Практическая работа № 7

Настройка AccessControlList.

ACL (AccessControlList) — это набор текстовых выражений, которые что-то разрешают, либо что-то запрещают. Обычно ACL разрешает или запрещает IP-пакеты, но помимо всего прочего он может заглядывать внутрь IP-пакета, просматривать тип пакета, TCP и UDP порты. Также ACL существует для различных сетевых протоколов (IP, IPX, AppleTalk и так далее). В основном применение списков доступа рассматривают с точки зрения пакетной фильтрации, то есть пакетная фильтрация необходима в тех ситуациях, когда у вас стоит оборудование на границе Интернет и вашей частной сети и нужно отфильтровать ненужный трафик.
Вы размещаете ACL на входящем направлении и блокируете избыточные виды трафика.

Функционал ACL состоит в классификации трафика, нужно его проверить сначала, а потом что-то с ним сделать в зависимости от того, куда ACL применяется. ACL применяется везде, например:

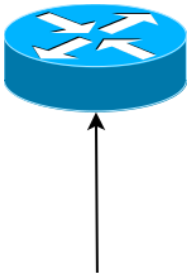
- На интерфейсе: *пакетная фильтрация*
- На линии Telnet: *ограничения доступа к маршрутизатору*
- VPN: *какой трафик нужно шифровать*
- QoS: *какой трафик обрабатывать приоритетнее*
- NAT: *какие адреса транслировать*

Для применения ACL для всех этих компонентов нужно понять как они работают. И мы в первую очередь будем касаться пакетной фильтрации. Применительно к пакетной фильтрации, ACL размещаются на интерфейсах, сами они создаются независимо, а уже потом они прикручиваются к интерфейсу. Как только вы его прикрутили к интерфейсу маршрутизатор начинает просматривать трафик. Маршрутизатор рассматривает трафик как входящий и исходящий. Тот трафик, который входит в маршрутизатор называется входящим, тот который из него выходит — исходящий. Соответственно ACL размещаются на входящем или на исходящем направлении.

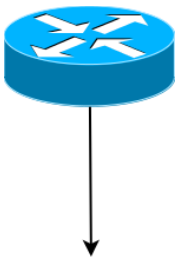
Входящий и исходящий трафик

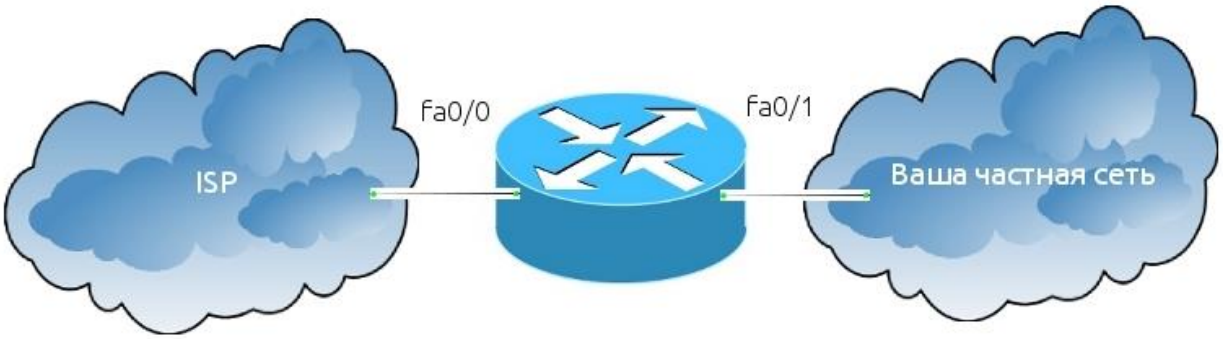
Для почину давайте-ка разберёмся с одной вещью. Что понимать под входящим и исходящим трафиком? Это нам в будущем понадобится.

Входящий трафик — этот тот, который приходит на интерфейс извне.



Исходящий — тот, который отправляется с интерфейса вовне.





Из вашей частной сети приходит пакет на интерфейс маршрутизатора fa0/1, маршрутизатор проверяет есть ли ACL на интерфейсе или нет, если он есть, то дальше обработка ведется по правилам списка доступа **строго в том порядке, в котором записаны выражения**, если список доступа разрешает проходить пакету, то в данном случае маршрутизатор отправляет пакет провайдеру через интерфейс fa0/0, если список доступа не разрешает проходить пакету, пакет уничтожается. Если списка доступа нет — пакет пролетает без всяких ограничений. Перед тем как отправить пакет провайдеру, маршрутизатор ещё проверяет интерфейс fa0/0 на наличие исходящего ACL. Дело в том, что ACL может быть прикреплен на интерфейсе как входящий или исходящий. К примеру у нас есть ACL с правилом запретить всем узлам в Интернете посылать в нашу сеть пакеты. Так на какой интерфейс прикрепить данную ACL? Если мы прикрепим ACL на интерфейс fa0/1 как исходящий, это будет не совсем верно, хотя и ACL работать будет. На маршрутизатор приходит эхо-запрос для какого-то узла в частной сети, он проверяет на интерфейсе fa0/0 есть ли ACL, его нет, дальше проверяет интерфейс fa0/1, на данном интерфейсе есть ACL, он настроен как исходящий, всё верно пакет не проникает в сеть, а уничтожается маршрутизатором. Но если мы прикрепим ACL за интерфейсом fa0/0 как входящий, то пакет будет уничтожатся сразу как пришел на маршрутизатор. Последнее решение является правильным, так как маршрутизатор меньше

нагружает свои вычислительные ресурсы. **Расширенные ACL нужно размещать как можно ближе к источнику, стандартные же как можно ближе к получателю**. Это нужно для того, чтобы не гонять пакеты по всей сети зря.

Сам же ACL представляет собой набор текстовых выражений, в которых написано **permit** (разрешить) либо **deny** (запретить), и обработка ведется строго в том порядке в котором заданы выражения. Соответственно когда пакет попадает на интерфейс он проверяется на первое условие, если первое условие совпадает с пакетом, дальнейшая его обработка прекращается. Пакет либо перейдет дальше, либо уничтожится.
Ещё раз, **если пакет совпал с условием, дальше он не обрабатывается**. Если первое условие не совпало, идет обработка второго условия, если оно совпало, обработка прекращается, если нет, идет обработка третьего условия и так дальше пока не проверятся все условия, **если никакое из условий не совпадает, пакет просто уничтожается**. Помните, в каждом конце списка стоит неявный denyany (запретить весь трафик). Будьте очень внимательны с этими правилами, которые я выделил, потому что очень часто происходят ошибки при конфигурации.

ACL разделяются на два типа:

- Стандартные (Standard): *могут проверять только адреса источников*
- 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. Мы в свою очередь будем работать с IP. Обозначаются они следующим образом, нумерованные списки доступа:

- Стандартные: *от 1 до 99*
- Расширенные: *от 100 до 199*

Символьные ACL разделяются тоже на стандартные и расширенные. Расширенные напомню могут проверять гораздо больше, нежели стандартные, но и работают они медленнее, так как придется заглядывать внутрь пакета, в отличии от стандартных где мы смотрим только поле SourceAddress (Адрес отправителя). При создании ACL каждая запись списка доступа обозначается порядковым номером, по умолчанию в рамках десяти

(10, 20, 30 и т.д). Благодаря чему, можно удалить конкретную запись и на её место вставить другую, но эта возможность появилась в Cisco IOS 12.3, до 12.3 приходилось ACL удалять, а потом создать заново полностью. **Нельзя разместить более 1 списка доступа на интерфейс, на протокол, на направление**. Объясняю: если у нас есть маршрутизатор и у него есть интерфейс, мы можем на входящее направление для IP-протокола разместить только один список доступа, например под номером 10. Ещё одно правило, касающееся самих маршрутизаторов, **ACL не действует на трафик, сгенерированный самим маршрутизатором**.
Для фильтрации адресов в ACL используется WildCard-маска. Это обратная маска.

Добавление стандартного ACL-листа выполняется командой:
left(config)#ip access-list standard ACL_NAME
ACL_NAME в данном случае – имя нашего списка доступа. В имени допустима
латиница и цифры. Регистр символов имеет значение! После запуска этой команды можно
добавлять правила, составляющие этот список. Формат команды таков:
left(config-ext-nacl)#<номер_правила><действие><адрес>

<номер_правила> определяет порядок просмотра правила в рамках данного листа.
<действие> может принимать значение permit- разрешить передачу пакета, deny-
запретить передачу пакета.
<адрес> представляет собой следующие варианты:
– ключевое слово any, означающее любой адрес;
– IP-адрес сети и маску этой сети (причем маск должна даваться в инвертированном виде!);
– ключевое слово host после которого указывается адрес единственного узла
Удаление ошибочного правила с номером <номер_правила> выполняется в следующем порядке:
– входим в необходимый список ACL
– выполняем команду
left(config-ext-nacl)# no <номер_правила>
аналогично может быть удален и весь ACL-лист целиком:
right(config)#no ip access-list standard < ACL_NAME >
Расширенный ACL-лист позволяет (как уже говорилось) помимо адреса отправителя пакета указывать адрес получателя пакета, используемый протокол и ряд
других параметров.
Добавление расширенного ACL выполняется командой следующего формата:
left(config)#ip access-list extended myACLname

Формат команды добавления нового элемента в расширенный ACL выглядит следующим образом:

left(config-ext-nacl)#<номер правила><действие><протокол><критерии>

Поле <протокол> может принимать значения:

– ahp Authentication Header Protocol
– eigrp Cisco's EIGRP routing protocol
– esp Encapsulation Security Payload
– gre Cisco's GRE tunneling
– icmp Internet Control Message Protocol
– ip Any Internet Protocol
– ospf OSPF routing protocol
– tcp Transmission Control Protocol
– udpUserDatagramProtocol

Поле <критерии> зависит от конкретного протокола и может содержать адреса,
маски, номера портов протоколов транспортного уровня.

После того, как добавили все правила во все ACL-листы эти правила необходимо
применить на требуемых интерфейсах. Формат команды имеет вид :

right(config-if)#ip access-group <ACL_NAME><направление>

Параметр <направление> может принимать значения in или out

Команда ipaccess-group находится в ветви interface меню глобальной конфигурации. Таким образом, перехода в данную ветвь в нашем примере необходимо
(после составления всех правил) выполнить (для входящего трафика):

right# config t
right(config)# inteface GigabitEthernet0/0/0
right(config-if)# ip-access-group myACLname in

Прежде чем добавлять новый элемент, просмотрим имеющиеся списки доступа:

left# show access-list

Задание.

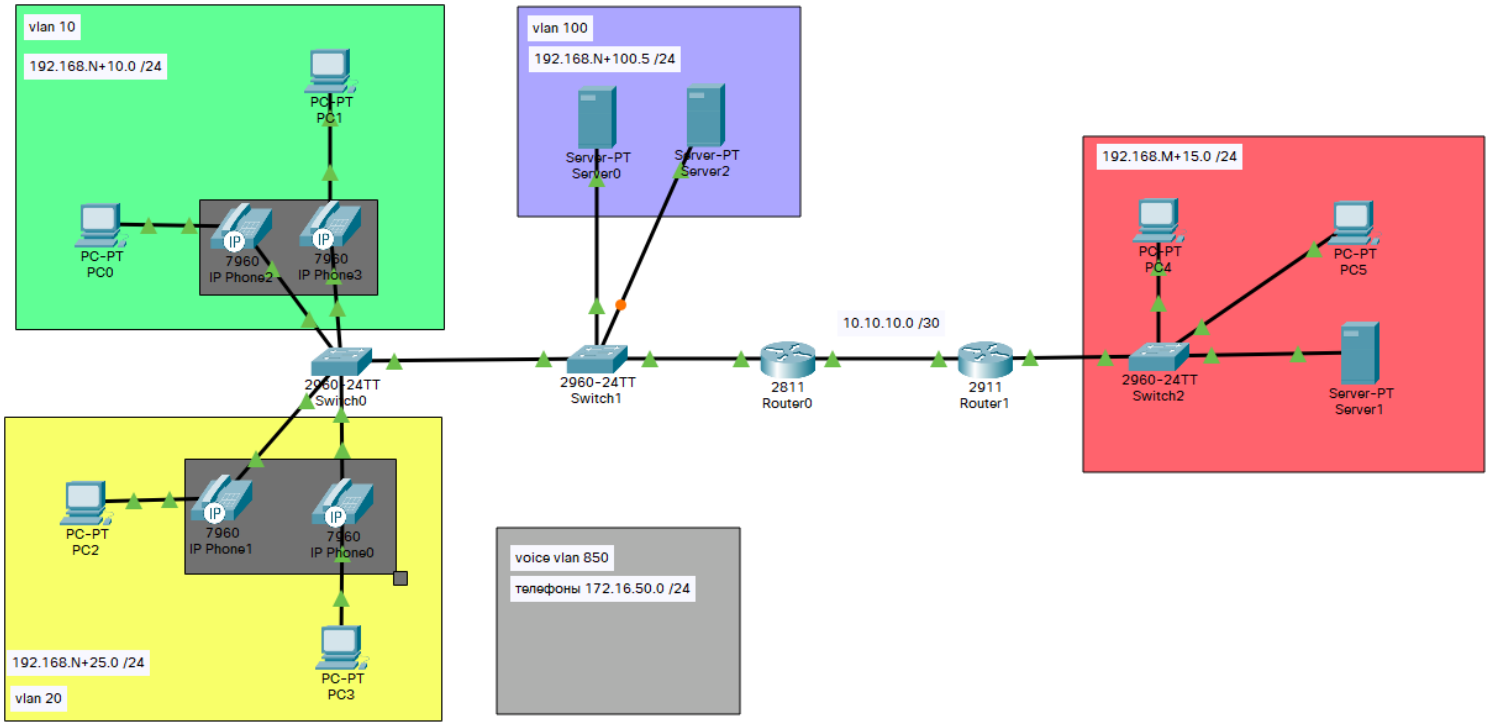


Рис 1.

Шаг 1.

Создать в CPTтопологию как на Рис.1, порты использовать любые, модель оборудование указаны на рисунке, адреса и сети подписаны.

Шаг 2.

1.Назначить адреса Компьютерам и Серверам, (**Nпоследняя** цифра студ. билета, **M предпоследняя** цифра).

2. НастроитьSwitch(vlan указаны на Рис. 1,не забываем про **description**).

Voicevlan 850

3. **!!!На switch0 иswitch1добавить все vlanы, которые они используются. (switch1 10,20,100 switch0 10,20,100)**

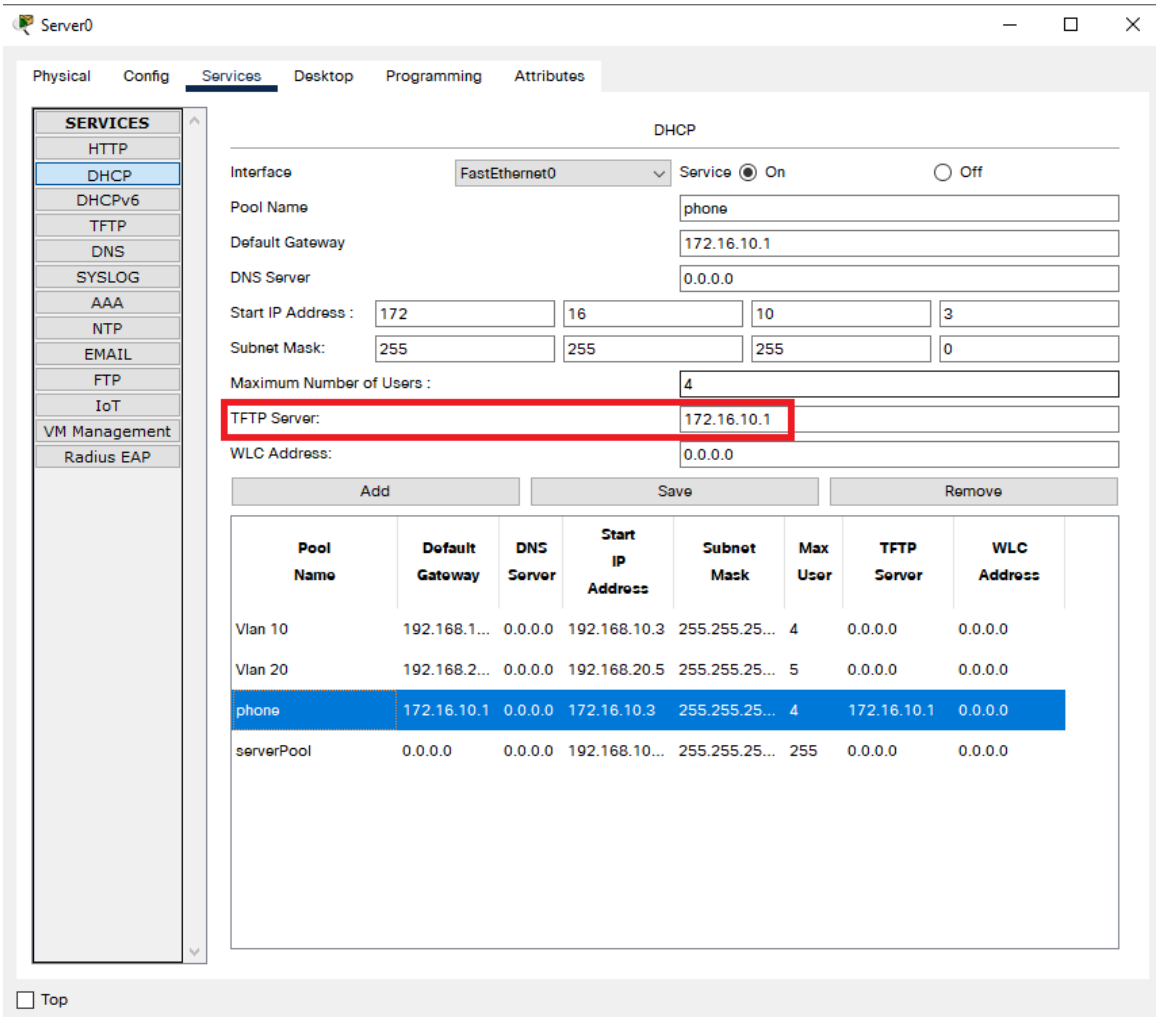
Шаг 3.

1. НастроитьRouter0 и 1.(не забываем использовать Router0 2811, Не забываем **hostname**и **description.**)

Шаг 4.

1.Настроить на Server0 DHCP пулы для vlan 10, 20 и телефонии.

2. Укажите прошивку TFTP серверадля телефонов.



3.Настроить телефонию на ROUTER2811. Использовать номер телефонов **77NM1 77NM2 77NM3 77NM4**

Включить питание на телефоне.

Шаг 5.

Выполнить эхо-тестирования(ping), заполнить ipадреса в таблице 1.

Таблица 1. Результаты эхо-тестирования.

Тестирующий Узел (IP)	Тестируемые узлы (IP-адреса)				
	PC0	Server0	Gi0/1.10	Gi0/1.20	Gi0/1.100
PC0					
PC1					
PC2					
PC3					

Шаг 6.

1.Настраиваем маршрутизацию на R1 и R2с использованием OSPF.

2. Провести это-тестирование.

Тестирующий Узел (IP)	Тестируемые узлы (IP-адреса)	
	Server1	PC5
PC0		
PC2		
Server0		

3.Проверить работу телефонов.

Шаг 7.

Настройка access-list

1.Настроить доступ через telnet и добавить пароль на enable.
Примердля telnet

R0(config)#user admin privilege 15 password cisco
R0(config)#line vty 0 15
R0(config-line)#login local
2.Настроить стандартный access-list для доступа к Router0только для компьютера PC2 admin

Пример

R0(config)# ip access-list standard Telnet
R0(config-std-nacl) # permit host _____
R0(config-std-nacl) # deny any
R0(config)#line vty 0 15
R0(config-line)# access-class<ACL_NAME><направление>

3. Настроить расширенный access-list для доступа на (**SERVER1**-web-server) PC2 и PC3. (PC2 имеет доступ на web страницу, а PC3 admin полный доступ(посмотри ip адрес PC3 или поменяйте на статический 192.168.20.20)
R0(config)#ip access-list extended From-Web
R0(config-ext-nacl)#permit tcp 192.168.20.0 0.0.0.255 host 192.168.M.2eq www
R0(config-ext-nacl)#permit ip host 192.168.20.20 host 192.168.M.2
R0(config-ext-nacl)#deny ip 192.168.20.0 0.0.0.255 host 192.168.M.2
R0(config-ext-nacl)#permit ip any any

4.Настроить access-list для доступа к Server2 только сети (Бухгалтерия 192.168.N+10.0)