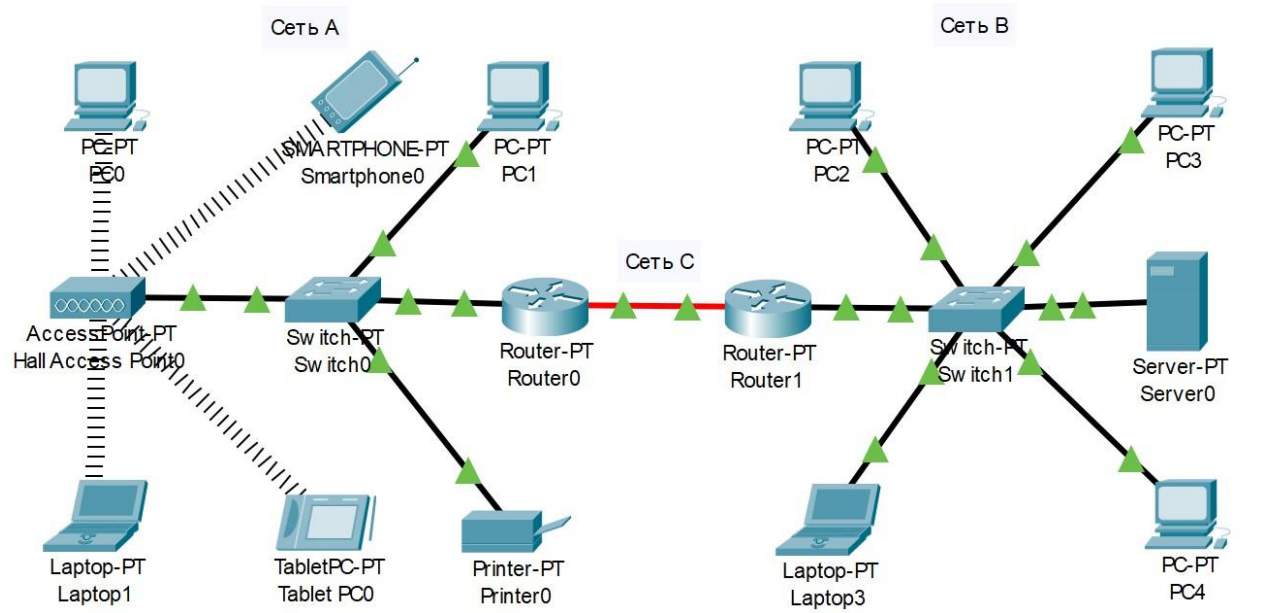
Топология 1



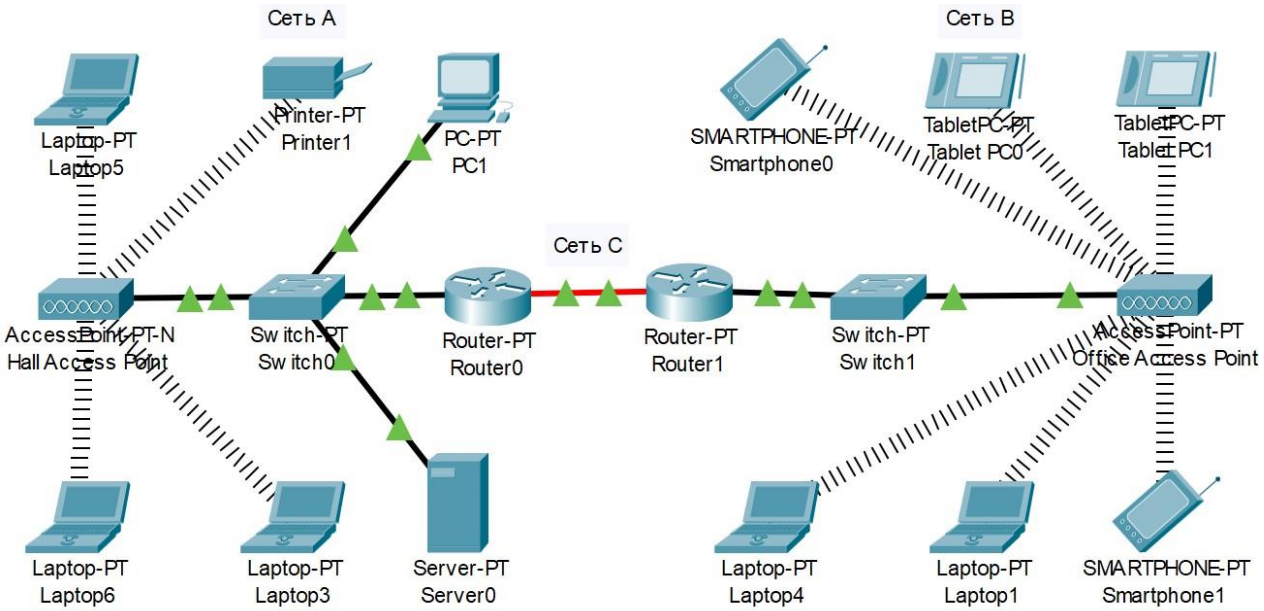
Топология 2



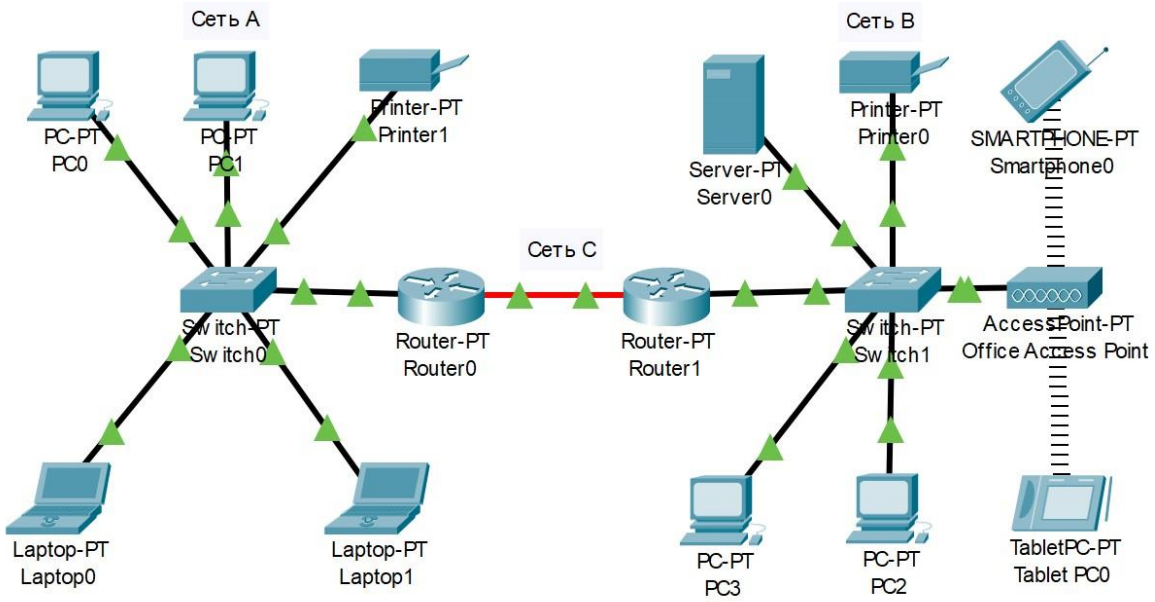
Топология 3



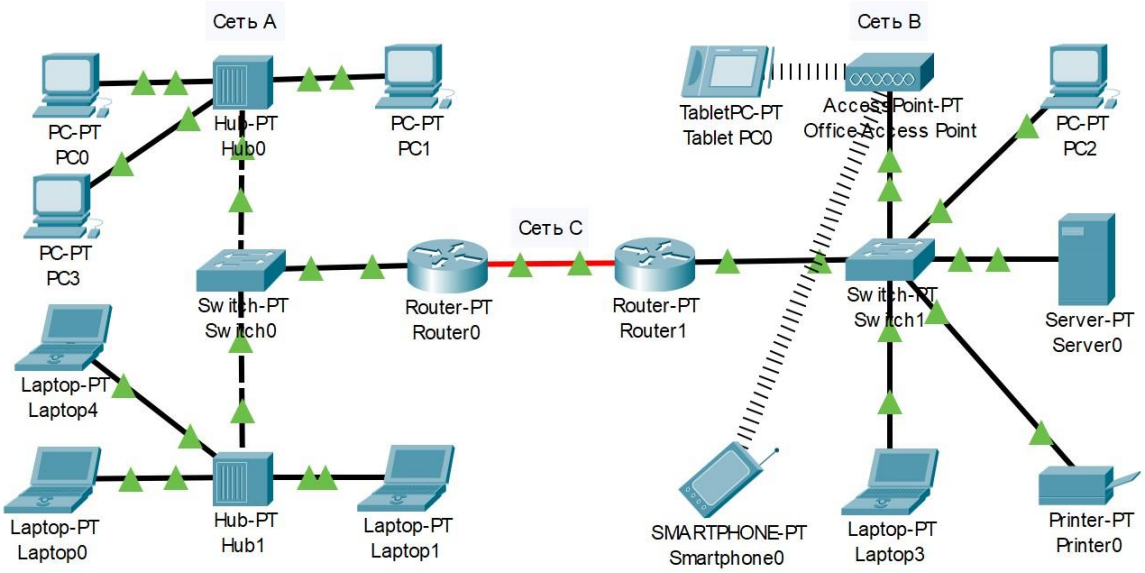
Топология 4



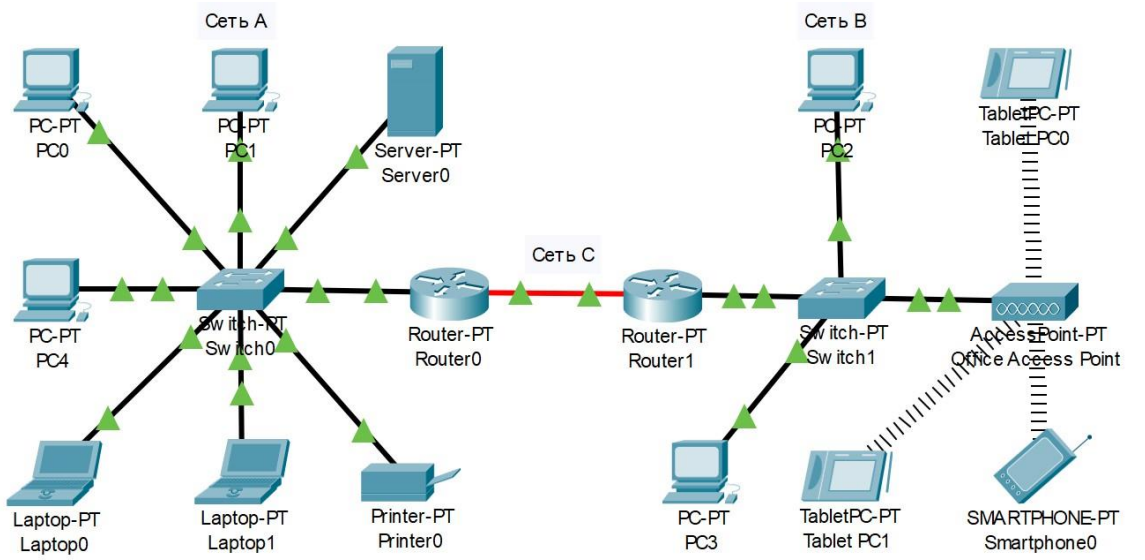
Топология 5



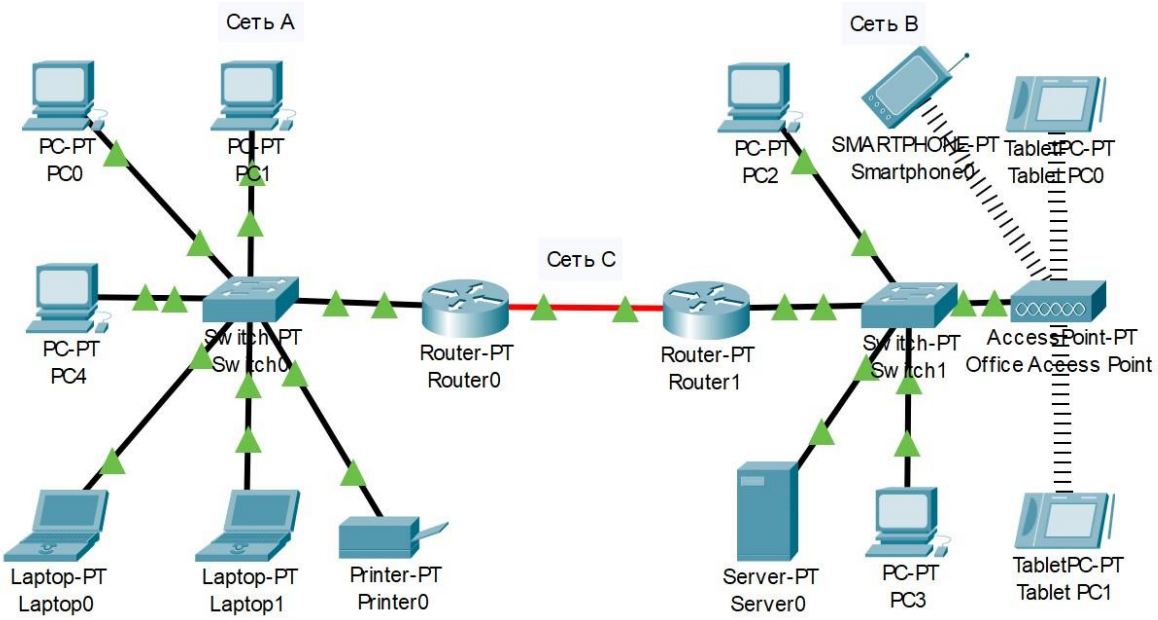
Топология 6



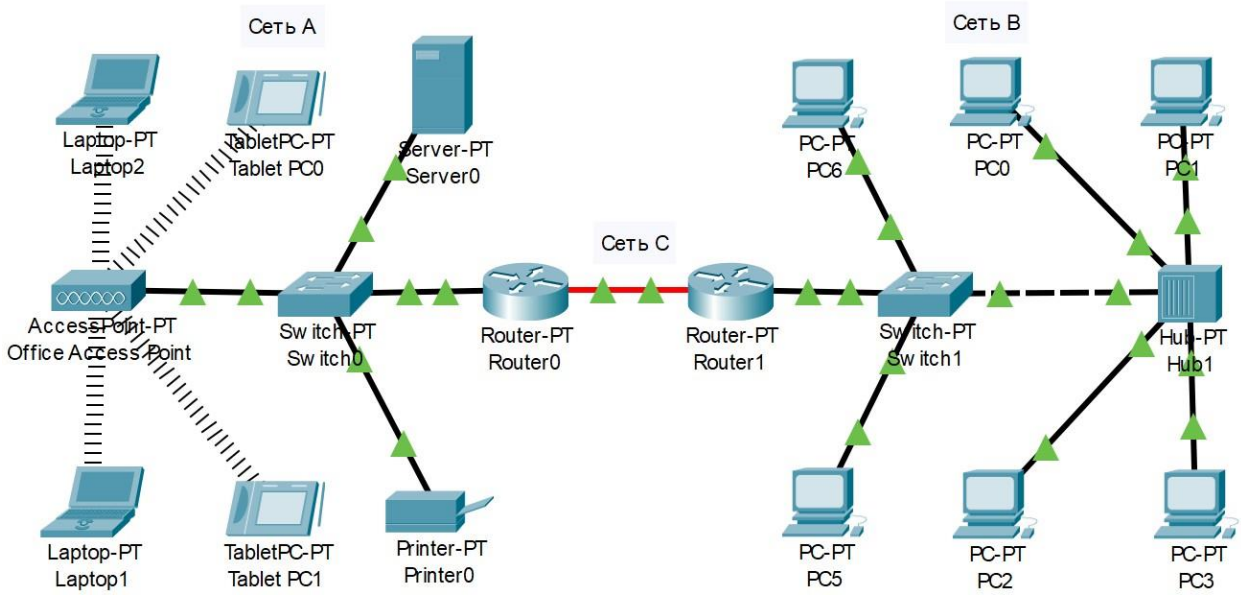
Топология 7



Топология 8

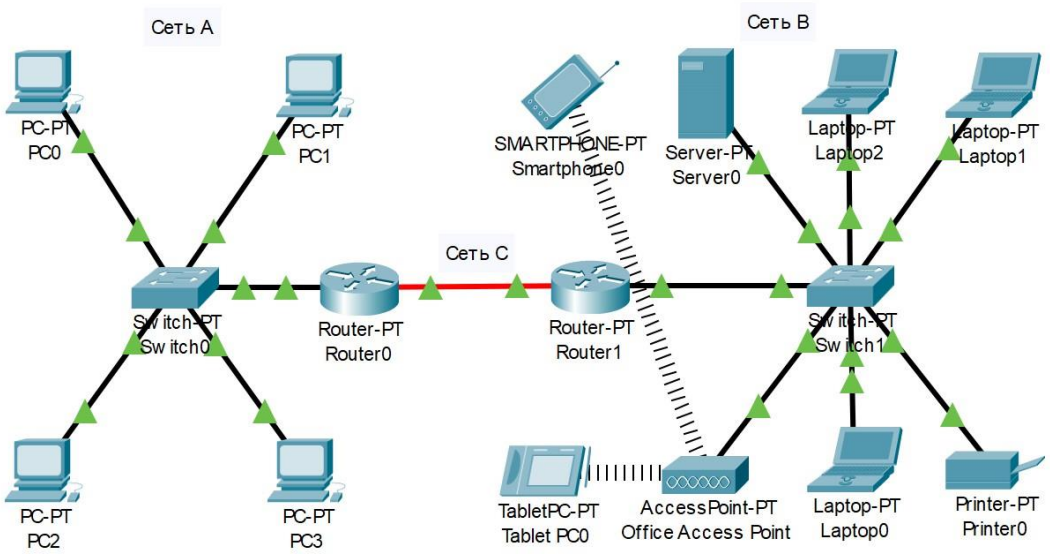


Топология 9



Окончание табл. 2.6

Топология 10



Контрольные вопросы

1. В чем состоит основное назначение DNS?

ЗАКЛЮЧЕНИЕ

Создание Интернета решило самую сложную на сегодняшний день инженерную проблему в мире. Сегодня Интернет объединяет миллиарды устройств по всему миру с помощью огромного количества маршрутизаторов и канальных соединений. Для того чтобы создать столь масштабную систему, инженеры разбили ее на четыре отдельных уровня.

Канальный/физический уровень отвечает за всю сложную инженерную работу, благодаря которой данные могут осуществить переход с помощью различных технологий, таких как беспроводная сеть Wi-Fi, проводной Ethernet, оптоволоконные или спутниковые соединения.

Сетевой (IP) уровень отвечает за маршрутизацию данных в условиях серии переходов. Он обеспечивает быструю и эффективную транспортировку данных от одного из миллиарда исходных устройств к любому из миллиардов конечных устройств. Сетевой уровень динамически корректирует и перенаправляет данные в зависимости от сетевой нагрузки, производительности канала или возникающих сбоев. Несмотря на эффективность этого уровня, иногда он может терять или даже отбрасывать данные. Сетевой уровень отвечает не за обеспечение общей надежности сети, а за обеспечение оптимальной маршрутизации и транспортировки данных.

Задача транспортного уровня – компенсировать недостатки сетевого и канального уровней. Он обеспечивает повторную передачу потерянных пакетов, а также упорядочение пакетов, пришедших не по порядку, перед их передачей в принимающее приложение. Также транспортный уровень отвечает за управление потоками между отправляющим и принимающим приложениями. Таким образом, он гарантирует, что при высокоскоростном и неперегруженном соединении данные будут перемещаться быстро, а при соединении с низкой скоростью или загруженном соединении – медленно. Посредством

регулирования потока данных и скорости передачи данных транспортный уровень обеспечивает бесперебойную работу Интернета даже при больших нагрузках.

Три перечисленных уровня призваны облегчать работу прикладного уровня. Благодаря этому приложению могут устанавливать сетевое соединение и отправлять/получать данные по этому соединению с помощью всего нескольких строк кода. Таким образом, приложения могут сосредоточиться на решении пользовательских задач. Стоит отметить, что благодаря вышеописанному подходу на сегодняшний день мы можем наблюдать широкий спектр самых разных приложений, для внедрения которых не требуется изменение основных интернет-протоколов. Без разделения Интернета на уровни было бы намного сложнее построить и развернуть постоянно совершенствующиеся версии сети. Если бы каждому приложению приходилось решать задачи всех четырех уровней, богатство и разнообразие сетевых приложений бы значительно сократилось. Прогресс, достигнутый за 50 лет постоянного совершенствования Интернета, действительно удивляет. Но относительно создания сетевых приложений мы все еще находимся в начале пути. Нетрудно представить себе Интернет, в котором каждый выключатель, лампочка, холодильник, стол, автомобиль, дорога, дрон и стул имеют свой интернет-адрес, и всем им необходимо взаимодействовать друг с другом. Таким образом, на горизонте возникают новые инженерные задачи, для решения которых, возможно, потребуется добавить в существующую интернет-архитектуру новые уровни. Однако, если у инженеров прошлого получилось разработать сетевые протоколы, позволяющие взаимодействовать между собой даже не сотням, а миллиардам устройств, подключенных к сети, то нынешние и будущие инженеры обязательно справятся со всеми проблемами, возникающими на их пути к обеспечению взаимодействия между собой триллионов устройств.

Жизнь в эпоху инфокоммуникаций откладывает отпечаток на личности, привычках, желаниях и возможностях. Мы используем технологии обмена информацией очень интенсивно. Многие сегодня уже не могут себя предста-

вить без мобильного телефона в руке, ноутбука в сумке, смарт-часов на запястье. Следует упомянуть важность вопросов анонимности в сети и обеспечения информационной безопасности при осуществлении информационного обмена для каждой личности, общества в целом и государства. Мы порой упускаем их из вида, но на самом деле должны постоянно думать, каким образом информация, передаваемая нами по открытым сетям связи, может быть использована с не совсем положительной стороны.

ГЛОССАРИЙ

Абстрактная модель – модель и набор терминов, которые используются для общего понимания проблемной области и руководства разработкой стандартов и реализаций для решения проблем.

Адрес – уникальный номер, который присваивается устройству и позволяет маршрутизировать сообщения на него.

Асимметричное шифрование – подход к шифрованию, при котором один (открытый) ключ используется для шифрования данных перед передачей, а другой (закрытый) ключ используется для дешифрования данных после их получения.

Базовая станция – название первого маршрутизатора, который обрабатывает пакеты по мере их передачи в Интернет.

Буферизация – временное хранение отправленных или полученных данных (до тех пор, пока устройство не убедится, что данные больше не понадобятся).

Веб-браузер – клиентское приложение, которое вы запускаете на своем устройстве для получения и отображения веб-страниц.

Веб-сервер – приложение, которое доставляет (обслуживает) веб-страницы.

Волоконно-оптический кабель – технология передачи данных, которая кодирует данные с помощью света, передаваемого по очень длинной нити из тонкого стекла или пластика. Оптоволоконные соединения работают быстро и могут покрывать очень большие расстояния.

Время жизни данных (TTL – Time To Live) – значение, содержащееся в каждом пакете, по мере прохождения пакета через маршрутизаторы уменьшающееся на единицу. Когда TTL достигает нуля, пакет отбрасывается.

Выделенный канал – постоянно активное соединение для отправки данных на большие расстояния, арендуемое у телефонной компании или другого подобного предприятия.

Глобальная сеть – сеть, покрывающая большие расстояния вплоть до возможности отправки сообщений по всему миру. Обычно такая сеть строится с использованием каналов связи, принадлежащих нескольким различным организациям.

Дешифрование – преобразование зашифрованного сообщения в обычное текстовое сообщение с использованием ключа безопасности.

Доменное имя – имя, присвоенное в домене верхнего уровня. Например, khanacademy.org – это домен, который назначается в домене верхнего уровня «.org».

Закрытый ключ – часть пары ключей, которая используется для дешифрования данных.

Зашифрованный текст – закодированная версия сообщения, которую нельзя прочитать, не зная ключа и техники дешифрования.

Идентификатор хоста – часть IP-адреса, которая используется для идентификации устройства в локальной сети.

Клиент – в сетевом приложении является запрашивающей или инициирующей соединение стороной.

Код состояния – один из аспектов HTTP, который призван указывать на выполнение или невыполнение запроса. Самый известный код состояния HTTP – «404». С его помощью HTTP-сервер сообщает HTTP-клиенту (т.е. браузеру), что запрошенный документ не может быть найден.

Локальная сеть – сеть, охватывающая ограниченную возможностью прокладки проводов или мощностью радиопередатчика область.

Магистральный маршрутизатор – маршрутизатор, который пересылает трафик в ядре Интернета.

Маршрутизатор – специализированное устройство, предназначенное для приема входящих пакетов по нескольким каналам и быстрой транспортировке пакетов по наиболее оптимальному исходящему каналу в целях ускорения их доставки.

«Маршрутный вихрь» – сбой, при котором образуется бесконечный цикл передачи пакетов между некоторым количеством маршрутизаторов. Возникает при сбоях в таблицах маршрутизации.

Модель реализации – модель и набор терминов, которые используются для руководства при разработке стандартов и реализации для решения конкретной проблемы.

Номер сети – часть IP-адреса, используемая для определения локальной сети, к которой подключено устройство.

Обычный текст – сообщение, которое будет зашифровано перед отправкой.

Открытый ключ – часть пары ключей, которая используется для шифрования данных.

Относительный адрес – относительное положение пакета в сообщении или потоке данных.

Пакет – фрагмент более крупного сообщения ограниченного размера. Большие сообщения или файлы разбиваются на множество пакетов, каждый из которых впоследствии отправляется отдельно. Обычно максимальный размер пакета составляет от 1000 до 3000 знаков.

Переход – физический участок сети. Как правило, на пути от исходного к конечному устройству пакеты совершают несколько переходов.

Пограничный маршрутизатор – маршрутизатор, обеспечивающий соединение между локальной и глобальной сетями (то же, что и шлюз).

Поддомен – имя, созданное «под» доменным именем. Например, «umich.edu» – это доменное имя, а «www.umich.edu» и «mail.umich. edu» являются поддоменами в «umich.edu». TLD (top-level domain) – домен верхнего уровня. Правая часть доменного имени. Примеры TLD: «.com», «.org» и «.ru». Недавно были добавлены новые домены верхнего уровня, такие как «.club» и «.help».

Подтверждение получения пакета – отправка конечным устройством на исходное устройство сообщения о том, что данные были получены обратно.

Порт – технология, позволяющая множеству различных серверных приложений ожидать входящих подключений на одном устройстве. Каждое приложение прослушивает отдельный порт. Клиентские приложения устанавливают соединения с хорошо известными номерами портов, чтобы убедиться, что они обращаются к правильному серверному приложению.

Прослушивание – процесс, начинающийся, когда серверное приложение запущено и готово принимать входящие соединения от клиентских приложений.

Размер группы сетевых пакетов – объем данных, который исходное устройство может отправить до ожидания подтверждения.

Регистратор – компания, которая может регистрировать, продавать и размещать доменные имена.

Секретный ключ коллективного использования – подход к шифрованию, подразумевающий использование одного и того же ключа для шифрования и дешифрования.

Сервер – в сетевом приложении отвечает на запросы или ожидает входящих подключений.

Сеть с двумя соединениями – сеть, которой характерно наличие как минимум двух возможных маршрутов между любой парой узлов в сети. При разрыве одного из соединений в такой сети общее соединение не будет затронуто.

Сеть передачи данных с промежуточным хранением – сеть, в которой при отправке данных с одного устройства на другое сообщение сохраняется на промежуточном устройстве до тех пор, пока для него не станет доступным исходящее сетевое соединение (обычно этот период довольно долгий).

Сокет – программная библиотека, доступная на многих языках программирования, которая делает создание сетевого подключения и обмен данными почти таким же простым, как открытие и чтение файла на любом вашем устройстве.

Таблицы маршрутизации – информация, хранящаяся в каждом маршрутизаторе и позволяющая отслеживать, какой исходящий канал следует использовать для каждого номера сети.

Токен – метод, позволяющий нескольким устройствам совместно использовать один и тот же физический носитель без риска возникновения конфликтов. Перед отправкой данных каждому устройству необходимо дождаться получения токена.

Управление потоками – подход, при котором исходное устройство периодически замедляется с целью контроля перегрузки сети или конечного

устройства. Также такой подход позволяет исходному устройству увеличивать скорость передачи данных в тех случаях, когда сеть и конечное устройство способны обрабатывать более высокие скорости передачи данных.

Центр сертификации – организация, проверяющая достоверность открытых ключей безопасности и дающая им цифровую подпись.

Широковещательная передача – отправка пакета таким образом, чтобы ее смогли получить все станции, подключенные к локальной сети.

Шифрование – преобразование обычного текстового сообщения в зашифрованное с ключом безопасности.

Шлюз – маршрутизатор, подключающий локальную сеть к более широкой сети (например, Интернет). С помощью шлюзов устройства могут отправлять данные за пределы локальной сети.

DHCP (Dynamic Host Configuration Protocol) – протокол динамической настройки узла. Протокол, позволяющий портативным устройствам получать IP-адрес при смене местоположения.

DNS (Domain Name System) – система доменных имен. Система протоколов и серверов, которые позволяют сетевым приложениям искать доменные имена и получать соответствующий для него IP-адрес.

HTML (HyperText Markup Language) – язык гипертекстовой разметки. Язык текстового формата, в котором разметка текста происходит с помощью тегов с символами «меньше» и «больше». Например: This is nice .

HTTP (HyperText Transport Protocol) – гипертекстовый транспортный протокол. Протокол прикладного уровня, который позволяет веб-браузерам получать веб-документы с веб-серверов.

ICANN (Internet Corporation for Assigned Names and Numbers) – Корпорация по управлению доменными именами и IP-адресами. Назначает домены верхнего уровня для Интернета и управляет ими.

IMAP (Internet Message Access Protocol) – протокол доступа к электронной почте, позволяющий почтовым клиентам входить в систему и получать почту с почтовых серверов с поддержкой IMAP.

IP-адрес – адрес, позволяющий устройству подключаться к глобальной сети и взаимодействовать с другими устройствами, имеющими IP-адреса. Для упрощения маршрутизации в ядре Интернета IP-адреса разбиваются

на сетевые номера и идентификаторы хоста. Примером IP-адреса может быть «212.78.1.25».

ISO (International Organization for Standardization) – международная организация по стандартизации. Всемирная организация, разрабатывающая стандарты в области вычислений, сетей и многих других областях.

MAC-адрес – адрес, назначаемый сетевому оборудованию при производстве.

NAT (Network Address Translation) – преобразование сетевых адресов. Механизм, позволяющий использовать глобальный IP-адрес для множества устройств в одной локальной сети.

RIR (Regional Internet Registries) – региональные интернет-реестры. Реестры, примерно соответствующие пяти существующим на Земле континентам, определяют IP-адреса для основных географических зон мира.

Secure Sockets Layer (SSL) – слой защищенных сокетов. Протокол, который позволяет приложению запрашивать шифрование соединения транспортного уровня при прохождении через сеть. Является предшественником протокола транспортного уровня (TLS).

Telnet – простое клиентское приложение, которое устанавливает TCP-соединения с различными комбинациями адресов/портов и позволяет передавать введенные данные. На заре Интернета telnet использовался для удаленного входа на устройство через сеть.

Traceroute – команда, доступная во многих системах Linux/UNIX, функция которой состоит в приблизительном определении маршрута пакета. В системах Windows может носить название tracert.

Transport Layer Security (TLS) – протокол, который позволяет приложению запрашивать шифрование соединения транспортного уровня при прохождении через сеть. Является последователем слоя защищенных сокетов (SSL).

OSI (The Open Systems Interconnection model) – модель взаимодействия открытых систем. Семиуровневая модель, используемая в целях организации разработки различных подходов к сетевой архитектуре.

СПИСОК ЛИТЕРАТУРЫ

1. **Северанс, Чарльз Р.** Как работают компьютерные сети и интернет / Чарльз Р. Северанспер ; пер. с англ. П. М. Бомбаковой. – М. : ДМК Пресс, 2022. – 116 с.

2. **Трещев, И.** Сети и телекоммуникации. Для студентов / И. Трещев. – Издательские решения, 2018. – 97 с.

3. **Таненбаум, Э.** Компьютерные сети / Э. Таненбаум, Д. Уэзеролл. – 5-е изд. – СПб. : Питер, 2012. – 960 с.

4. **Одом, У.** Официальное руководство Cisco по подготовке к сертификационным экзаменам CCNA ICND2 200-101: маршрутизация и коммутация, акад. изд. / У. Одом ; пер. с англ. – М. : ООО «И. Д. Вильяме», 2015. – 736 с.

5. **Одом, У.** Официальное руководство Cisco по подготовке к сертификационным экзаменам CCENT/CCNA ICNDl 100-101, акад. изд. / У. Одом ; пер. с англ. – М. : ООО «И. Д. Вильяме», 2015. – 912 с.

6. **Кофлер, М.** Linux. Установка, настройка, администрирование / М. Кофлер. – СПб. : Питер, 2014. – 768 с.

7. **Коваленко, И. Н.** Компьютерные сети : конспекты лекций / И. Н. Коваленко. – Ростов н/Д, 2016. – 145 с.

СОДЕРЖАНИЕ

Учебное электронное издание

КОНКИНА Виктория Викторовна
БОРИСЕНКО Андрей Борисович
КОРОБОВА Ирина Львовна

СЕТИ И ТЕЛЕКОММУНИКАЦИОННЫЕ ТЕХНОЛОГИИ

Учебное пособие

Редактор Л. В. Комбарова
Графический и мультимедийный дизайнер Н. И. Кужильная
Обложка, упаковка, тиражирование Л. В. Комбарова

ISBN 978-5-8265-2632-3

Подписано к использованию 06.09.2023.
Тираж 50 шт. Заказ № 98

Издательский центр ФГБОУ ВО «ТГТУ»
392000, г. Тамбов, ул. Советская, д. 106, к. 14
Тел./факс (4752) 63-81-08.
E-mail: izdatelstvo@tstu.ru