Оглавление

1 Обзор симулятора Cisco Packet Tracer

Cisco Packet Tracer – это многофункциональная программа моделирования сетей, которая позволяет студентам экспериментировать с поведением сети и оценивать возможные сценарии. Packet Tracer предоставляет функции моделирования, визуализации, авторской разработки, аттестации и совместного сотрудничества, а также облегчает преподавание и изучение сложных технологических принципов.

1.1 Загрузка Cisco Packet Tracer

Для загрузки Packet Tracer, выполните следующие действия:

1. Зарегистрируйтесь по ссылке https://www.netacad.com/ru/courses/packet-tracer/introduction-packet-tracer.
2. Запишитесь на курс Introduction to Packet Tracer.
3. Завершите регистрацию в Сетевой академии.
4. Запустите курс Introduction to Packet Tracer.
5. Инструкции по загрузке находятся в материалах курса.

1.2 Интерфейс Cisco Packet Tracer

После запуска Cisco Packet Tracer, по умолчанию открывает главное окно программы. Данное окно содержит следующие области:

1. Главное меню. Данная панель содержит основные команды.
2. Основная панель инструментов. На этой панели отображаются значки ярлыков для наиболее часто используемых команд меню.
3. Панель общих инструментов. Эта панель предоставляет доступ к следующим часто используемым инструментам рабочей области: «Select», «Inspect», «Delete», «Resize Shape», «Place Note», «Drawing Palette», «Add Simple PDU», и «Add Complex PDU».
4. Логическое / физическое рабочее пространство и панель навигации. Позволяет переключаться между Физическим рабочим пространством и Логическим рабочим пространством с помощью вкладок на этой панели.
5. Рабочая область. Отображает созданную сеть, позволяет наблюдать за симуляциями и просматривать различную информацию и статистику.
6. Панель переключения режимов реального времени / симуляции. Позволяет переключаться между режимом реального времени и режимом симуляции с помощью вкладок на этой панели. Кроме того, она содержит часы, которые отображают относительное время в режиме реального времени и режиме моделирования.

7. Панель компонентов сети. В данной панели находятся устройства и соединения предназначенные для размещения в рабочей области. Панель содержит окно выбора типа устройства и окно выбора конкретного устройства.

8. Панель выбора типа устройства. На этой панели содержатся типы устройств и подключений, доступных в Packet Tracer.

9. Панель выбора для конкретного устройства. В этом поле выбираются устройства которые необходимо добавить в сеть и какие подключения установить.

10. Панель созданных пакетов. Эта панель управляет пакетами, которые были помещены в сеть во время сценариев моделирования.

1.3 Работа с объектами сети

Для размещения сетевого объекта на схеме необходимо выбрать в нижней панели инструментов его класс, а затем модель. Выбрав необходимое оборудование его можно перетащить в рабочую область или щелчком мышки указать место в рабочей области, куда следует его поместить.

Для соединения сетевых устройств необходимо выбрать класс «Соединительные кабели», далее выбрать необходимый тип кабеля (или выбрать «автоматическое определение»), указать начальное устройство, выбрать один из его сетевых портов, затем указать конечное устройство и один из его портов. В случае применения объекта «Автоматическое определение типа сетевого кабеля», порт и тип кабеля будут выбираться автоматически (номер порта будет выбираться в порядке возрастания).

Конфигурирование устройства производится по двойному щелчку на нем. В открывшемся окне пользователь может включить/выключить устройство (соответствующим тумблером на его изображении в области «Physical Device View»), изменить аппаратурную конфигурацию добавив или удалить модули, используя область MODULES.

Выбрав вкладку «Config» пользователь может задать некоторые конфигурационные параметры (например, настроить сетевой интерфейс, определить имя устройства и т.п.). На вкладке «CLI» предоставляется доступ к командному интерфейсу устройства (если он предусмотрен).

Для конечных устройств реализованы дополнительные вкладки. На вкладке «Desktop» расположены эмуляторы работы некоторых утилит рабочего стола (командная строка, интернет-браузер и т.п.). «Software/Services» - конфигурирование программного обеспечения, которое должно быть установлено на реально действующем конечном устройстве.

1.4 Пошаговая отладка передачи информации в исследуемой сети

Отладка исследуемой сети может производиться двумя способами: имитируя деятельность администратора с реальным оборудованием (Realtime) и с применением средств моделирования (Simulation).

В первом случае пользователь среды может выполнять необходимые действия над сетевыми объектами и принимать решения о функциональности собранной им сети. Во втором случае используются встроенные средства среды имитационного моделирования, которые позволяют пошагового наглядно продемонстрировать этапы передачи информации по сети.

Анализируемые задания по передачи данных по сети объединяются в сценарий. В среде допускает создавать несколько сценариев и переключаться между ними для анализа работы сети.

Для создания задания по передаче данных по протоколу ICPM (ping) используется кнопка «Add Simple PDU». Пользователь задает начальный сетевой узел (который будет генерировать данные) и конечный сетевой узел. В результате автоматически создается одно задание в текущем сценарии.

Для формирования передач данных по сети с указанием параметров передаваемой информации (протокол, порт и т.п.) используется кнопка «Add Complex PDU». Нажав на соответствующую кнопку в вертикальной панели пользователь должен указать протокол передачи, источник передаваемой информации и задать дополнительные параметры.

Результаты выполнения заданий по передаче данных отображаются в области сценариев. В режиме реального времени результаты выполнения заданий выводятся сразу же по окончании имитации. В случае, если пользователь попытается при создании простого задания указать устройство (источник или приемник), не имеющего настроенного сетевого интерфейса, то сразу будет выдано сообщение об ошибке.

Переключившись в режим пошагового выполнения пользователь получает возможность наглядно посмотреть каким образом передаются данные по сети. Переход к следующему шагу производится нажатием на кнопку «Next». Перейти к предыдущему шагу можно нажав на клавишу «Back». Нажав на кнопку «Play» запускается автоматический переход к следующему шагу. Кнопка «Reset simulation» – сбрасывает исследуемую сеть в исходное состояние. При выборе пакета в списке открывается окно в котором содержится значения полей PDU всех уровней модели OSI.

2 Конфигурирования устройств Cisco

2.1 Интерфейс командной строки Cisco IOS

Интерфейс командной строки (CLI) Cisco IOS – основной интерфейс, используемый для конфигурирования, мониторинга и обслуживания устройств Cisco. Этот пользовательский интерфейс позволяет непосредственно выполнять команды Cisco IOS с помощью консоли маршрутизатора, терминала или с использованием удаленного доступа.

Чтобы облегчить конфигурирование устройств Cisco, интерфейс командной строки Cisco IOS разделен на отдельные командные режимы. В каждом командном режиме предусмотрен собственный набор команд для конфигурирования, обслуживания и мониторинга работы маршрутизатора и сети. Совокупность доступных в конкретный момент команд зависит от текущего командного режима. Ввод вопросительного знака (?) после системного приглашения позволяет вывести список доступных команд для каждого командного режима.

Применение определенных команд обеспечивает переход от одного командного режима к другому. Стандартный порядок, в котором пользователю следует осуществлять доступ к режимам, таков: пользовательский режим EXEC, привилегированный режим EXEC; режим глобальной конфигурации; режимы специальной конфигурации, подрежимы конфигурации и подрежимы конфигурации 2-го уровня.

Сеанс на маршрутизаторе обычно начинается в пользовательском режиме EXEC, который представляет собой один из двух уровней доступа режима EXEC. В целях безопасности в пользовательском режиме EXEC доступно лишь ограниченное подмножество команд EXEC. Этот уровень доступа предназначен для задач, не изменяющих конфигурацию маршрутизатора, например, определение статуса маршрутизатора.

Для получения доступа ко всем командам необходимо перейти в привилегированный режим EXEC, который обеспечивает второй уровень доступа режима EXEC. Обычно для входа в привилегированный режим EXEC требуется ввести пароль. В привилегированном режиме EXEC можно вводить любую команду EXEC.

Приведенный ниже пример демонстрирует процесс доступа к привилегированному режиму EXEC:

```
Router> enable
Password:<letmein>
Router#
```

Из привилегированного режима EXEC можно перейти в режим глобальной конфигурации. В этом режиме возможен ввод команд, позволяющих конфигурировать общие характеристики системы. Режим глобальной конфигурации может использоваться также для перехода в специфические режимы конфигурирования. Режимы конфигурирования, включая режим глобальной конфигурации, позволяют вносить изменения в текущую конфигурацию. Если конфигурация позднее сохраняется, то эти команды сохраняются после перезагрузки маршрутизатора.

В приведенном ниже примере показан процесс перехода в режим глобальной конфигурации из привилегированного режима EXEC:

```
Router# configure terminal
Enter configuration commands, one per line. End with CNTL/Z.
Router(config)#
```

Команды, вводимые в режиме глобальной конфигурации, при вводе изменяют текущую конфигурацию. Иными словами, изменения конфигурации вступают в силу при каждом нажатии клавиши Enter или Return после ввода правильной команды. Тем не менее эти изменения не сохраняются в файле конфигурации запуска, пока не будет введена команда режима EXEC:

```
Router# copy running-config startup-config
```

Из режима глобальной конфигурации можно перейти во множество режимов конфигурации, специфических для конкретного протокола или функции. В следующем примере пользователь входит в режим конфигурирования интерфейса FastEthernet0/0. Новое приглашение hostname(config-if)#, указывает на режим конфигурирования интерфейса.

```
Router(config)# interface FastEthernet0/0
Router(config-if)#
```

Из режима конфигурирования интерфейса можно перейти в режим конфигурирования субинтерфейса. В режиме конфигурирования субинтерфейса можно задавать параметры множества виртуальных интерфейсов (они называются субинтерфейсами) на единственном физическом интерфейсе.

В следующем примере задаются параметры субинтерфейса. Субинтерфейс получает обозначение "0.1", указывающее на то, что это субинтерфейс 0 линии 2. Приглашение hostname(config-subif)#, указывает на конфигурирование субинтерфейса.

```
Router(config)# interface FastEthernet0/0
Router(config-if)# interface FastEthernet0/0.1
Router(config-subif)#
```

2.2 Дополнение частичного имени команд

Если не удается запомнить полное имя команды или хотелось бы сократить количество вводимых символов, можно вводить первые несколько букв команды и затем нажать клавишу Tab. Синтаксический анализатор командной строки дополнит команду, если введенная строка уникальна для данного командного режима.

Интерфейс командной строки распознает команду в том случае, если введено достаточно символов, чтобы сделать команду уникальной. Например, при вводе conf в привилегированном режиме EXEC интерфейс командной строки сможет ассоциировать введенные символы с командой configure, так как только команда configure начинается с conf. В следующем примере CLI распознаёт уникальную для привилегированного режима EXEC строку conf при нажатии клавиши Tab:

```
Router# conf<Tab>
Router# configure
```

При использовании функции дополнения команды интерфейс командной строки отображает полное имя команды. Команда не выполняется, пока не будет нажата клавиша Enter или Return. Благодаря этому есть возможность изменить команду, если полная команда – это не то, что требовалось ввести с помощью сокращения. Если введена совокупность символов, которые могут обозначать более одной команды, система выдает сообщение, указывающий на то, что строка не уникальна.

Если CLI не может дополнить команду, можно ввести вопросительный знак (?), чтобы получить список команд, начинающихся с этой совокупности символов. Не оставляйте пробел между последней введенной буквой и вопросительным знаком (?).

Например, при вводе co? будет выведен список всех команд, доступных в текущем командном режиме:

```
Router# co?
configure connect copy
Router# co
```

Обратите внимание на то, что символы, введенные до вопросительного знака, отображаются на экране, чтобы дать возможность закончить ввод команды.

2.3 VLAN

В терминологии Cisco определяется два типа VLAN портов:

1. access – порт принадлежащий одному VLAN и передающий нетегированный трафик

2. trunk – порт передающий тегированный трафик одного или нескольких VLAN.

Dynamic Trunk Protocol (DTP) — проприетарный протокол Cisco, который позволяет коммутаторам динамически распознавать настроен ли соседний коммутатор для поднятия транка и какой протокол использовать (802.1Q или ISL). Включен по умолчанию.

Создание VLAN с идентификатором 2 и задание имени для него:

```
Switch(config)# vlan 2
Switch(config-vlan)# name test
```

Удаление VLAN с идентификатором 2:

```
Switch(config)# no vlan 2
```

Настройка access портов

Задание access порта:

```
Switch(config)# interface fa0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 2
```

Просмотр информации о VLAN'ах:

```
Switch# show vlan brief
```

Настройка trunk портов

Задание trunk порта:

```
Switch(config)# interface fa0/22
Switch(config-if)# switchport mode trunk
```

Динамическое создание транков (DTP)

Режимы DTP на интерфейсе:

1. auto – порт находится в автоматическом режиме и будет переведён в состояние trunk, только если порт на другом конце находится в режиме on или desirable. Т.е. если порты на обоих концах находятся в режиме "auto", то trunk применяться не будет.

2. desirable – порт находится в режиме "готов перейти в состояние trunk"; периодически передает DTP-кадры порту на другом конце, запрашивая удаленный порт

перейти в состояние trunk (состояние trunk будет установлено, если порт на другом конце находится в режиме on, desirable, или auto).

3. nonegotiate – порт готов перейти в режим trunk, но при этом не передает DTP-кадры порту на другом конце. Этот режим используется для предотвращения конфликтов с другим не cisco оборудованием. В этом случае коммутатор на другом конце должен быть вручную настроен на использование trunk.

Перевести интерфейс в режим auto:

```
Switch(config-if)# switchport mode dynamic auto
```

Перевести интерфейс в режим desirable:

```
Switch(config-if)# switchport mode dynamic desirable
```

Перевести интерфейс в режим nonegotiate:

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

Проверить текущий режим DTP:

```
Switch# show dtp interface
```

Настройка разрешенных VLAN

По умолчанию в транке разрешены все VLAN. Можно ограничить перечень VLAN, которые могут передаваться через конкретный транк.

Указать перечень разрешенных VLAN для транкового порта fa0/22:

```
Switch(config)# interface fa0/22
Switch(config-if)# switchport trunk allowed vlan 1-2,10,15
```

Добавление ещё одного разрешенного VLAN:

```
Switch(config)# interface fa0/22
Switch(config-if)# switchport trunk allowed vlan add 160
```

Удаление VLAN из списка разрешенных:

```
Switch(config)# interface fa0/22
Switch(config-if)# switchport trunk allowed vlan remove 160
```

Native VLAN

В стандарте 802.1Q существует понятие native VLAN. Трафик этого VLAN передается нетегированным. По умолчанию это VLAN 1. Однако можно изменить это и указать другой VLAN как native.

Настройка VLAN 5 как native:

```
Switch(config-if)# switchport trunk native vlan 5
```

2.4 Маршрутизации между VLAN

Каждый VLAN обычно находится в своей собственной подсети, коммутаторы в основном работают на уровне 2 модели OSI, и поэтому они не проверяют логические адреса. Поэтому пользовательские узлы, расположенные в разных VLAN, не могут обмениваться данными по умолчанию. Во многих случаях может потребоваться подключение между пользователями, расположенными в разных VLAN. Это можно реализоваться с помощью маршрутизации между VLAN.

Традиционная маршрутизации между VLAN

В этом типе маршрутизации между VLAN маршрутизатор подключается к коммутатору с использованием нескольких интерфейсов. Один для каждой VLAN. Интерфейсы на маршрутизаторе настроены как шлюзы по умолчанию для VLAN, настроенных на коммутаторе.

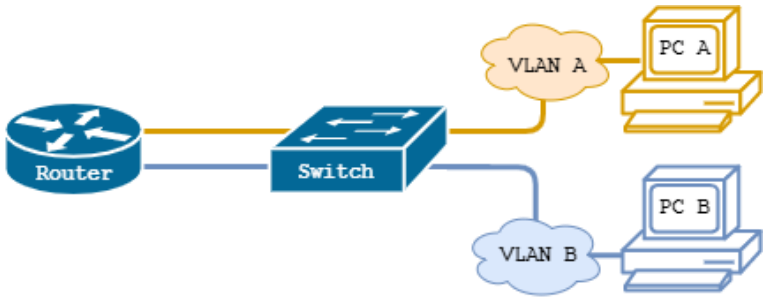

Рисунок 1 – Традиционная маршрутизации между VLAN

Порты, которые подключаются к маршрутизатору с коммутатора, настраиваются в режиме доступа в соответствующих VLAN.

Когда пользовательский узел отправляет сообщение пользователю, подключенному к другой VLAN, сообщение перемещается из его узла в порт доступа, который подключается к маршрутизатору в их VLAN. Когда маршрутизатор получает пакет, он проверяет IP-адрес назначения пакета и перенаправляет его в правильную сеть, используя порт доступа для VLAN назначения.

Настройка интерфейса Switch:

```
Switch(config)# interface fastEthernet0/1
Switch(config-if)# switchport mode access
Switch(config-if)# switchport access vlan 10
Switch(config-if)# exit
```

Настройка интерфейса Router:

```
Router(config)# interface fastEthernet0/1
Router(config-if)# ip address 10.1.1.1 255.255.255.0
Router(config-if)# exit
```

Router-on-a-stick маршрутизации между VLAN

Маршрутизатор подключается к коммутатору с помощью единого интерфейса. Порт коммутатора, подключенный к маршрутизатору, настроен как магистральный канал. Единственный интерфейс на маршрутизаторе тогда настроен с несколькими IP-адресами, которые соответствуют VLAN на коммутаторе. Этот интерфейс принимает трафик от всех VLAN и определяет сеть назначения на основе IP-адреса источника и назначения в пакетах. Затем он передает данные на коммутатор с правильной информацией VLAN.

На маршрутизаторе физический интерфейс делится на меньшие интерфейсы, называемые подынтерфейсами. Когда маршрутизатор получает помеченный трафик, он перенаправляет трафик на подинтерфейс, имеющий IP-адрес назначения. Подынтерфейсы не являются реальными интерфейсами, но они используют физические интерфейсы локальной сети на маршрутизаторе для пересылки данных в различные VLAN.

Пример настройки:

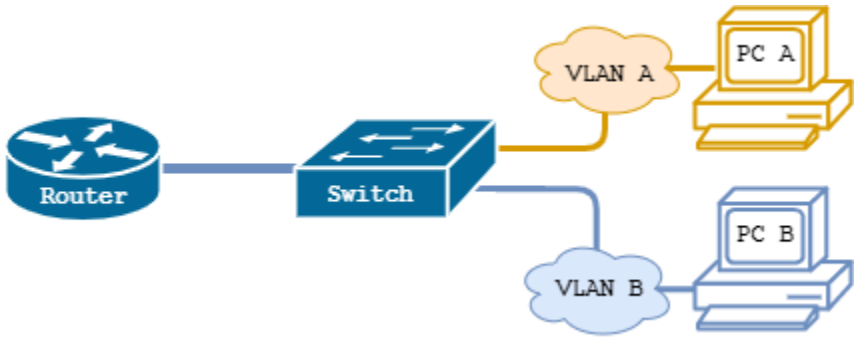

Рисунок 2 – Логическая схема сети

Настройка trunk интерфейса Switch:

```
Switch(config)# interface fastEthernet0/1
Switch(config-if)# switchport trunk encapsulation dot1q
Switch(config-if)# switchport mode trunk
Switch(config-if)# switchport trunk allowed vlan 1,10
Switch(config-if)# exit
```

Настройка субинтерфейса Router:

```
Router(config)# interface fastEthernet0/1.1
Router(config-subif)# ip address 10.1.1.1 255.255.255.0
Router(config-subif)# encapsulation dot1q 10
Router(config-subif)# exit
```

2.5 NAT

NAT (Network address translation) – технология трансляции сетевых адресов. Осуществляя трансляцию NAT, маршрутизатор изменяет IP-адрес отправителя в тот момент, когда пакет покидает частную сеть. Программное обеспечение Cisco IOS поддерживает несколько разновидностей трансляции NAT:

1. Статическая трансляция NAT – каждому частному IP-адресу соответствует один публичный IP. При использовании статической трансляции маршрутизатор NAT просто устанавливает взаимно однозначное соответствие между частным и зарегистрированным IP-адресом, от имени которого он выступает.

2. Динамическая трансляция NAT – преобразование внутренних IP-адресов во внешние происходит динамически. Создается пул возможных публичных IP-адресов и из этого пула динамически выбираются IP -адреса для преобразования.

3. Трансляция адресов портов PAT – транслирует сетевой адрес в зависимости от TCP/UDP-порта получателя.

Статическая трансляция адресов

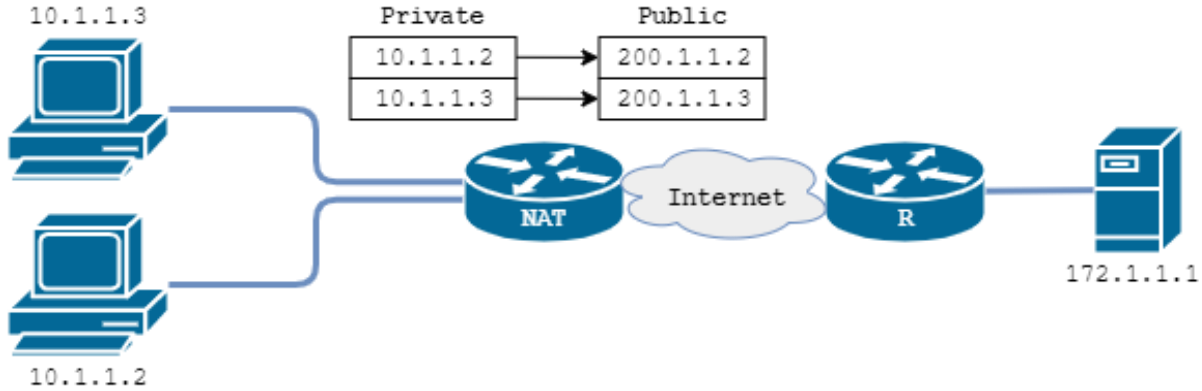

Рисунок 3 – Статическая трансляция адресов

Настройка статической трансляции NAT на оборудовании Cisco по сравнению с другими ее вариантами требует наименьших действий. При этом нужно установить соответствие между локальными (частными) и глобальными (открытыми) IP-адресами. Кроме того, необходимо указать маршрутизатору, на каких интерфейсах следует использовать трансляцию NAT, поскольку она может быть включена не на всех интерфейсах. В частности, маршрутизатору нужно указать каждый интерфейс и является ли он внутренним или внешним.

Внутренний интерфейс:

```
Router(config)# interface fastEthernet0/1
Router(config-if)# ip nat inside
```

Внешний интерфейс:

```
Router(config)# interface fastEthernet0/0
Router(config-if)# ip nat outside
```

Трансляция адреса в адрес:

```
Router(config)# ip nat inside source static 200.1.1.2 10.1.1.2
```

Трансляция адреса в адрес интерфейса:

```
Router(config)# ip nat inside source static 200.1.1.3 interface
fastEthernet0/1
```

Трансляция одной сети в другую (транслируется часть сети, а часть хоста сохраняется):

```
Router(config)# ip nat inside source static network 200.1.1.0 10.1.1.0
/24
```

Динамическая трансляция адресов

Динамический NAT использует пул публичных адресов и назначает их по принципу «первым пришел, первым обслужен». Когда внутреннее устройство запрашивает доступ к внешней сети, динамический NAT назначает доступный общедоступный IPv4-адрес из пула. Подобно статическому NAT, динамический NAT требует наличия достаточного количества общедоступных адресов для удовлетворения общего количества одновременных сеансов пользователя. Помимо динамического выделения внешних адресов, динамически NAT отличается от статического тем, что без отдельной настройки проброса портов уже невозможно внешнее соединение на один из адресов пула.

Например, на рисунке 7 установлен пул из 2 глобальных IP-адресов 200.1.1.2 и 200.1.1.3. Трансляция NAT настроена для преобразования всех внутренних локальных адресов 10.1.1.0/24.

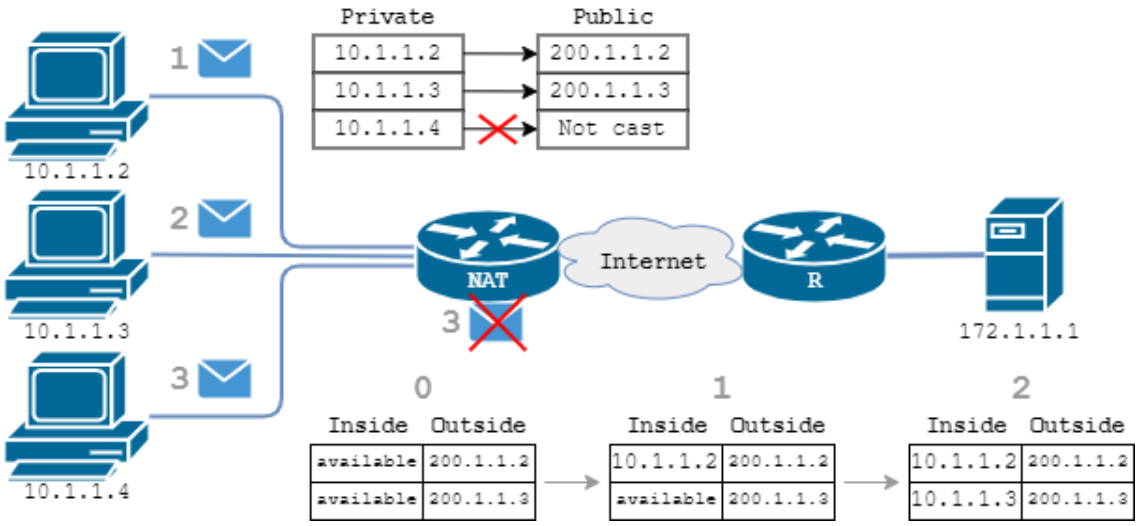

Рисунок 4 – Динамическая трансляция адресов

Настройка списка доступа соответствующего внутренним частным адресам:

```
Router(config)# access-list 1 permit 10.1.1.0 0.0.0.255
```

Создание пула адресов:

```
Router(config)# ip nat pool DYN 200.2.2.2 200.2.2.3 netmask 255.255.255.0
```

Настройка правила трансляции:

```
Router(config)# ip nat inside source list 1 pool DYN
```

Трансляция адресов портов PAT

С помощью PAT несколько адресов могут быть сопоставлены с одним или несколькими адресами, поскольку каждый частный адрес также отслеживается номером порта. Когда устройство инициирует сеанс TCP/IP, оно генерирует значение порта источника TCP или UDP для уникальной идентификации сеанса. Когда NAT-маршрутизатор получает пакет от клиента, он использует номер своего исходного порта, чтобы однозначно идентифицировать конкретный перевод NAT. Когда ответ возвращается с сервера, номер порта источника, который становится номером порта назначения в обратном пути, определяет, какое устройство маршрутизатор перенаправляет пакеты.

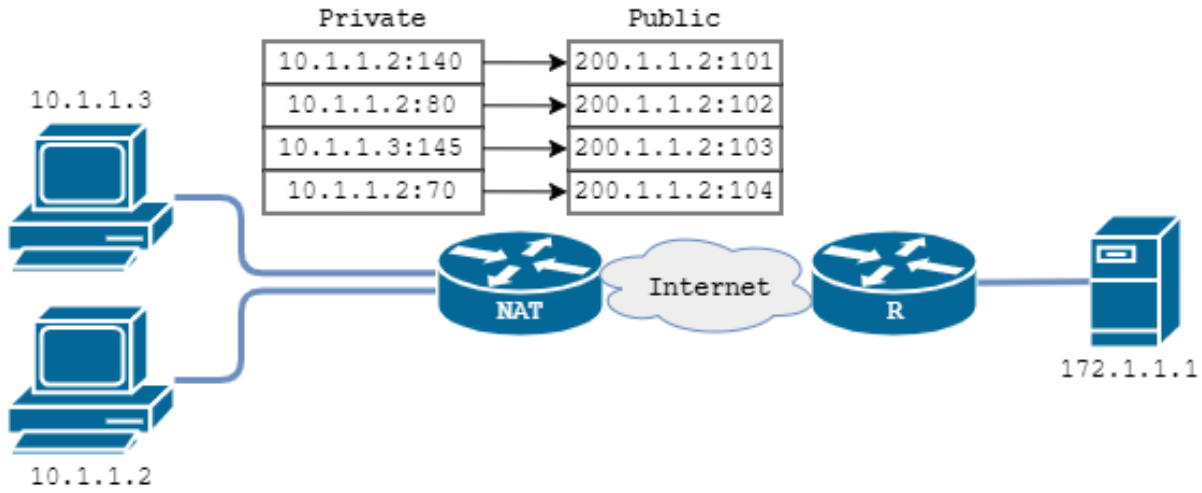

Рисунок 5 – Динамическая трансляция адресов

Настройка списка доступа соответствующего внутренним частным адресам:

```
Router(config)# access-list 1 permit 10.1.1.0 0.0.0.255
```

Настройка правила трансляции:

```
Router(config)# ip nat inside source list 1 interface fastethernet 0/1 overload
```

Дополнительные команды NAT

Ограничение максимального количества пользователей, за одним IP-адресом:

```
Router(config)# ip nat translation max-entries all-host 200 # 200 сессий на один адрес
```

Смена времени жизни (по умолчанию 24 часа):

```
Router(config)#ip nat translation tcp-timeout 1200
Router(config)#ip nat translation udp-timeout 30
Router(config)#ip nat translation icmp-timeout 5
```

Определение порта, отличного от общепринятого:

```
Router(config)#ip nat service list 10 ftp tcp port 2021 # ftp перебрасывается на нестандартный порт 2021
```

Фильтрация по определенному порту:

```
Router(config)#no ip nat service sip udp port 5060
Router(config)#no ip nat service sip tcp port 5060
```

Проверка работы NAT:

```
Router# show ip nat translations # Выводит активные преобразования
Router# show ip nat statistics # выводит статистику по преобразованиям
```

Логирование операций NAT:

```
Router(config)# ip nat log translations syslog
```

2.6 DHCP

DHCP (Dynamic Host Configuration Protocol) — сетевой протокол, позволяющий компьютерам автоматически получать IP-адрес и другие параметры, необходимые для работы в сети TCP/IP.

По умолчанию на маршрутизаторах Cisco включена функциональная возможность DHCP-сервера. Если же она не активирована, то это возможно сделать командой:

```
Router(config)# service dhcp
```

Отключить данную функциональную можно командой:

```
Router(config)# no service dhcp
```

Настройка пула DHCP

Настройка DHCP-пула на маршрутизаторе:

```
Router(config)# service dhcp
Router(config)# ip dhcp pool guest
Router(config-pool)# network 10.1.1.0 255.255.255.0
```

В качестве дополнительных параметров пула можно указать:

1. Шлюз по умолчанию:

```
Router(config-pool)# default-router 10.1.1.1
```

2. Имя домена для DHCP-клиента:

```
Router(config-pool)# domain-name mydomain.local
```

3. DNS-сервер:

```
Router(config-pool)# dns-server 10.2.5.11
```

4. Сервер WINS для сетей microsoft netbios:

```
Router(config-pool)# netbios-name-server 10.3.5.20
```

5. Дополнительные параметры конфигурации пула:

```
Router(config-pool)#option 33 ip 10.1.1.102 172.17.0.1 # Опция выдачи клиентам статических маршрутов без маски
Router(config-pool) #option 4 ip 10.1.1.12 # ip адрес Time Server сети
```

Исключение IP-адресов из пула:

```
Router(config)# ip dhcp excluded-address 10.1.1.1
Router(config)# ip dhcp excluded-address 10.1.1.2 10.1.1.10 #
Исключение диапазона адресов 10.1.1.2 - 10.1.1.10
```

Ручное резервирование IP-адресов:

```
Router(config-pool)#host 10.1.1.118 255.255.255.0
Router(config-pool)#client-identifier 0100.0476.106c.bc # 01 вначале MAC для клиентов Windows, 00 для клиентов Linux
Router(config-pool)#client-name Test
```

Дополнительные команды DHCP

Установка времени аренды IP-адреса от 0 до 365 дней:

```
Router(dhcp-config)# lease 5
```

Настройка перенаправления запросов DHCP на сервер:

```
Router(config)# interface fastEthernet0/0
Router(config-if)#ip helper-address 10.1.2.42 #
```

Настройка агента базы DHCP-сервера и параметров агента:

```
Router(config)# ip dhcp database url [timeout seconds | write-delay seconds]
```

Отключение агента базы данных DHCP-сервера:

```
Router(config)# no ip dhcp conflict logging
```

Очистка таблицы соответствия адресов выданных с пула DHCP-сервером:

```
Router#clear ip dhcp binding
```

Удаление привязки для конкретного IP-адреса:

```
Router# clear ip dhcp binding 10.84.168.181
```

Просмотр информации об адресах, которые были выданы DHCP-сервером:

```
Router# show ip dhcp binding
```

Просмотр информации о конфликтах, при назначении IP-адресов:

```
Router# show ip dhcp conflict
```

Просмотр статистики DHCP-сервера:

```
Router# show ip dhcp server statistics
```

Во время процесса назначения адреса DHCP-сервер проверяет, что предлагаемый адрес не используется. Для этого он до отправки ответа DHCP-клиенту посылает по предлагаемому адресу серию ping-пакетов.По умолчанию DHCP-сервер ждет две секунды ответ, а затем повторяет ICMP запрос.Эти параметры можно изменить.

Изменить количество отправляемых ICMP запросов (0 отключает ping):

```
ip dhcp ping packets <0-10>
```

Изменение таймаута между запросами:

```
Router(config)# ip dhcp ping timeout 300
```

2.7 ACL

ACL (Access Control List) – последовательный набор правил разрешающих или запрещающих прохождение пакета. Маршрутизатор последовательно проверяет каждый пакет на соответствие правилам. После первого же совпадения принимается решение и дальше правила не обрабатываются.

ACL разделяются на два типа:

1. Стандартные (Standard): могут проверять только адреса источников
2. Расширенные (Extended): могут проверять адреса источников, а также адреса получателей, в случае IP ещё тип протокола и TCP/UDP порты

Обозначаются списки доступа либо номерами, либо символьными именами. ACL также используются для разных сетевых протоколов. Обозначаются они следующим образом, нумерованные списки доступа:

– Стандартные: от 1 до 99

– Расширенные: от 100 до 199

Списки ACL создаются отдельно и потом присваиваются к интерфейсу. Существуют следующие правила применимые к спискам доступа:

1. Обработка ведется строго в том порядке, в котором записаны условия.
2. Если пакет совпал с условием, дальше он не обрабатывается.
3. В конце каждого списка доступа стоит неявный deny any (запретить всё).
4. Нельзя разместить более 1 списка доступа на интерфейс, на протокол, на направление.
5. ACL не действует на трафик, сгенерированный самим маршрутизатором.
6. Для фильтрации адресов используется WildCard (Обратная) маска.

Формат команды создания ACL:

```
Router(config)#ip access-list {standard | extended} {<номер ACL> | <имя ACL>}
```

standard – стандартный ACL

extended – расширенный ACL

Применение ACL на интерфейсе:

```
router(conf-if)# ip access-group <номер ACL>|<имя ACL> <in | out>
```

in – входящее направление

out – исходящее направление

Стандартные списки ACL

Стандартные ACL управляют трафиком, сравнивая адрес источника пакетов с адресами, заданными в списке. Формат синтаксиса команды стандартного ACL:

```
Router(config)#access-list <номер списка от 1 до 99> {permit | deny | remark} {address | any | host} [source-wildcard] [log]
```

permit – пропустить пакет. deny – отбросить пакет.

host – конкретный IP-адрес узла. address – адрес сети. any – любой хост.

source source-wildcard – инвертированная маска сети.

log – логирование пакетов проходящих через данную запись ACL

Пример стандартного нумерованного ACL, который запрещает хосту 10.0.3.2 доступ в сегмент сервера, но разрешает всем остальным:

```
Router(conf)# access-list 1 deny 10.0.3.2
Router(conf)# access-list 1 permit any
```

Применение ACL на интерфейсе (если применить ACL на интерфейсе fa0/0 в направлении in, то он заблокирует и доступ хоста 10.0.3.2 в интернет):

```
Router(conf)# interface fa0/1
Router(conf-if)# ip access-group 1 out
```

Тот же пример только с именованным ACL:

```
Router(conf)# ip access-list standard Test_stand
Router(config-std-nacl)# deny 10.0.3.2
Router(config-std-nacl)# permit any
```

Применение ACL на интерфейсе:

```
Router(conf)# interface fa0/1
Router(conf-if)# ip access-group Test_stand out
```

Расширенные списки ACL

Расширенный ACL позволяет фильтровать трафик по следующим параметрам:

– Адрес отправителя

– Адрес получателя

– TCP/UDP порт отправителя

– TCP/UDP порт получателя

– Протоколу, завёрнутому в ip (отфильтровать только tcp, udp, icmp, gre и т.п.)

– Типу трафика для данного протокола (для icmp отфильтровать только icmp-reply).

```
Router(config)#access-list <номер ACL от 100 до 199>|<имя ACL> {permit | deny | remark} protocol source [source-wildcard] [operator operand] [port <порт или название протокола> [established]
```

protocol source – фильтруемый протокол (ICMP, TCP, UDP, IP, OSPF и т.д)

operator:

A.B.C.D – адрес получателя

any – любой конечный хост

eq –только пакеты на этом порте

gt – только пакеты с большим номером порта

host – единственный конечный хост

lt – только пакеты с более низким номером порта

neq – только пакеты не на данном номере порта

range – диапазон портов

port – номер порта (TCP или UDP), можно указать имя

established – разрешение прохождения TCP-сегментов, которые являются частью уже созданной TCP-сессии

Пример расширенного нумерованного ACL:

```
Router(config)# access-list 100 deny tcp host 10.0.3.2 host 192.168.3.10 eq 3389
Router(config)# access-list 100 deny tcp host 10.0.3.1 any eq 80
Router(config)# access-list 100 permit ip 10.0.3.0 0.0.0.255 any
Router(config)# access-list 100 deny ip any any
```

Описание примера:

1. запретить хосту 10.0.3.2 доступ к серверу по RDP,
2. запретить хосту 10.0.3.1 доступ по HTTP,
3. разрешить всем остальным из сети 10.0.3.0 всё,
4. запретить всем остальным хостам (хотя в конце и так есть невидимое deny ip any any, его часто дописывают в конце правил, чтобы явно видеть по срабатыванию счетчиков, что трафик заблокировал ACL):

Применение ACL на интерфейсе:

```
R1(conf)# interface fa0/0
R1(conf-if)# ip access-group 100 in
```

2.8 PPPoE

PPPoE комбинирует два широко распространенных стандарта: Ethernet и PPP. Это позволяет использовать аутентификацию для раздачи клиентам IP адресов, что часто используется ISP провайдерами.

PPPoE позволяет использовать стандартный метод аутентификации используя PPP поверх сети Ethernet. Как уже было сказано, PPPoE позволяет аутентифицированную раздачу IP, при этом клиент PPPoE и сервер PPPoE подключены протоколом 2-го уровня над DSL.

Подключение по PPPoE проходи две фазы:

1. Active Discovery Phase. Клиент обнаруживает сервер PPPoE, который называется access concentrator. Во время этой фазы назначается Session ID и устанавливается PPPoE layer.

2. PPP Session phase. На этой фазе применяются установки PPP и происходит аутентификация. После того как соединение установлено, PPPoE работает на втором уровне. Пакеты инкапсулируются в заголовки PPPoE и передаются по PPP.

Конфигурация PPPoE сервера

Конфигурация сервера PPPoE включает следующие этапы:

1. Настройка локального IP-пула
2. Конфигурация группы широкополосного доступа
3. Создание виртуального шаблона
4. Настройки интерфейса виртуального шаблона
5. Назначение PPPoE BBA Group для интерфейса

Настройка локального IP-пула выполняется аналогично созданию пула IP на DHCP-сервере. В качестве примера создадим пул IP-адресов с именем ABCpool, первый IP-адрес будет 192.168.0.2, а последний - 192.168.0.254:

```
Server(config)# ip local pool ABCpool 192.168.0.2 192.168.0.254
```

Конфигурация группы широкополосного доступа (Broadband Access) - группа BBA обрабатывает сообщения сеанса PPPoE. В ней можно ограничить количество сеансов для определенных клиентов. Существует одна глобальная группа BBA, можно создать много локальных групп BBA. Пример создание Глобальной группы BBA с именем GGroup:

```
Server(config) # bba-group pppoe global GGroup
```

Создание виртуального шаблона для подключения клиентов:

```
Server(config-bba-group)# virtual-template 1
```

В созданном виртуальном шаблоне настроим IP-адрес 192.168.0.1/24 как адрес интерфейса. И зададим созданный ранее ABCpool пул IP-адресов по умолчанию.

```
Server(config-bba-group)# interface virtual-template 1
Server(config-if)# ip address 192.168.0.1 255.255.255.0
Server(config-if)# peer default ip address pool ABCpool
```

Назначение PPPoE BBA Group для интерфейса:

```
Server(config)# interface fastethernet 0/1
Server(config-if)# pppoe enable group GGroup
```

Конфигурация PPPoE клиента

На стороне клиента, нужно создать интерфейс номеронабирателя. Затем назначить пул номеронабирателя и определить инкапсуляцию как PPP. Установить параметр указывающий, что клиенту необходимо получать IP-адреса от сервера PPPoE с помощью команды ip address negotiated. Установить значение MTU, не превышающее длину Ethernet по умолчанию (1500). В примере установим MTU как 1492, с 8-байтовым заголовком PPP, общий заголовок будет равен 1500. Пример указанной ранее настройки:

```
Client(config)# interface dialer1
Client(config-if)# dialer pool 1
Client(config-if)# encapsulation ppp
Client(config-if)# ip address negotiated
Client(config-if)# mtu 1492
```

Дополнительные команды PPPoE

Установка аутентификации клиентов PPPoE:

```
Server(config-bba-group)# interface virtual-template 1
Server(config-if)#ppp authentication chap
```

Создание учетной записи клиента PPPoE

```
Server(config)# username one password 0 one
```

Сводная информация о сеансах PPPoE:

```
Client# show pppoe summary
```

Сводная информация о состоянии протокола PPPoE:

```
Client# show pppoe interfaces
```

Информация о предельном количестве сеансов PPPoE:

```
Client# show pppoe limits
```

Статистика пакетов полученных и отправленных сеансами PPPoE

```
Client# show pppoe statistics
```

2.9 VPN

VPN (Virtual Private Network) – обобщённое название технологий, позволяющих обеспечить одно или несколько сетевых соединений поверх другой сети. Несмотря на то, что коммуникации осуществляются по сетям с меньшим или неизвестным уровнем доверия (например по публичным сетям) уровень доверия к построенной логической сети не зависит от уровня доверия к базовым сетям благодаря использованию средств криптографии (шифрования, аутентификации, инфраструктуры открытых ключей, средств для защиты от повторов и изменений передаваемых по логической сети сообщений).

IPSec (IP Security) – группа протоколов, обеспечивающих работу сервисов конфиденциальности, аутентификации и проверки целостности передаваемой информации на сетевом уровне модели OSI (L3, Network).

IKE (Internet Key Exchange) – протокол, используемый для автоматического создания, установки соединения, изменения и удаления Security Associations (SA) между двумя хостами.

SA (Security Associations) – уникальный идентификатор конечной точки туннеля.

Рассмотрим Site-to-site VPN. Данный тип VPN соединения представляет собой интернет-WAN инфраструктуру, предназначенную для расширения сетевых ресурсов филиалов, домашних офисов и организаций деловых партнеров. Весь трафик, пересылаемый между узлами, шифруется с помощью протокола IPsec. Основное достоинство такого типа туннеля – это выделенный маршрутизирующий виртуальный интерфейс SVTI для VPN соединения, к которому можно применять различные политики, включая такие как QoS, uRPF, CBAC/ZBF, PBR, NAT, ACL и другие.

Пример настройки Site-to-site VPN с аутентификацией по pre-shared key

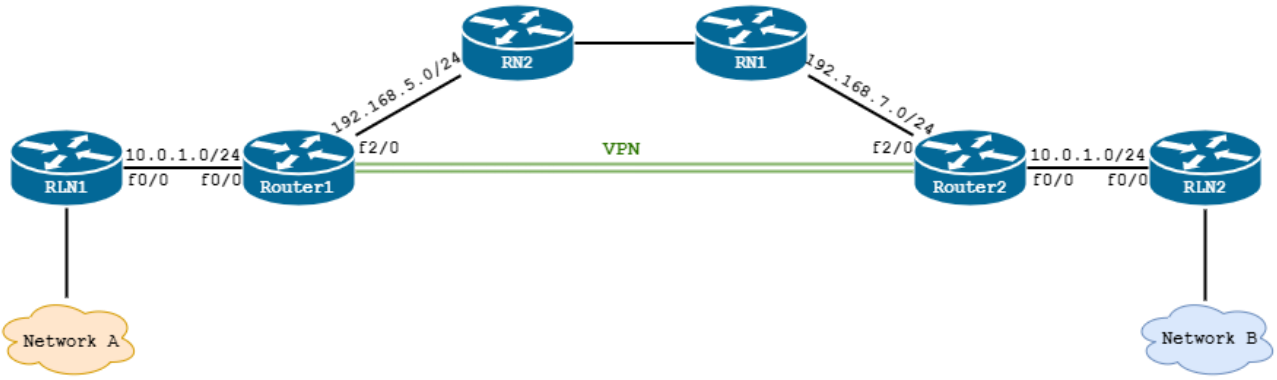

Рисунок 6 – Схема сети рассматриваемого примера

Активация лицензии Security Technology Package на Router1 и Router2:

```
Router(config)# license boot module c2900 technology-package securityk9
Router(config)# end
Router# copy running-config startup-config
Router# reload
```

Настройка политики IKE (ISAKMP). Политика IKE одинаковая на Router1 и Router2:

```
Router(config)# crypto isakmp policy 10
Router(config-isakmp)# encr aes
Router(config-isakmp)# authentication pre-share
Router(config-isakmp)# group 5
Router(config-isakmp)# hash sha
```

Настройка pre-shared ключ, который будет использоваться для аутентификации:

Настройка ключа на Router1:

```
Router(config)# crypto isakmp key cisco address 192.168.7.2
```

Настройка ключа на Router2:

```
Router(config)# crypto isakmp key cisco address 192.168.5.1
```

Настройка шифрования трафика между сетями:

Настройка ACL на Router1:

```
Router1(config)# ip access-list extended MAP_VPN
Router1(config-ext-nacl)# permit ip 10.0.10.0 0.0.0.255 10.0.20.0 0.0.0.255
```

Настройка ACL на Router2:

```
Router2(config)# ip access-list extended MAP_VPN
Router2(config-ext-nacl)# permit ip 10.0.20.0 0.0.0.255 10.0.10.0 0.0.0.255
```

Настройка политики для защиты передаваемых данных (одинаковая на Router1 и Router2):

```
Router(config)# crypto ipsec transform-set MAP_set esp-aes esp-sha-hmac
```

Настройка crypto map и применение на внешнем интерфейсе.

Настройка crypto map на Router1 и применение на интерфейса f2/0:

```
Router1(config)# crypto map MAP1 10 ipsec-isakmp
Router1(config-crypto-map)# set peer 192.168.7.2
Router1(config-crypto-map)# set transform-set MAP_set
Router1(config-crypto-map)# match address MAP_VPN
```

```
Router1(config)# interface FastEthernet2/0
Router1(config-if)# crypto map MAP1
```

Настройка crypto map на Router2:

```
Router2(config)# crypto map MAP1 10 ipsec-isakmp
Router2(config-crypto-map)# set peer 192.168.5.1
Router2(config-crypto-map)# set transform-set MAP_set
Router2(config-crypto-map)# match address MAP_VPN
Router2(config)# interface FastEthernet2/0
Router2(config-if)# crypto map MAP1
```

Дополнительные команды VPN

Просмотр установленных SA первой фазы:

```
Router# show crypto isakmp sa
```

Просмотр информации crypto-map:

```
Router# show crypto map
```

Просмотр Дополнительной информации:

```
Router# show crypto session brief
Router# show crypto session
Router# show crypto session detail
Router# show crypto ruleset
```

2.10 Динамическая маршрутизация

При динамической маршрутизации происходит обмен маршрутной информацией между соседними маршрутизаторами, в ходе которого они сообщают друг другу, какие сети в данный момент доступны через них. Информация обрабатывается и помещается в таблицу маршрутизации. К наиболее распространенным внутренним протоколам маршрутизации относятся:

1. RIP (Routing Information Protocol) — протокол маршрутной информации
2. OSPF (Open Shortest Path First) — протокол выбора кратчайшего маршрута
3. EIGRP (Enhanced Interior Gateway Routing Protocol) — усовершенствованный протокол маршрутизации внутреннего шлюза
4. IGRP (Interior Gateway Routing Protocol) — протокол маршрутизации внутреннего шлюза

Большинство алгоритмов маршрутизации может быть отнесено к одной из двух категорий: дистанционно-векторные протоколы (RIPv1, RIPv2, RIPng, IGRP, EIGRP, EIGRP for IPv6) и протоколы с учетом состояния канала (OSPFv2, OSPFv3, IS-IS, IS-IS for IPv6).

По области применения протоколы разделяют на:

1. Протоколы междоменной маршрутизации (EGP): BGP
2. Протоколы внутридоменной маршрутизации (IGP): OSPF, RIP, EIGRP, IS-IS

Routing Information Protocol (RIP)

Протокол RIP является дистанционно-векторным протоколом маршрутизации. Протоколы динамической маршрутизации определяют оптимальный путь к необходимой сети на основании значения, которое называется метрикой. В качестве метрики в протоколе RIP используется количество транзитных устройств или переходов (hop count – прыжок пакета) из одной сетевой структуры в другую. Максимальное число таких переходов равно 15. А все сети, число переходов до которых превышает 15, считаются недостижимыми. Маршрутизаторы, на которых настроен протокол RIP, периодически (по умолчанию каждые 30 с) пересылают полные анонсы маршрутов, в которых содержится информация обо всех известных им сетях.

Настройка RIP (Routing Information Protocol) состоит из трех шагов:

1. Включения протокола глобальной командой:

```
Router(config)# router rip
```

2. Выбор сетей, которые протокол будет вещать:

```
Router(config-router)# network A.B.C.D
```

Пример настройки сети с использованием протокола RIP

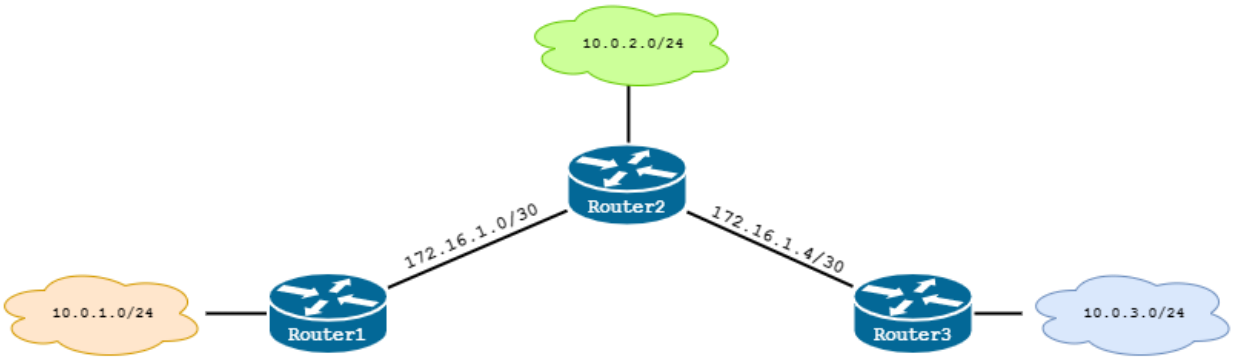

Рисунок 7 – Схема сети рассматриваемого примера

Настройка Router1:

```
Router1(config)# router rip
Router1(config-router)# network 10.0.0.0
Router1(config-router)# network 172.0.0.0
```

Настройка Router2:

```
Router2(config)# router rip
Router2(config-router)# network 10.0.0.0
Router2(config-router)# network 172.0.0.0
```

Настройка Router3:

```
Router3(config)# router rip
Router3(config-router)# network 10.0.0.0
Router3(config-router)# network 172.0.0.0
```

Open Shortest Path First (OSPF)

Протокол OSPF является протоколом маршрутизации с учетом состояния каналов. В этом классе протоколов в качестве метрики используется стоимость маршрута, которая рассчитывается на основе пропускной способности каждого канала на пути от маршрутизатора до необходимой сети. Поэтому процесс работы протокола OSPF условно можно разделить на три этапа: обнаружение соседних маршрутизаторов, обмен базами маршрутов и расчет оптимальных маршрутов.

Принцип работы протокола заключается в следующем:

1. Маршрутизаторы обмениваются маленькими HELLO-пакетами
2. Обменявшись пакетами, они устанавливают соседские отношения, добавляя каждый друг друга в свою локальную таблицу соседей
3. Маршрутизаторы собирают состояния всех своих линков (связей с соседями), включающие в себя id Маршрутизатора, id соседа, сеть и префикс между ними, тип сети,

стоимость линка (метрику) и формируют пакет, называемый LSA (Link State Advertisement).

4. Маршрутизатор рассылает LSA своим соседям, те распространяют LSA дальше.

5. Каждый маршрутизатор, получивший LSA добавляет в свою локальную табличку LSDB (Link State Database) информацию из LSA.

6. В LSDB скапливается информация, обо всех парах соединённых в сети маршрутизаторов, то есть каждая строчка таблицы — это информация вида: «Маршрутизатор A имеет соединение со своим соседом маршрутизатором B, между ними сеть такая-то с такими-то свойствами».

7. После обмена LSA, каждый маршрутизатор знает про все линки, на основании пар строится полная карта сети, включающая все маршрутизаторы и все связи между ними.

8. На основании этой карты каждый маршрутизатор индивидуально ищет кратчайшие с точки зрения метрики маршруты во все сети и добавляет их в таблицу маршрутизации.

Передача статического маршрута по умолчанию средствами OSPF:

```
Router(config-router)#default-information originate
```

Параметры, статистика протоколов маршрутизации запущенных на маршрутизаторе:

```
Router# show ip protocols
```

Информация о Router ID, таймерах и статистика:

```
Router# show ip ospf
```

Маршруты полученные по протоколу OSPF:

```
Router# show ip route ospf
```

Информация о настройках OSPF на интерфейсах:

```
Router# show ip ospf interface
```

База данных состояния каналов:

```
Router# show ip ospf database
```

Пример настройки сети с использованием протокола OSPF

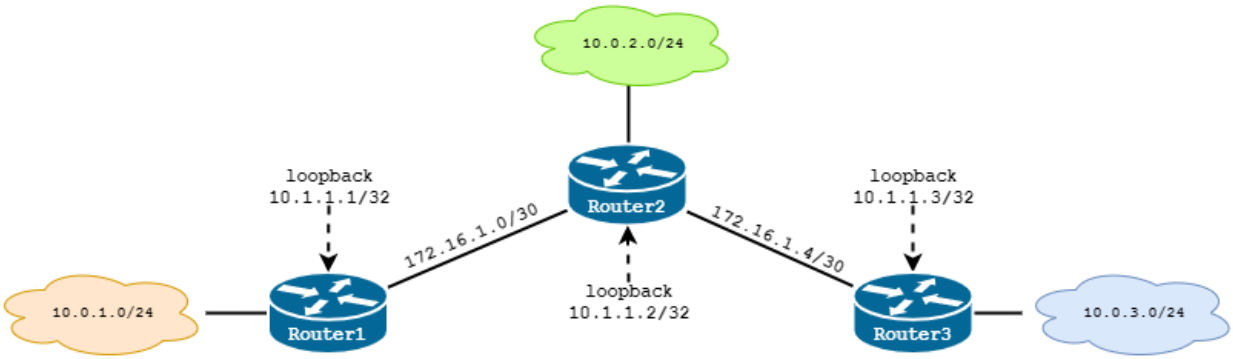

Рисунок 7 – Схема сети рассматриваемого примера

Настройка Router1:

```
Router1(config)# route ospf 1
Router1(config-router)# network 10.0.1.0 0.0.0.255 area 1
Router1(config-router)# network 172.16.1.0 0.0.0.3 area 0
```

Настройка Router2:

```
Router2(config)# router ospf 1
Router2(config-router)# network 10.0.2.0 0.0.0.255 area 2
Router2(config-router)# network 172.16.1.0 0.0.0.3 area 0
Router2(config-router)# network 172.16.1.4 0.0.0.3 area 0
```

Настройка Router3:

```
Router3(config)# router ospf 1
Router3(config-router)# network 10.0.3.0 0.0.0.255 area 3
Router3(config-router)# network 172.16.1.4 0.0.0.3 area 0
```

Список литературы

1. Служба поддержки Cisco [Электронный ресурс] – URL: https://www.cisco.com/c/ru_ru/support

2. Сборник статей по сетевым технологиям [Электронный ресурс] – URL:http://xgu.ru

3. Jesin A.Packet Tracer NetworkSimulator. Birmingham: Packt Publishing, 2014 — 134c.

4. Odom W. Официальное руководство Cisco по подготовке к сертификационным экзаменам CCNA ICND2 200-105: маршрутизация и коммутация. Москва: Вильямс, 2019 — 1452c.

5. Bonnie J. Cisco IOS in a Nutshell, 2nd Edition. Sebastopol: O'Reilly Media, 2009. — 800 c

6. Andrew S. Tanenbaum, David J. Wetherall. Компьютерные сети. СПб.: Питер, 2012. — 960 c

Лабораторная работа №1. Маршрутизация

Цель работы

Изучить принципы маршрутизации в IP-сетях, получить практические навыки настройки маршрутизаторов Cisco с применением маршрутизации интерфейсов и маршрутизации виртуальных локальных сетей (VLAN).

Задание

Настроить взаимодействие двух IP-сетей между собой и с внешней сетью средствами маршрутизатора. Настроить простейшие правила фильтрации трафика средствами ACL.

Ситуация 1. Сети изолированы друг от друга физически, т.е. построены на различных коммутаторах, не связанных друг с другом непосредственно.

Ситуация 2. Изоляция сетей обеспечивается за счет применения технологии виртуальных локальных сетей.

В качестве внешней сети (Internet) использовать маршрутизатор.

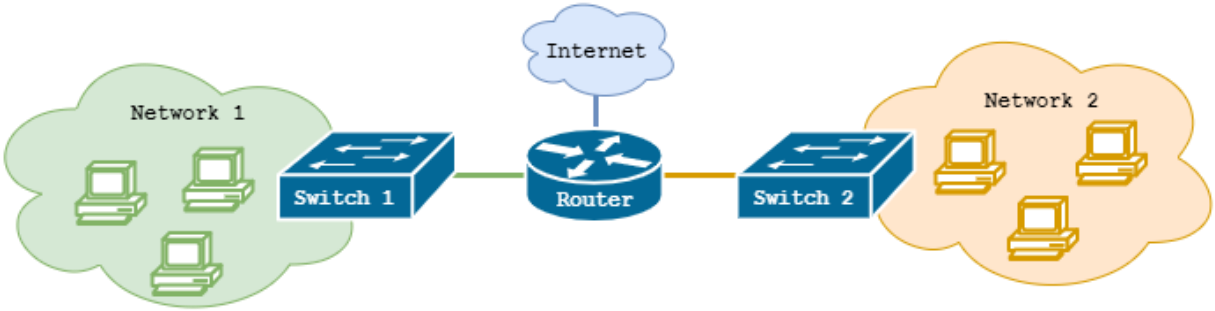

Рисунок 1 – Принципиальная схема сети для ситуации 1

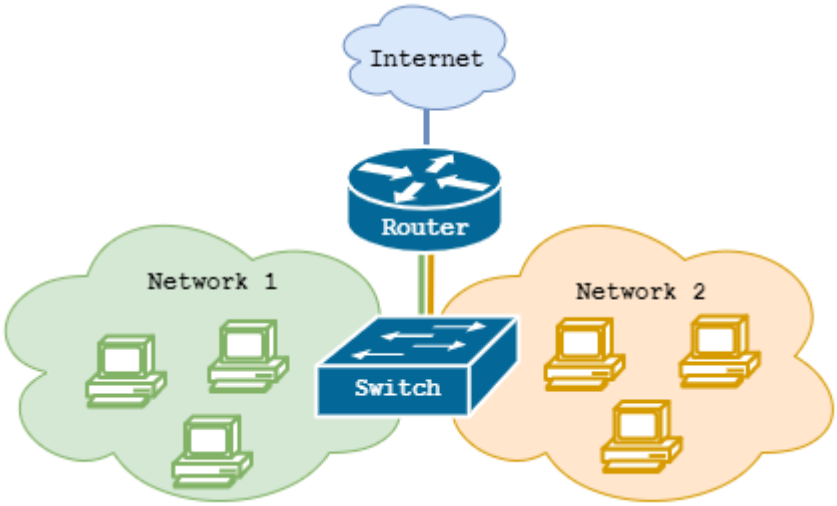

Рисунок 2 – Принципиальная схема сети для ситуации 2

Содержание отчета

1. Титульный лист

2. Цель работы, задание

4. Ситуация 1

4.1. Схемы ЛВС с указанием IP-адресов по варианту

4.2. Содержание файлов конфигурации устройств

4.3. Информация о портах устройств (команда show ip interface brief)

4.4. Информация списках ACL (команда show ip access-list)

4.5. Результат проверки достижимости сетей по варианту в окне симуляции

5. Ситуация 2

5.1. Схемы ЛВС с указанием IP-адресов по варианту

5.2. Содержание файлов конфигурации устройств

5.3. Информация о портах маршрутизатора (команда show ip interface brief)

5.4. Информация о базе VLAN коммутатора (команда show vlan)

5.5. Информация списках ACL (команда show ip access-list)

5.6. Результат проверки достижимости сетей по варианту в окне симуляции

6. Вывод

Контрольные вопросы

1. Протокол ARP.

2. Маршрутизация без масок.

3. Маршрутизация с масками.

4. Технология бесклассовой междоменной маршрутизации (CIDR).

5. Классификация протоколов маршрутизации.

6. Алгоритмы маршрутизации дистанционно-векторного типа.

7. Алгоритмы состояния связей.

Лабораторная работа №2. Настройка удаленного доступа DSL

Цель работы

Изучить технологии удаленного доступа к сети Интернет, получить практические навыки настройки DSL-модема для организации удаленного доступа по телефонному абонентскому окончанию.

Задание

Настроить подключение локальной сети офиса организации к сети Интернет по каналу DSL. Для эмуляции DSL соединения использовать устройства DSL Modem и PT-Cloud (Эмулятор сети провайдера) в соответствии с рисунком 1.

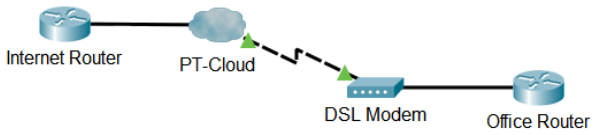

Рисунок 1 – Эмуляция DSL соединения

Ситуация 1: провайдер предоставляет маршрутизируемое подключение к сети Интернет и выделяет один «внешний» IP-адрес. Необходимо настроить доступ к сети Интернет с использованием технологии NAPT и доступ к заданному сетевому сервису локальной сети из сети Интернет.

Ситуация 2: провайдер предоставляет доступ в сеть Интернет через PPPoE-подключение.

Проверить подключение на примере взаимодействия рабочих станций и сетевого сервиса по варианту.

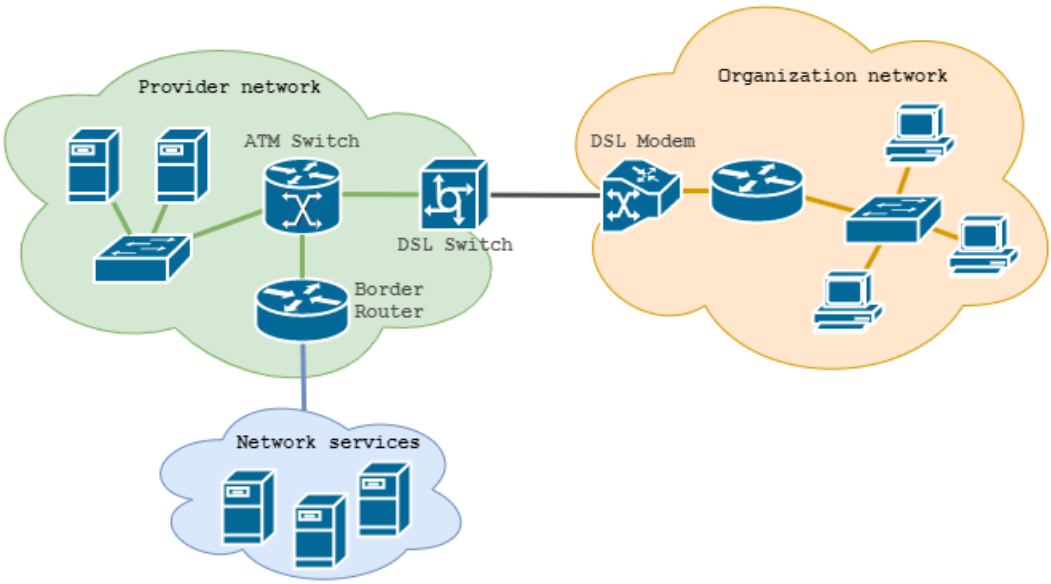

Рисунок 2 – Принципиальная схема сети

Содержание отчета

1. Титульный лист

2. Цель работы, задание

3. Схемы ЛВС с указанием IP-адресов по варианту

4. Ситуация 1

4.1. Содержание файлов конфигурации устройств

4.2. Информация о портах маршрутизаторов (команда show ip interface brief).

4.3. Информация о NAT трансферах (команда show ip nat translations)

4.4. Результат проверки работоспособности подключения в окне симуляции

5. Ситуация 2

5.1. Содержание файлов конфигурации устройств

5.2. Информация о портах маршрутизаторов (команда show ip interface brief).

5.3. Информация о сеансах PPPoE (команда show pppoe session)

5.4. Сводная информация о состоянии протокола (команда show pppoe interfaces)

5.5. Статистика пакетов полученных и отправленных сеансами (команда show pppoe statistics)

5.6. Результат проверки работоспособности подключения в окне симуляции

6. Вывод

Контрольные вопросы

1. Технология виртуальных каналов

2. Классификация технологий удаленного доступа

3. Классификация абонентов по требованиям к средствам удаленного доступа

4. Протокол PPP (RFC1661)

5. Протокол PPPoA (RFC2364)

6. Инкапсуляция по протоколу RFC1481

7. Протокол PPPoE (RFC2516)

Лабораторная работа №3. Настройка динамической маршрутизации

Цель работы

Изучить протоколы динамической маршрутизации в IP-сетях, получить практические навыки настройки маршрутизаторов с применением динамической маршрутизации RIP и OSPF.

Задание

Настроить взаимодействие IP-сетей с использованием маршрутизаторов Cisco. Настроить обмен маршрутной информацией по протоколам RIP и OSPF. Проверить работоспособность сети.

Проверить настройку маршрутизации на примере взаимодействия рабочих станций, принадлежащих различным сетям, и взаимодействие рабочих станций с внешней сетью.

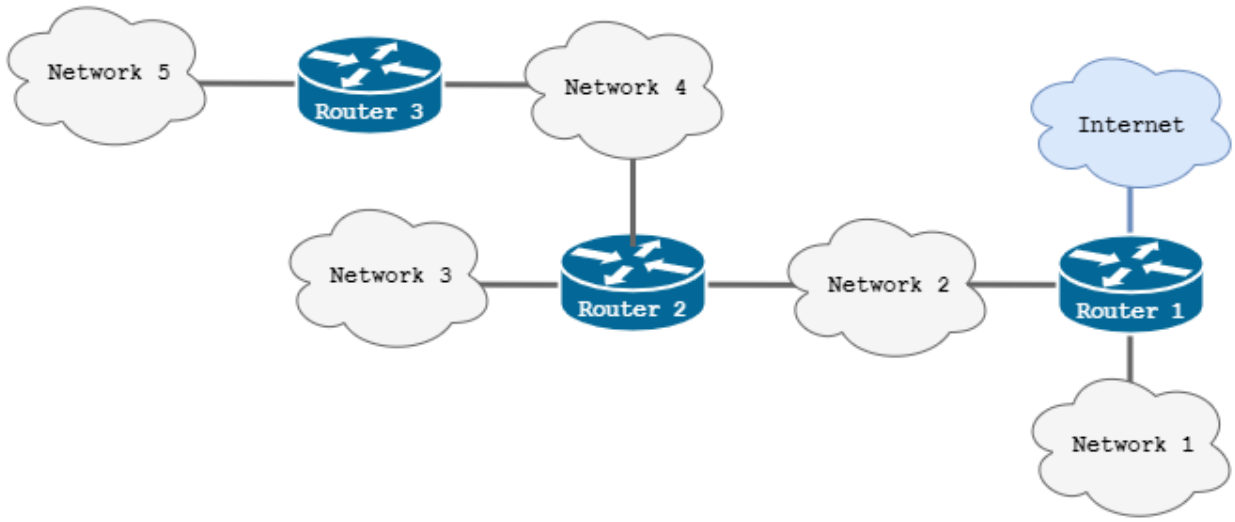

Рисунок 1 – Принципиальная схема сети

Содержание отчета

1. Титульный лист.
2. Цель работы, задание.
3. Схемы ЛВС с указанием IP-адресов по варианту
4. Протокол RIP
4.1. Схемы ЛВС с указанием выделенных областей
4.2. Содержание файлов конфигурации устройств
4.3. Информация о портах маршрутизаторов (команда show ip interface brief)
4.4. Таблица маршрутизации (команда show ip route)

4.5. Параметры, статистика протоколов маршрутизации запущенных на маршрутизаторах (команда show ip protocols)

4.6. База данных RIP (команда show ip rip database)

4.7. Результаты проверки динамической маршрутизации по протоколу RIP в окне симуляции

5. Протокол OSPF

5.1. Схемы ЛВС с указанием выделенных областей

5.2. Содержание файлов конфигурации устройств

5.3. Информация о портах маршрутизаторов (команда show ip interface brief)

5.4. Таблица маршрутизации (команда show ip route)

5.5. Параметры, статистика протоколов маршрутизации запущенных на маршрутизаторах (команда show ip protocols)

5.6. Информация о настройках OSPF на интерфейсах (команда show ip ospf interface)

5.7. База данных состояния каналов (команда show ip ospf database)

6. Вывод

Контрольные вопросы

1. Построение таблицы маршрутизации с помощью протокола RIP.

2. Адаптация маршрутизаторов RIP к изменениям состояния сети.

3. Методы борьбы с ложными маршрутами в протоколе RIP.

4. Формат RIP пакета.

5. Алгоритм работы OSPF.

6. Нахождение оптимальных маршрутов и генерация таблицы маршрутизации OSPF.

7. Изменение конфигурации сети OSPF.

8. Состояния соседей OSPF.

9. Типы областей OSPF.

10. Классификация маршрутизаторов OSPF.

11. Формат OSPF пакета.

Лабораторная работа №4. Виртуальные частные сети VPN

Цель работы

Изучить технологии виртуальных частных сетей для организации защищенных каналов связи через сети общего пользования (Интернет), получить практические навыки настройки виртуальных частных сетей на маршрутизаторах Cisco.

Задание

Настроить безопасное взаимодействие двух IP-сетей между собой через сеть общего пользования (Интернет), средствами маршрутизаторов Cisco. Аутентификацию настроить с использованием сертификатов безопасности. Проверить работу защищенного соединения на примере взаимодействия рабочих станций, принадлежащих различным сетям.

Рисунок 1 – Принципиальная схема сети

Содержание отчета

1. Титульный лист
2. Цель работы, задание
3. Схемы ЛВС с указанием IP-адресов по варианту
4. Содержание файлов конфигурации устройств
5. Информация о портах маршрутизатора (команда show ip interface brief)
6. Информация об установленных SA (команда show crypto ipsec sa)
7. Структура и содержание пакетов при передачи между сетями
8. Результат проверки работоспособности в окне симуляции
9. Вывод

Контрольные вопросы

1. Общие принципы туннелирования трафика.
2. TUN/TAP туннели.
3. Аутентификация в OpenVPN.
4. Библиотека OpenSSL. Средства защиты трафика.
5. Архитектура протоколов SSL/TLS.
6. Аутентификация в SSL/TLS.
7. Процедура установления соединения в SSL/TLS.

Приложение 1. Варианты заданий для лабораторной работы №1

Таблица 1. Варианты адресов сетей

№	Сеть 1	Сеть 2	№	Сеть 1	Сеть 2
1	10.100.1.0/24	10.100.200.0/24	21	10.100.21.0/24	10.100.180.0/24
2	10.100.2.0/24	10.100.199.0/24	22	10.100.22.0/24	10.100.179.0/24
3	10.100.3.0/24	10.100.198.0/24	23	10.100.23.0/24	10.100.178.0/24
4	10.100.4.0/24	10.100.197.0/24	24	10.100.24.0/24	10.100.177.0/24
5	10.100.5.0/24	10.100.196.0/24	25	10.100.25.0/24	10.100.176.0/24
6	10.100.6.0/24	10.100.195.0/24	26	10.100.26.0/24	10.100.175.0/24
7	10.100.7.0/24	10.100.194.0/24	27	10.100.27.0/24	10.100.174.0/24
8	10.100.8.0/24	10.100.193.0/24	28	10.100.28.0/24	10.100.173.0/24
9	10.100.9.0/24	10.100.192.0/24	29	10.100.29.0/24	10.100.172.0/24
10	10.100.10.0/24	10.100.191.0/24	30	10.100.30.0/24	10.100.171.0/24
11	10.100.11.0/24	10.100.190.0/24	31	10.100.31.0/24	10.100.170.0/24
12	10.100.12.0/24	10.100.189.0/24	32	10.100.32.0/24	10.100.169.0/24
13	10.100.13.0/24	10.100.188.0/24	33	10.100.33.0/24	10.100.168.0/24
14	10.100.14.0/24	10.100.187.0/24	34	10.100.34.0/24	10.100.167.0/24
15	10.100.15.0/24	10.100.186.0/24	35	10.100.35.0/24	10.100.166.0/24
16	10.100.16.0/24	10.100.185.0/24	36	10.100.36.0/24	10.100.165.0/24
17	10.100.17.0/24	10.100.184.0/24	37	10.100.37.0/24	10.100.164.0/24
18	10.100.18.0/24	10.100.183.0/24	38	10.100.38.0/24	10.100.163.0/24
19	10.100.19.0/24	10.100.182.0/24	39	10.100.39.0/24	10.100.162.0/24
20	10.100.20.0/24	10.100.181.0/24	40	10.100.40.0/24	10.100.161.0/24

Таблица 2. Варианты правил фильтрации

№	Откуда	Куда	Разрешение
1	Сеть 1	Сеть 2	Полный доступ
	Сеть 2	Сеть 1	RDP
	Сеть 1	Внешняя сеть	HTTP, POP3, SMTP
	Сеть 2	Внешняя сеть	HTTP, HTTPS
2	Сеть 1	Сеть 2	Полный доступ
	Сеть 2	Сеть 1	SNMP, Telnet, TFTP
	Сеть 1	Внешняя сеть	HTTP, IMAP, NNTP
	Сеть 2	Внешняя сеть	RDP
3	Сеть 1	Сеть 2	SNTP, Telnet, POP3, SMTP
	Сеть 2	Сеть 1	Полный доступ
	Сеть 1	Внешняя сеть	IMAP, POP3, SMTP
	Сеть 2	Внешняя сеть	HTTP, HTTPS, FTP
4	Сеть 1	Сеть 2	SMTP, HTTP
	Сеть 2	Сеть 1	RDP, Telnet
	Сеть 1	Внешняя сеть	HTTPS, POP3, IMAP
	Сеть 2	Внешняя сеть	HTTP
5	Сеть 1	Сеть 2	RDP, POP3, SMTP, IMAP
	Сеть 2	Сеть 1	Полный доступ
	Сеть 1	Внешняя сеть	HTTP, HTTPS
	Сеть 2	Внешняя сеть	POP3

Приложение 2. Варианты заданий для лабораторной работы №2

Таблица 1. Варианты адресов сетей и сетевых служб

№	Адрес сети	Сетевая служба	№	Адрес сети	Сетевая служба
1	10.100.1.0/24	RDP	21	10.100.21.0/24	RDP
2	10.100.2.0/24	HTTP	22	10.100.22.0/24	HTTP
3	10.100.3.0/24	SNMP	23	10.100.23.0/24	SNMP
4	10.100.4.0/24	POP3	24	10.100.24.0/24	POP3
5	10.100.5.0/24	SMTP	25	10.100.25.0/24	SMTP
6	10.100.6.0/24	RDP	26	10.100.26.0/24	RDP
7	10.100.7.0/24	HTTP	27	10.100.27.0/24	HTTP
8	10.100.8.0/24	SNMP	28	10.100.28.0/24	SNMP
9	10.100.9.0/24	POP3	29	10.100.29.0/24	POP3
10	10.100.10.0/24	SMTP	30	10.100.30.0/24	SMTP
11	10.100.11.0/24	RDP	31	10.100.31.0/24	RDP
12	10.100.12.0/24	HTTP	32	10.100.32.0/24	HTTP
13	10.100.13.0/24	SNMP	33	10.100.33.0/24	SNMP
14	10.100.14.0/24	POP3	34	10.100.34.0/24	POP3
15	10.100.15.0/24	SMTP	35	10.100.35.0/24	SMTP
16	10.100.16.0/24	RDP	36	10.100.36.0/24	RDP
17	10.100.17.0/24	HTTP	37	10.100.37.0/24	HTTP
18	10.100.18.0/24	SNMP	38	10.100.38.0/24	SNMP
19	10.100.19.0/24	POP3	39	10.100.39.0/24	POP3
20	10.100.20.0/24	SMTP	40	10.100.40.0/24	SMTP

Приложение 3. Варианты заданий для лабораторной работы №3

Для каждой сети указано количество обслуживаемых компьютеров.

Таблица 1. Варианты адресов сетей

№	Диапазон IP-адресов	Network 1	Network 2	Network 3	Network 4	Network 5
1	10.10.0.0/16	73	15	87	59	30
2	10.11.0.0/16	42	39	95	48	67
3	10.12.0.0/16	53	13	31	62	17
4	10.13.0.0/16	73	93	26	59	62
5	10.14.0.0/16	70	26	95	51	16
6	10.15.0.0/16	28	65	48	88	57
7	10.16.0.0/16	59	24	35	53	45
8	10.17.0.0/16	68	67	54	53	24
9	10.18.0.0/16	23	79	59	57	62
10	10.19.0.0/16	62	87	84	67	49
11	10.20.0.0/16	46	80	57	91	58
12	10.21.0.0/16	13	25	94	45	95
13	10.22.0.0/16	66	31	96	72	28
14	10.23.0.0/16	78	80	84	23	79
15	10.24.0.0/16	92	47	55	30	37
16	10.25.0.0/16	68	31	20	100	92
17	10.26.0.0/16	73	48	86	77	20
18	10.27.0.0/16	29	48	52	80	47
19	10.28.0.0/16	84	40	31	18	84
20	10.29.0.0/16	53	45	14	33	61

Приложение 4. Варианты заданий для лабораторной работы №4

Таблица 1. Варианты адресов сетей

№	IP-сеть 1	IP-сеть 2	№	IP-сеть 1	IP-сеть 2
1	10.100.1.0/24	10.100.200.0/24	21	10.100.21.0/24	10.100.180.0/24
2	10.100.2.0/24	10.100.199.0/24	22	10.100.22.0/24	10.100.179.0/24
3	10.100.3.0/24	10.100.198.0/24	23	10.100.23.0/24	10.100.178.0/24
4	10.100.4.0/24	10.100.197.0/24	24	10.100.24.0/24	10.100.177.0/24
5	10.100.5.0/24	10.100.196.0/24	25	10.100.25.0/24	10.100.176.0/24
6	10.100.6.0/24	10.100.195.0/24	26	10.100.26.0/24	10.100.175.0/24
7	10.100.7.0/24	10.100.194.0/24	27	10.100.27.0/24	10.100.174.0/24
8	10.100.8.0/24	10.100.193.0/24	28	10.100.28.0/24	10.100.173.0/24
9	10.100.9.0/24	10.100.192.0/24	29	10.100.29.0/24	10.100.172.0/24
10	10.100.10.0/24	10.100.191.0/24	30	10.100.30.0/24	10.100.171.0/24
11	10.100.11.0/24	10.100.190.0/24	31	10.100.31.0/24	10.100.170.0/24
12	10.100.12.0/24	10.100.189.0/24	32	10.100.32.0/24	10.100.169.0/24
13	10.100.13.0/24	10.100.188.0/24	33	10.100.33.0/24	10.100.168.0/24
14	10.100.14.0/24	10.100.187.0/24	34	10.100.34.0/24	10.100.167.0/24
15	10.100.15.0/24	10.100.186.0/24	35	10.100.35.0/24	10.100.166.0/24
16	10.100.16.0/24	10.100.185.0/24	36	10.100.36.0/24	10.100.165.0/24
17	10.100.17.0/24	10.100.184.0/24	37	10.100.37.0/24	10.100.164.0/24
18	10.100.18.0/24	10.100.183.0/24	38	10.100.38.0/24	10.100.163.0/24
19	10.100.19.0/24	10.100.182.0/24	39	10.100.39.0/24	10.100.162.0/24
20	10.100.20.0/24	10.100.181.0/24	40	10.100.40.0/24	10.100.161.0/24

Таблица 2. Варианты адресов сетей VPN-туннеля

№	Подсеть VPN-туннеля	№	Подсеть VPN-туннеля
1	10.200.21.0/24	21	10.200.1.0/24
2	10.200.22.0/24	22	10.200.2.0/24
3	10.200.23.0/24	23	10.200.3.0/24
4	10.200.24.0/24	24	10.200.4.0/24
5	10.200.25.0/24	25	10.200.5.0/24
6	10.200.26.0/24	26	10.200.6.0/24
7	10.200.27.0/24	27	10.200.7.0/24
8	10.200.28.0/24	28	10.200.8.0/24
9	10.200.29.0/24	29	10.200.9.0/24
10	10.200.30.0/24	30	10.200.10.0/24
11	10.200.31.0/24	31	10.200.11.0/24
12	10.200.32.0/24	32	10.200.12.0/24
13	10.200.33.0/24	33	10.200.13.0/24
14	10.200.34.0/24	34	10.200.14.0/24
15	10.200.35.0/24	35	10.200.15.0/24
16	10.200.36.0/24	36	10.200.16.0/24
17	10.200.37.0/24	37	10.200.17.0/24
18	10.200.38.0/24	38	10.200.18.0/24
19	10.200.39.0/24	39	10.200.19.0/24
20	10.200.40.0/24	40	10.200.20.0/24