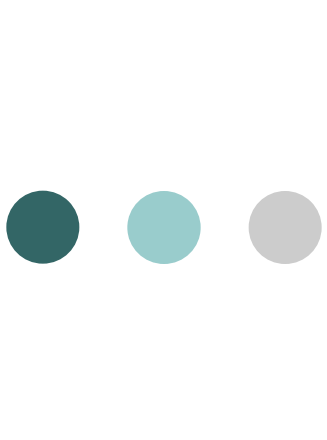


# Списки доступа

Каждое, правило в стандартном списке доступа сдержит три важных элемента:

- **число, идентифицирующее список при обращении к нему в других частях конфигурации маршрутизатора;**
- **инструкцию deny (запретить) или permit (разрешить);**
- **идентификатор пакета (например, адрес).**

Фильтрация стандартных списков доступа основана на адресах в исходной сети.

Типичный стандартный список доступа выглядит так:

**access-list 1 deny 10.10.1.0 0.0.0.255**

**access-list 1 deny 10.10.2.0 0.0.0.255**

**access-list 1 permit any**

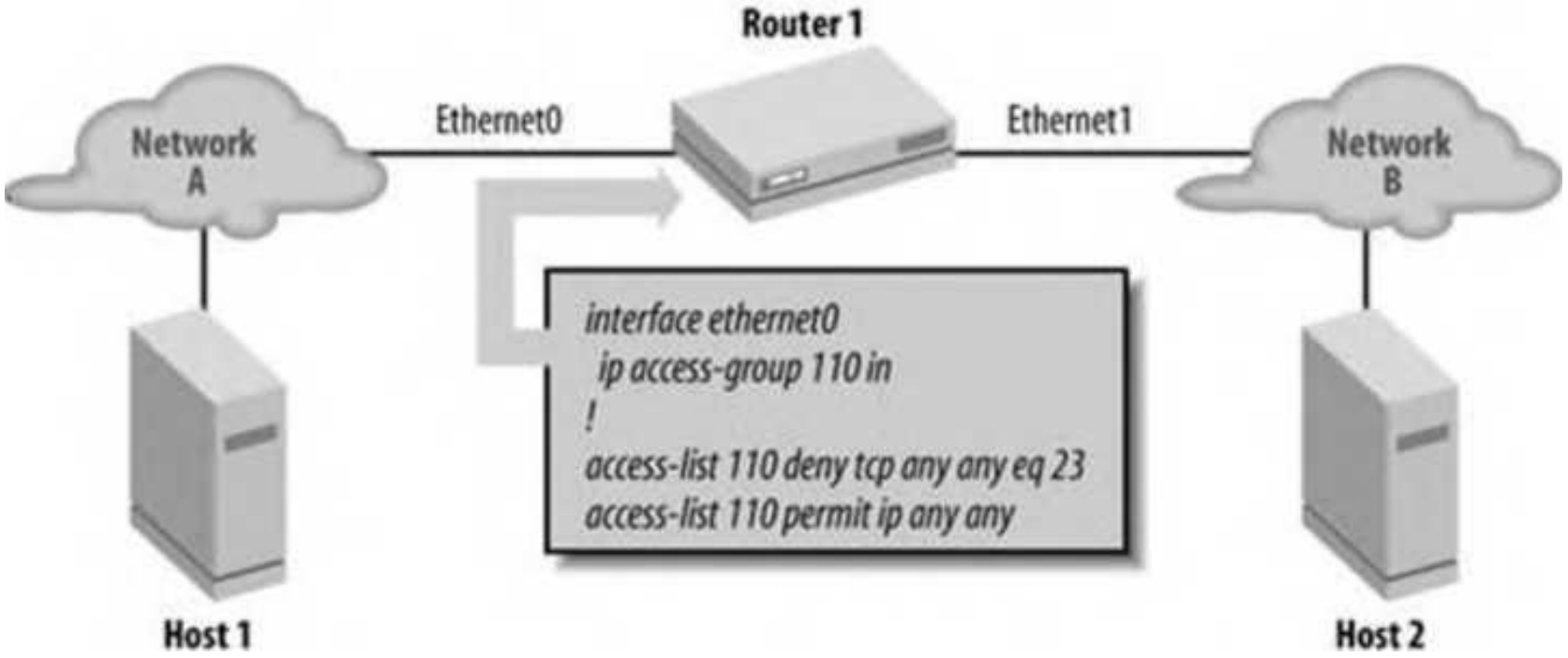
Создав список, можно применить его к пакетам, проходящим в том или ином направлении через определенный интерфейс.

Команда **ip access-group:**

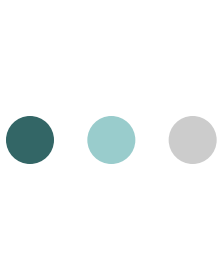
**interface ethernet0**

**ip access-group 1 in**

**Расширенные списки** доступа позволяют создавать намного более сложные фильтры, использующие исходные и целевые адреса в дополнение к информации протоколов более высокого уровня.




**Расширенный список доступа для блокировки протокола telnet**



```
! Запретить трафик на порте 80 (www-трафик)
ip access-list 111 deny tcp any any eq 80
ip access-list 111 permit ip any any
interface ethernet0
! Применить список доступа 111 к исходящему трафику
ip access-group 111 out
```

## Соответствие пакетов пунктам списка

**access-list** номер действие источник

Параметры команды:

- ***номер***

  Число от 1 до 99, идентифицирующее список.

- ***действие***

  Ключевое слово permit или deny, в зависимости от того, хотите вы разрешить или блокировать пакеты.

- ***источник***

  Адрес источника пакетов.

# Задание адресов в списках доступа

| Синтаксис | Пример | Объяснение |
|---|---|---|
| адрес маска | 192.168.2.0 0.0.0.255 | Описывает блок IP-адресов. Маска используется в качестве шаблона: единица (1) указывает, что соответствующий бит в адресе может быть любым. Нуль (0) в маске означает, что соответствующий бит должен быть точно таким же, как в указанном адресе. В этом примере указаны адреса от 192.168.2.0 до 192.168.2.255. Таким образом, маска говорит, что при сравнении адресов мы игнорируем последний байт адреса |
| host адрес | host 192.168.2.1 | Адрес должен точно совпадать с указанным |
| any | any | Любой IP-адрес |

## Пары адрес-маска (шаблоны)

Шаблонная маска выглядит как маска подсети, но в действительности ею не является, а представляет собой дополнение соответствующей маски подсети.

Например, чтобы разрешить любой IP-трафик в сеть 192.168.2.0/24 (то есть 192.168.2.0 с маской подсети 255.255.255.0), соответствующий пункт списка доступа должен выглядеть так:

**access-list 10 permit 192.168.2.0 0.0.0.255**

# Преобразование адресов в двоичный вид

| | Десятичная форма | Двоичная форма |
|---|---|---|
| Шаблонная маска | 0.0.0.255 | 00000000.00000000.00000000.11111111 |
| Адрес в списке доступа | 192.168.2.0 | 11000000.10101000.00000010.00000000 |
| Целевой IP-адрес | 192.168.2.1 | 11000000.10101000.00000010.00000001 |

Так происходит вычисление:

Шаблонная маска 00000000.00000000.00000000.11111111

Список доступа 11000000.10101000.00000010.00000000

**Результат 1 11000000.10101000.00000010.11111111**

Шаблонная маска 00000000.00000000.00000000.11111111

Целевой IP-адрес 11000000.10101000.00000010.00000001

**Результат 2 11000000.10101000.00000010.11111111**

Результат 1 = Результат 2

## Вычисление шаблона для маски подсети

Для каждого байта маски подсети вычисляйте соответствующий байт шаблонной маски по формуле:

***Шаблон* = 255 - *Подсеть***

## Вычисление шаблона для маски подсети

Шаблонная маска, соответствующая маске подсети 255.255.255.224 (30 узлов в подсети), равна 0.0.0.31 (255 - 224 = 31). Далее показаны два пункта списка доступа, в которых применяются шаблонные маски:

```
! Для сети 192.168.2.64 255.255.255.224
access-list 10 permit 192.168.2.64 0.0.0.31
! Для сети 192.168.2.96 255.255.225.224
access-list 10 permit 192.168.2.96 0.0.0.31
```

## Обработка списков доступа

Список доступа выглядел так:

access-list 1 deny 10.10.1.0 0.0.0.255

access-list 1 deny 10.10.2.0 0.0.0.255

access-list 1 permit any

## Неявное запрещение

При каждом создании списка доступа маршрутизатор добавляет в конце его строку, означающую следующее: «если совпадения в списке не найдено, трафик нужно запретить». Если бы мы могли ее видеть, то эта строка выглядела бы так:

access-list 1 deny any

# Номера списков доступа

| Тип списка | Диапазон номеров |
|---|---|
| Стандартные списки доступа для протокола IP | 1–99 |
| Расширенные списки доступа для протокола IP | 100–199 |
| Код типа для протокола Ethernet | 200–299 |
| DECnet | 300–399 |
| XNS | 400–499 |
| Расширенный список XNS | 500–599 |
| AppleTalk | 600–699 |
| Ethernet-адрес | 700–799 |
| Novell | 800–899 |
| Расширенный список Novell | 900–999 |
| Novell SAP | 1000–1099 |
| Дополнительные стандартные списки доступа для протокола IP | 1300–1999 |
| Дополнительные расширенные списки доступа для протокола IP | 2000–2699 |
| Именованные списки доступа | Нет |
| Возвратные списки доступа | Нет |
| Динамические списки доступа | Нет |

## Расширенные списки доступа

Расширенные списки добавляют возможность фильтрации на основе протокола и порта, указанного в пакете.

Вот синтаксис расширенного списка доступа:

**access-list** номер действие протокол источник исх_порт приемник цел_порт[дополнительные_аргументы]

## Расширенные списки доступа

Параметры ***действие*** и ***источник*** те же, что и для стандартных списков доступа.

Прочие поля:

- ***номер***

  Идентификационный номер списка. Для расширенных списков доступа номер выбирается из диапазона от 100 до 199.

- ***протокол***

  Указание на протокол, к которому применяется правило. Допустимые значения: ip, tcp, udp и icmp.

- .

## Расширенные списки доступа

- ***исх порт***

Для TCP- и UDP-пакетов это исходящий порт пакета. Существует несколько способов указания портов. Это поле относится к необязательным: если протоколом является IP или ICMP, оно опускается.

- ***приемник***

Целевой адрес пакета; указывается так же, как и адрес источника. То есть можно задать IP-адрес с шаблонной маской, написать ключевое слово host и IP-адрес определенного узла либо использовать ключевое слово any.

## Расширенные списки доступа

- ***цел_порт***
- Для TCP- и UDP-пакетов это — порт назначения пакета. Существует несколько способов указания портов. Это поле относится к необязательным: если протоколом является IP или ICMP, оно опускается.
- ***дополнительные_аргументы***

Необязательное ключевое слово, которое указывается только для протокола TCP. Например, ключевое слово established является необязательным.

## Задание портов

- **lt n**

Все номера портов, меньшие *n.*

- **gt** *n*

Все номера портов, большие *n.*

- **eq** *n*

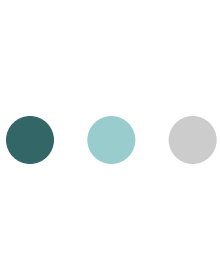
Порт *n.*

- **neq** *n*

Все порты, за исключением *n.*

- **range** *n m*

Все порты от n до m включительно.



```
access-list 110 permit tcp any host 10.10.1.5 eq 25
access-list 110 permit tcp any host 10.10.1.5 eq 80

access-list 110 deny tcp any any lt 1024

access-list 110 deny tcp any any range 3000 3010

access-list 110 permit udp any any eq 3535
```

## Установка соединений

К правилам доступа протокола TCP можно добавлять ключевое слово **established**. Технически это ключевое слово включает механизм отбора пакетов, для которых установлен бит АСК (acknowledgment — подтверждение) или RST (reset —сброс). Если установлен либо бит АСК, либо бит RST, маршрутизатор считает, что этот пакет не является первым пакетом сеанса, то есть сеанс уже начат.

## Установка соединений

```
! Список доступа для входящего трафика
access-list 110 permit tcp any any established
access-list 110 deny ip any any
! Список доступа для исходящего трафика
access-list 111 permit tcp any any eq telnet
access-list 111 deny ip any any
interface serial0
access-group 110 in
access-group 111 out
```

# Правила для протокола ICMP

Типичное правило для протокола ICMP:

**access-list 110 permit icmp any any echo-reply**

# Правила для протокола ICMP

```
! Разрешить входящие эхо-запросы в сеть
access-list 110 permit icmp any any echo
! Разрешить ответы на эхо-запросы
access-list 110 permit icmp any any echo-reply
! Разрешить подавление источника ICMP-пакетов (управление
потоком данных)
access-list 110 permit icmp any any source-quench
! Разрешить выбор маршрутов для MTU
access-list 110 permit icmp any any packet-too-big
! Разрешить пакеты с истекшим временем жизни, что полезно
! для трассировки маршрутов
access-list 110 permit icmp any any time-exceeded
! Запретить все прочие ICMP-пакеты
access-list 110 deny icmp any any
```

# Применение списка доступа к интерфейсу или линии связи

Для применения списка доступа к интерфейсу используйте команду **access-group**.

Вот два примера:

```
interface ethernet0
    ip access-group 110 in
    ip access-group 112 out
```

Чтобы применить стандартный список доступа к линии связи, используйте команду **access-class**, например:

```
line vty0
    access-class 10 in
```

## Именованные списки доступа

Операционная система IOS 11.2 и более поздние позволяют обходиться без номеров и давать спискам доступа осмысленные имена. Для создания простого списка доступа с именем **simplelist** используйте следующую команду:

**ipaccess-list standard simplelist**

Чтобы создать расширенный список доступа с именем **inboundfilter**, используйте команду:

**ipaccess-list extended inboundfilter**

# Именованные списки доступа

```
Router(config)#ip access-list standard filter
Router(config-std-nacl)#permit 10.10.1.0 0.0.0.255
Router(config-std-nacl)#deny 10.10.0.0 0.0.255.255
Router(config-std-nacl)#permit any any
```

## Именованные списки доступа

Чтобы применить именованный список доступа к интерфейсу, используйте команду access-group, но вместо номера укажите имя списка:

**interface serial 1**

**ip access-group filter1 in**

# Именованные списки доступа

```
Router(config)#ip access-list standard filter1
Router(config-std-nacl)#no permit 10.10.1.0 0.0.0.255
Router(config-std-nacl)#exit
Router#show access-list filter1
ip access-list standard filter1
deny 10.10.0.0 0.0.255.255
permit any any
```

## Ввод несмежных портов

Начиная с IOS версии 12.4 в системе реализована возможность ввода несмежных портов в одной строке именованного списка доступа.

**ip access-list extended acllist1**

**permit tcp any host 192.168.1.1 eq telnet www smtp pop3**

## Возвратные списки доступа

Возвратные списки доступа динамически изменяются в зависимости от того, какие службы нужны пользователям.

## Создание возвратного списка для исходящего трафика

```
ip access-list extended outlist
! Разрешается весь трафик; он добавляется в возвратный список tmplist
    permit tcp any any reflect tmplist
```

## Создание возвратного списка для входящего трафика

**ip access-list extended in list**

! разрешить tcp -трафик к нашему веб-серверу

**permit tcp any host 192.168.1.1 eq www**

! оценить временный список

**evaluate tmplist**

! запретить все остальное

**deny ip any any**

## Применение к интерфейсу возвратных списков для входящего и исходящего трафиков

```
interface serial0
    description Internet Gateway interface
    ip access-group inlist in
    ip access-group outlist out
```

## Применение к интерфейсу возвратных списков для входящего и исходящего трафиков

```
Routerl#show access-list
Extended IP access list inlist
permit tcp any host 192.168.1.1 eq www
evaluate tmplist
deny ip any any
Extended IP access list outlist
permit tcp any any reflect tmplist
Reflexive IP access list tmplist
```

# Установка тайм-аута для возвратного списка доступа

**ip reflexive-list timeout 200**

## Замечания относительно возвратных списков

- Возвратные списки предназначены для работы на маршрутизаторах-шлюзах (маршрутизаторах, через которые осуществляется подключение к Интернету, к совместно используемой магистральной сети, к сети другой компании или организации).
- Ключевое слово **reflect,** формирующее возвратный список, можно применять только в командах разрешения **(permit).**
- Записи в возвратных списках автоматически уничтожаются через определенный период бездействия, даже если сеанс не завершен.

## Замечания относительно возвратных списков

- Записи удаляются из временного списка после завершения сеанса.
- Во временных списках IP-адреса и порты исходящего трафика «меняются местами» с IP-адресами и портами входящего трафика.
- Возвратные списки не работают на таких протоколах, как FTP, в которых входной порт не соответствует выходному порту.

# Добавление комментариев в список доступа

В списки доступа можно добавлять комментарии, используя ключевое слово **remark.**

```
access-list 110 remark Блокировка трафика в 192.168.1.0. Вечные проблемы
access-list 110 deny ip 192.168.1.0 0.0.0.255 any
access-list 110 remark Боб целый день висит в Интернете, остановите Боба
access-list 110 deny tcp host 192.168.2.1 any eq www
```

## Настройка времени в списках доступа

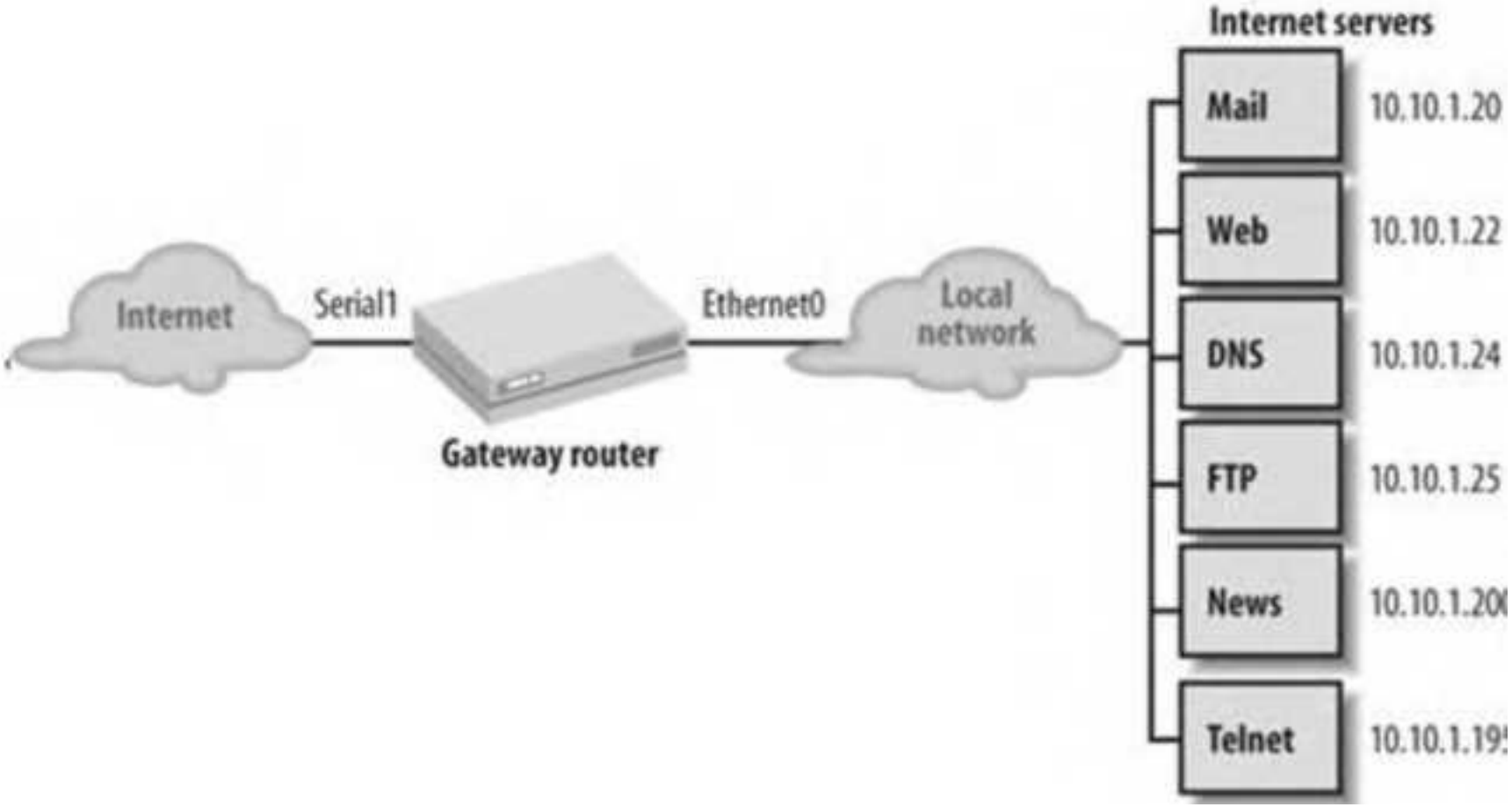
Можно использовать команду **time-range** для задания временного диапазона и после этого применить временной диапазон к правилам списка доступа, задав время их действия.

```
time-range block-http
        periodic weekdays 8:00 to 17:00
```

# Настройка времени в списках доступа

```
! Временной диапазон работает только в расширенных списках доступа
ip access-list extended list1
  ! block-http - это имя определенного ранее временного диапазона
  deny tcp any any eq www time-range block-http
  permit any any
```

# Создание маршрутизатора-шлюза



# Подделка IP-адреса

Подделкой (**spoofing**) IP-адресов называют преобразование сетевых пакетов, в котором IP-адреса источника соответствуют внутренним узлам.

## Подделка IP-адреса

```
! Список для блокировки подделанных адресов
access-list 111 deny ip 198.168.10.0 0.0.0.255 any
access-list 111 permit ip any any
! Интерфейс подключения к Интернету
! Для блокировки подделанных адресов нужно применить
! список 111 к входящим пакетам
interface serial 0
ip access-group 111 in
```

# Разрешение протокола FTP при помощи списка доступа

- Клиент запрашивает FTP-сеанс на FTP-сервере, используя порт **21 (ftp),** а сервер выполняет аутентификацию.
- FTP-клиент отправляет на FTP-сервер команду PORT. Эта команда говорит серверу, какой порт следует использовать для сеанса. FTP-клиент ожидает данные на этом порте.
- Сервер открывает новое соединение между портом 20 **(ftp-data)** и предоставленным портом FTP-клиента.
- Клиент и сервер начинают обмен данными через новый порт.

# Разрешение протокола FTP при помощи списка доступа

- Клиент запрашивает FTP-сеанс на FTP-сервере, используя порт **21 (ftp),** а сервер выполняет аутентификацию.
- FTP-клиент отправляет на FTP-сервер команду PORT. Эта команда говорит серверу, какой порт следует использовать для сеанса. FTP-клиент ожидает данные на этом порте.
- Сервер открывает новое соединение между портом 20 **(ftp-data)** и предоставленным портом FTP-клиента.
- Клиент и сервер начинают обмен данными через новый порт.

**access-list 110 permit tcp any eq ftp-data any gt 1024**

## Пассивное FTP-соединение

В случае пассивного FTP-соединения клиент отправляет серверу команду **PASV** вместо **PORT**. Это заставляет FTP-сервер обмениваться данными с FTP-клиентом через тот порт, который уже используется. Таким образом, входящие пакеты выглядят так, как будто они принадлежат уже установленному соединению на FTP-порте.

# Реальный список доступа

```
! Блокировать подделку наших IP-адресов
access-list 110 deny ip 10.10.1.0 0.0.0.255 any
! Разрешить возвращение обратно в сеть любых исходящих TCP-соединений
access-list 110 permit tcp any any established
! Разрешить передачу электронной почты (порт SMTP 25) на наш SMTP-сервер
access-list 110 permit tcp any host 10.10.1.20 eq smtp
! Разрешить веб-трафик (порт 80) только на наш веб-сервер
access-list 110 permit tcp any host 10.10.1.22 eq www
! Разрешить DNS-трафик на наш DNS-сервер; разрешить TCP и UDP
access-list 110 permit tcp any host 10.10.1.24 eq domain
access-list 110 permit udp any host 10.10.1.24 eq domain
```

# Реальный список доступа

! Разрешить внутренним узлам обращаться

! к внешнему DNS-серверу (192.168.1.100)

access-list 110 permit upd host 192.168.1.100 eq domain any gt 1023

! Разрешить FTP-трафик на наш FTP-сервер

access-list 110 permit tcp any host 10.10.1.25 eq ftp

access-list 110 permit tcp any host 10.10.1.25 eq ftp-data

! Разрешить передачу новостей внутреннему NNTP-клиенту

! только с легитимных NNTP-серверов

access-list 110 permit tcp host 198.168.1.98 host 10.10.1.200 eq nntp

access-list 110 permit tcp host 192.168.1.99 host 10.10.1.200 eq nntp

! Разрешить telnet-соединения (порт 23) только с одним узлом!

access-list 110 permit tcp any host 10.10.1.195 eq telnet

## Реальный список доступа

```
! Некоторые вещи необходимо запретить: Xwindows, NFS
access-list 110 deny tcp any any range 6000 6003
access-list 110 deny tcp any any range 2000 2003
access-list 110 deny tcp any any eq 2049
access-lis t 110 deny udp any any eq 2049
! Так как мы применяем непассивное FTP-соединение на
наших FTP-клиентах, следующая строка необходима, чтобы
разрешить возвращение этих FTP-сеансов обратно в сеть.
! Если у вас есть FTP-сервер, для него следует создать
! отдельный пункт.
access-list 110 permit tcp any eq ftp-data any gt 1024
```

# Реальный список доступа

! Разрешить ICMP-трафик в нашу сеть
! Внимание! ICMP - это больше, чем просто эхо-запрос. Если вы решите
! запретить его. то следует явно запретить определенные типы ICMP-команд
! (echo, echo-reply и т. п.).
! Механизмы выбора маршрутов для MTU и подавления источника работают
! благодаря протоколу ICMP и очень важны для некоторых соединений.
! Сначала запрещаем перенаправление широковещательных ICMP-рассылок
**access-list 110 deny icmp any any redirect**
! Затем разрешаем все остальное
**access-list 110 permit icmp any any**

# Оптимизация списков доступа

Router#show access-list 124

Extended IP access list 124

deny ip 10.10.1.0 0.0.0.255 any (1855 matches)

permit tcp any any established (6105063 matches)

permit tcp any host 10.10.1.20 eq smtp (10246 matches)

permit tcp any host 10.10.1.21 eq pop3 (11220 matches)

permit tcp any host 10.10.1.22 eq www (72583 matches)

# Оптимизация списков доступа

```
! Невозможно изменить первую строку: она ищет подделанные адреса
! и поэтому должна быть первой
access-list 124 deny ip 10.10.1.0 0.0.0.255 any
access-list 124 permit tcp any any established
access-list 124 permit tcp any host 10.10.1.22 eq www
access-list 124 permit tcp any host 10.10.1.21 eq pop3
access-list 124 permit tcp any host 10.10.1.20 eq smtp
```

# Эмуляция анализатора пакетов

Сделаем список доступа, захватывающий
весь IP-трафик, а затем добавим команду debug:
**access-list 110 permit ip any any**
**debug ip packet 110**

Применим список к интерфейсу в обоих направлениях:
interface ethernet0
**ip access-group 110 in**
**ip access-group 110 out**

Завершив анализ трафика, выключите режим регистрации:
**no debug ip packet 110**

# Регистрация попыток нарушения списка доступа

За счет учета IP-трафика можно регистрировать попытки нарушения списка доступа:

**ip accounting access-violations**

Ключевое слово log включает регистрацию записей, совпадающих с этой строкой:

**access-list 110 deny tcp any host 10.10.1.22 eq www log**

Ключевое слово **log-input** не только включает регистрацию пакетов, но и предоставляет сведения об исходном интерфейсе пакетов.

Если настроить маршрутизатор с применением команды **logging buffered**, то можно будет просматривать сохраненные записи журнала при помощи команды **show log**.

# Безопасное обновление списков доступа

Обеспечить безопасность, не отключая интерфейс, — использовать команду **access-group**. В этом случае в любой момент у интерфейса может быть только один входящий и один исходящий список доступа.

Например:

**interface ethernet 0**

**ip address 10.10.1.1 255.255.255.0**

**ip access-group 110 in**

**ip access-group 115 out**

**interface ethernet 0**

**lp access-group 112 in**

# Загрузка списка на маршрутизатор по протоколу TFTP, RCP или SCP

- При помощи протокола TFTP, RCP или SCP загрузите всю конфигурацию на сервер:

  **Router#copy running-config tftp**
- Отредактируйте файл конфигурации, удалив все, за исключением списков доступа.
- Добавьте команду **no access-list** перед каждым списком доступа в файле:

  **no access-list 10**

  **access-list 10 deny 1.2.3.4 1.5.6.7**
- Отредактируйте списки доступа и сохраните файл, чтобы в будущем не повторять шаги 1-3.

# Загрузка списка на маршрутизатор по протоколу TFTP, RCP или SCP

- При помощи протокола TFTP, RCP или SCP скопируйте файл обратно на маршрутизатор.

  **Router#copy tftp running-config**

- Этот файл будет содержать только списки доступа, а не всю конфигурацию маршрутизатора; он удалит список доступа, а затем создаст новый с нужной конфигурацией:
- Когда работа новых списков доступа будет вас полностью устраивать, сохраните рабочую конфигурацию:

  **Router#copy running-config startup-config**