Трансляция сетевых адресов (NAT)

Статическая NAT отображает один внутренний адрес на один внешний.
Динамическая NAT отображает частный IP-адрес на один из свободных из группы зарегистрированных IP-адресов .

Каждый внутренний сетевой адрес компьютера клиента отображается на один и тот же внешний IP-адрес, но с разными номерами портов - **Port Address Translation (PAT).**

Трансляция сетевых адресов (NAT)

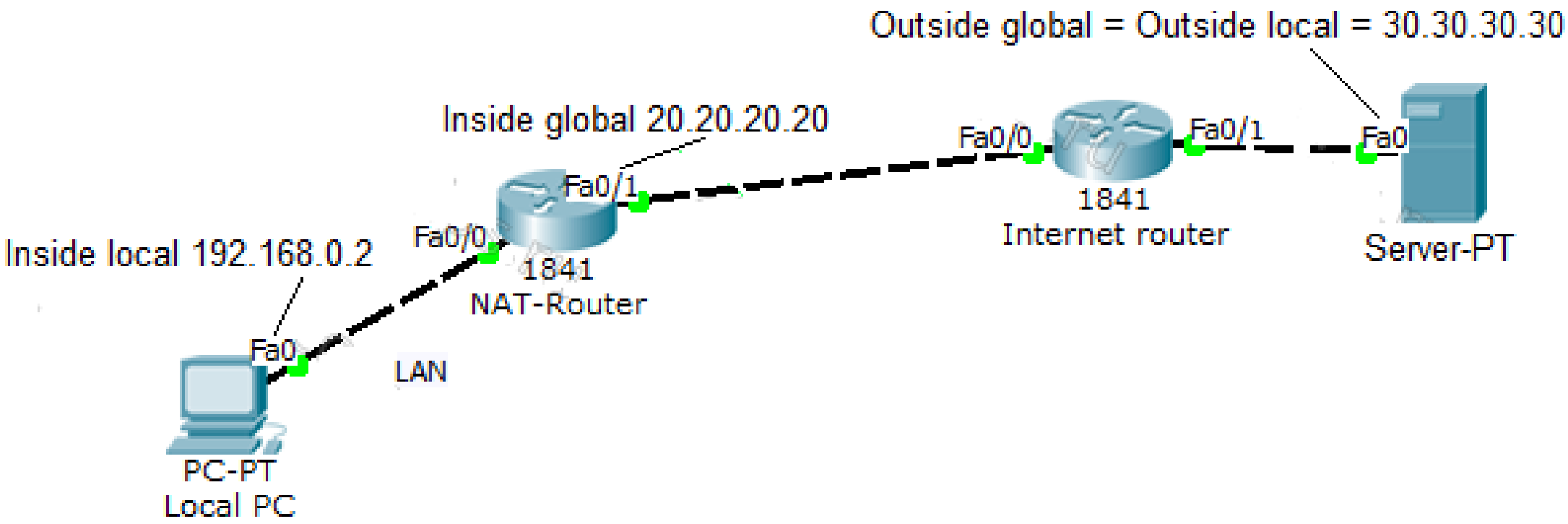
Различают 4 адреса, которые фигурируют в процессе трансляции адресов:

1) **Inside local**; 2) **Inside global**; 3) **Outside local**; 4) **Outside global**.

Inside local – Это частный адрес компьютера локальной сети, которому предоставлена возможность работать с сетью Интернет. Пакет запроса на обслуживание сервером отправляется на адрес внешнего маршрутизатора глобальной сети (***outside global***).

Когда пакет проходит через NAT-модуль, то адрес отправителя подменяется на публичный адрес, имеющийся в распоряжении NAT-модуля (***inside global***). Если сервер-получателя имеет публичный адрес ***outside global*** и доступен по нему извне, то ***outside global*** и ***outside local*** – **совпадают**, если сам сервер тоже скрыт за каким-то NAT-устройством, то именно оно получает вместо него запрос на ***outside global*** адрес и транслирует адрес получателя на ***outside local*** (во внутренней сети получателя).

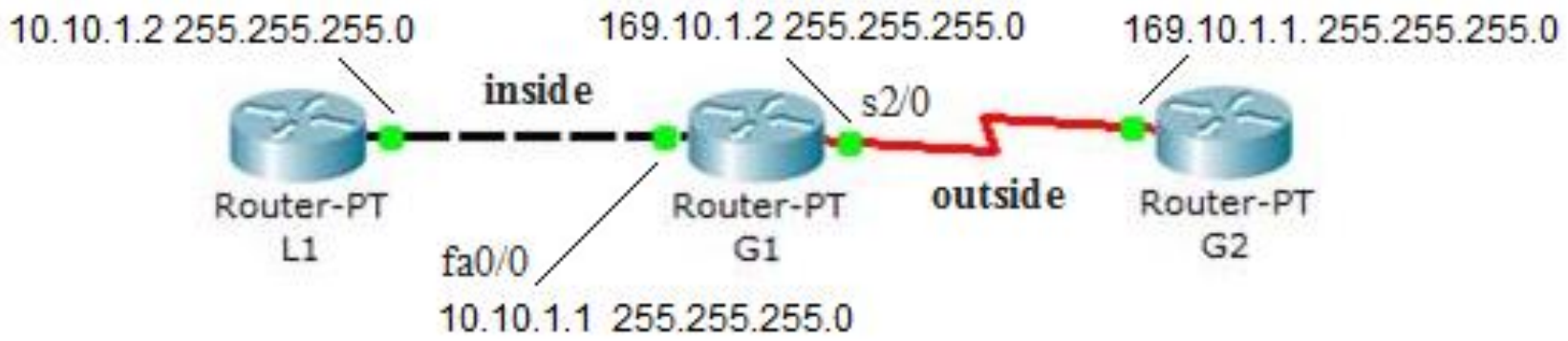
Трансляция сетевых адресов (NAT)



Статическая трансляция сетевых адресов (NAT)

Задание режима статической трансляции осуществляется путем ввода команды

R1(config)**#ip nat inside source static <локальный адрес> <глобальный адрес>**



G1(config)#ip nat inside source static 10.10.1.2 169.10.1.2

G1(config)#interface fa0/0

G1(config-if)# ip nat inside

G1(config-if)#interface s2/0

G1(config-if#ip nat outside

G1#**show ip nat translations**

Pro	Inside global	Inside local	Outside local	Outside global
---	169.10.1.2	10.10.1.2	---	---

Динамическая трансляция сетевых адресов (NAT)

Динамическая трансляция предполагает назначения диапазона адресов (адресного пула) с указанием начального и конечного адресов и маски сети.

ip nat pool <имя> <первый адрес> <последний адрес> netmask < обратная маска подсети> или prefix-length <длина префикса>

Определение пула адресов **pool1**

G1(config)#**ip nat pool pool1 169.10.1.50 169.10.1.100 netmask 255.255.255.0**

Задание для модуля NAT преобразование адресов из списка 1 в пул адресов **pool1**

G1(config)#**ip nat inside source list 1 pool pool1**

Создание списка доступа 1 для внутренних адресов, которые будут преобразовываться

G1(config)#**access-list 1 permit 10.10.1.0 0.0.0.255**

G1(config)#interface fa0/0

G1(config-if)# ip nat inside

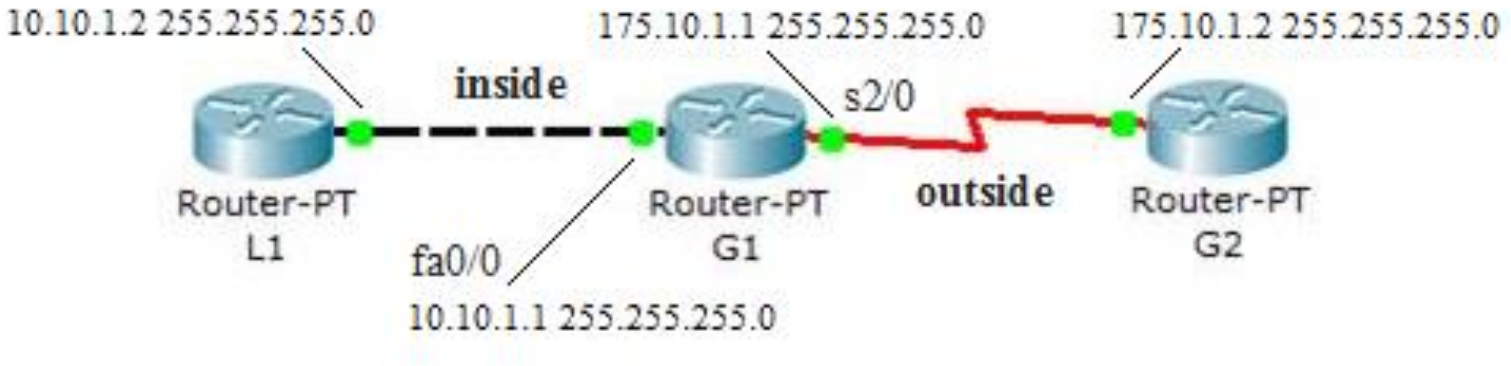
G1(config-if)#interface s2/0

G1(config-if#ip nat outside

G1#show ip nat

Pro	Inside global	Inside local	Outside local	Outside global
tcp	169.10.1.50:1025	10.10.1.2:1025	175.10.1.2:23	175.10.1.2:23
tcp	169.10.1.50:1026	10.10.1.2:1026	175.10.1.2:23	175.10.1.2:23
tcp	169.10.1.50:1027	10.10.1.2:1027	175.10.1.2:23	175.10.1.2:23
tcp	169.10.1.50:1028	10.10.1.2:1028	175.10.1.2:23	175.10.1.2:23
tcp	169.10.1.50:1029	10.10.1.2:1029	175.10.1.2:23	175.10.1.2:23

Трансляция сетевых адресов (портов) (PAT)



G1(config)#**ip nat inside source list 1 interface s2/0 overload**

G1(config)#**access-list 1 permit 10.10.1.0 0.0.0.255**

G1(config)#**interface fa0/0**

G1(config-if)# **ip nat inside**

G1(config-if)#**interface s2/0**

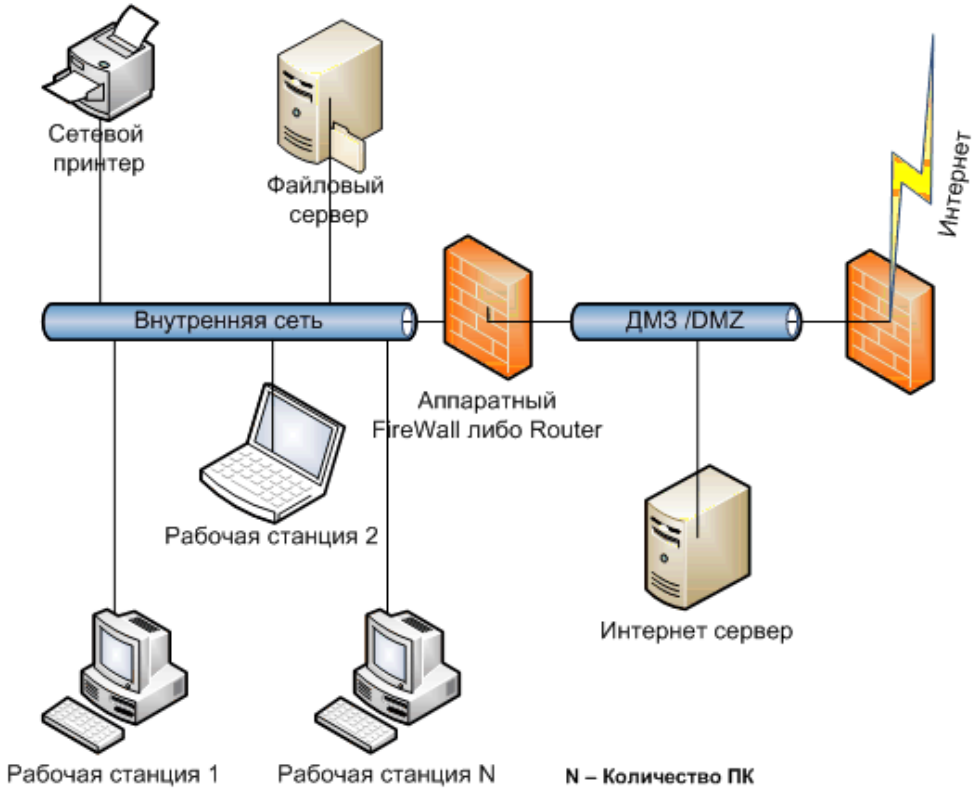
G1(config-if#**ip nat outside**

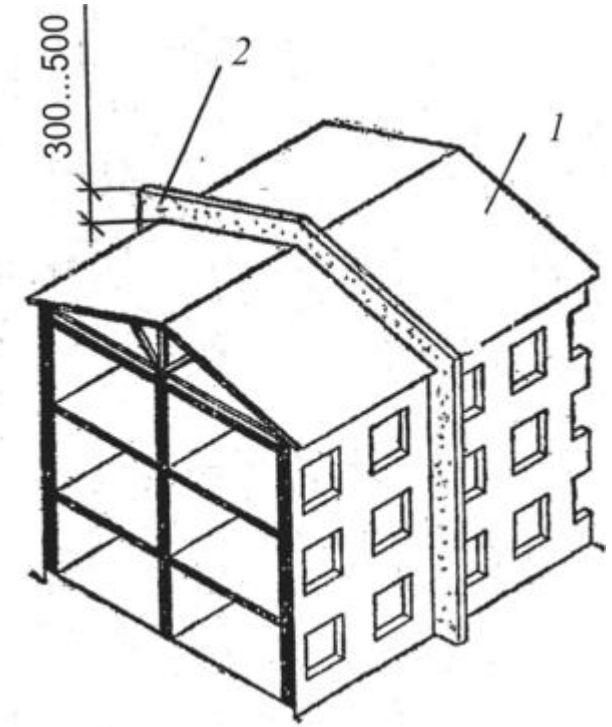
G1#show ip nat

Pro	Inside global	Inside local	Outside local	Outside global
tcp	175.10.1.1:1025	10.10.1.2:1025	175.10.1.2:23	175.10.1.2:23
tcp	175.10.1.1:1026	10.10.1.2:1026	175.10.1.2:23	175.10.1.2:23
tcp	175.10.1.1:1027	10.10.1.2:1027	175.10.1.2:23	175.10.1.2:23
tcp	175.10.1.1:1028	10.10.1.2:1028	175.10.1.2:23	175.10.1.2:23
tcp	175.10.1.1:1029	10.10.1.2:1029	175.10.1.2:23	175.10.1.2:23

Демилитаризованная зона (DMZ)

DMZ (*Demilitarized Zone*, ДМЗ) — сегмент сети, содержащий общедоступные сервисы и отделяющий их от частных. В качестве общедоступного может выступать, например, веб-сервис, при этом другие локальные ресурсы (например, файловые серверы, рабочие станции) необходимо изолировать от внешнего доступа.

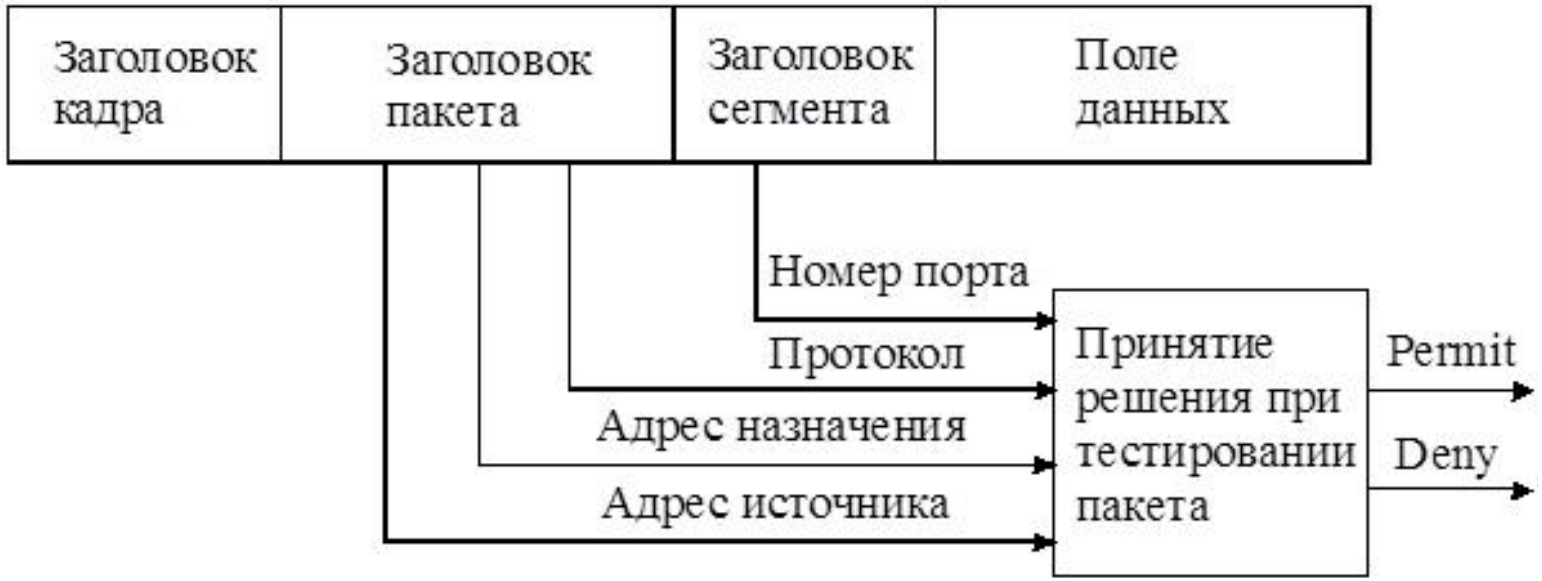




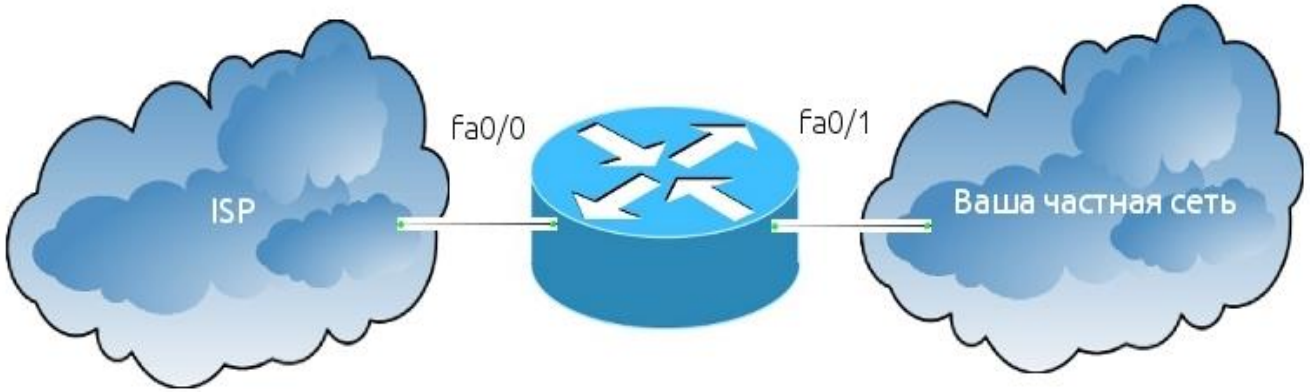
СПИСКИ УПРАВЛЕНИЯ ДОСТУПОМ *ACL* (*Access Control Lists*)

Списки управления доступом являются частью комплексной системы безопасности сети. Они содержат набор инструкций (**директив**) какие **порты** и **адреса** блокировать, а какие наоборот разрешить.

Включают перечень особых директив (предписаний): «**разрешить**» (***permit***) и «**запретить**» (***deny***). В процессе приема кадра осуществляется проверка полей заголовка пакета и сегмента.



СПИСКИ УПРАВЛЕНИЯ ДОСТУПОМ *ACL - ФУНКЦИОНИРОВАНИЕ*



Пакет из локальной частной сети приходит 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 в списке доступа разрешается проходить пакету, то маршрутизатор отправляет пакет провайдеру через интерфейс **fa0/0**, если список доступа не разрешает проходить пакету, пакет уничтожается. Если список доступа отсутствует — пакет пропускается без всяких ограничений. Перед тем как отправить пакет провайдеру, маршрутизатор ещё проверяет интерфейс **fa0/0** на наличие исходящего ACL.

ACL не оказывает никакого влияния на трафик, генерируемый самим маршрутизатором.

СПИСКИ УПРАВЛЕНИЯ ДОСТУПОМ *ACL* (*Access Control Lists*)

Каждое предписание в списке доступа записывается отдельной строкой. Для одного списка можно определить несколько директив.

В конце каждого списка стоит неявное правило «deny all».

Для протокола IP поддерживаются списки доступа:

- ✓ **стандартные** (проверяют только адрес отправителя пакета, номера 1-99);
- ✓ **расширенные** (проверяют адрес отправителя, адрес получателя, порты, тип протокола и др. номера 100-199).
- ✓ **динамические (Dynamic ACL)**, в котором некоторые строчки списка до поры до времени не работают, но когда администратор подключается к маршрутизатору по протоколу **telnet**, эти строчки включаются на ограниченное время, то есть администратор может оставить для себя «дыру» в безопасности для отладки или выхода в сеть.

СПИСКИ УПРАВЛЕНИЯ ДОСТУПОМ *ACL*

TimeBased ACL — временные ACL, у которых некоторые строчки срабатывают только в какое-то время. Например, с помощью таких ACL легко настроить, чтобы в офисе доступ в Интернет был только в рабочее время.

Reflexive ACL — зеркальные списки контроля доступа, позволяют запоминать, кто обращался из данной сети наружу (с каких адресов, с каких портов, на какие адреса, на какие порты) и автоматически формировать зеркальный ACL, который будет пропускать обратный трафик извне вовнутрь только в том случае, если изнутри было обращение к данному ресурсу.

ПРИМЕРЫ СТАНДАРТНЫХ СПИСКОВ ДОСТУПА

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

remark - комментарий; **source-wildcard** – инвертированная маска **000…01111**

0 в обратной маске означает проверку этого бита адреса, а **1** означает, что проверка этого бита не производится.

Any — это специальное слово, которое заменяет адрес сети и обратную маску соответствующие 0.0.0.0 0.0.0.0 и означает, что под правило проверки подпадают абсолютно все хосты из любых сетей.

Специальное слово — **host** означает, что неявно используется маска 0.0.0.0 — то есть проверке подлежит один единственный указанный адрес.

Пример 1.

Маршрутизатор должен разрешить прохождение трафика из сети только компьютеру (хосту) с адресом 192.168.3.2.

access-list 1 permit 192.168.3.2 0.0.0.0

ПРИМЕРЫ СТАНДАРТНЫХ СПИСКОВ ДОСТУПА

Пример 2.

Разрешить прохождение пакетов через маршрутизатор от всех хостов сети с номером 140.12.11.0, кроме хостов 140.12.11.5 и 140.12.11.6, а также разрешить прохождение всего остального трафика через интерфейс, на котором установлен список доступа:

access-list 2 deny host 140.12.11.5

access-list 2 deny host 140.12.11.6

access-list 2 permit 140.12.11.0 0.0.0.255

access-list 2 permit any

ПРИМЕРЫ РАСШИРЕННЫХ СПИСКОВ ДОСТУПА

Router(config)#access-list access-list-number {**deny** | **permit**} **protocol source** \ source-wildcard **destination** destination-wildcard \ [precedence precedence] [tos tos] [established] [log]

established: разрешается прохождение TCP-сегментов, которые являются частью уже созданной TCP-сессии.

log: вызывает выдачу записи о совпадении пакета с данным критерием на консоль и в системный **лог-файл.**
tos: Type of Service (тип обслуживания); **precedence**: приоритет.

Пример:

Блокировать (запретить) доступ TCP пакетов со всех хостов к серверу с IP-адресом 140.12.11.10

!**access-list** 102 deny TCP 0.0.0.0 255.255.255.255 140.12.11.10 0.0.0.0
!или сокращенная запись:
access-list 102 deny TCP any host 140.12.11.10

ОБРАБОТКА СПИСКОВ ДОСТУПА

Трафик, поступающий на маршрутизатор, сравнивается с записями ACL на основе очередности появления записей в маршрутизаторе. Новые записи добавляются в конец списка. Маршрутизатор продолжает поиск до нахождения соответствия. Если маршрутизатор доходит до конца списка, не найдя соответствий, трафик не принимается. По этой причине наиболее часто используемые записи должны располагаться в начале списка. Существует **неявный запрет** на трафик, который не разрешен. Список ACL с единственной записью “**deny**” приводит к запрету всего трафика. Необходимо использовать как минимум одну разрешающую запись ACL, иначе весь трафик будет блокироваться.

Результаты применения этих двух списков ACL (101 и 102) аналогичны.

access-list 101 permit ip 10.1.1.0 0.0.0.255 172.16.1.0 0.0.0.255

access-list 102 permit ip 10.1.1.0 0.0.0.255 172.16.1.0 0.0.0.255
access-list 102 deny ip any any

СПИСКИ УПРАВЛЕНИЯ ДОСТУПОМ *ACL*

После того, как список создан, необходимо определить направление (входящий или исходящий) трафика и на каком интерфейсе он будет фильтроваться:

Router(config-if)# ip access-group номер_списка in | out

Пример назначения списка доступа 33 интерфейсу fast ethernet:

Router(config)#interface fa 0/1
Router(config-if)#ip access-group 33 in

Запретить весь TCP трафик от любого хоста на конкретный хост с адресом 172.16.1.5. Причем запрет действует при условии, что запросы идут на порты получателя от 5001 и выше

Router(config)#access-list 100 deny tcp any host 172.16.1.5 gt 5000

Для просмотра настроек используются следующие команды:

Router# show running-config
Router# show ip access-lists

ИМЕНОВАННЫЕ СПИСКИ УПРАВЛЕНИЯ ДОСТУПОМ *NACL*

Ничем не отличаются от стандартных и расширенных списков, однако позволяют гибко редактировать вновь созданные списки.

Стандартные и расширенные списки **редактировать нельзя**. К примеру, нельзя в середину списка вставить команду или удалить ее. Для этого нужно сначала **деактивировать список на самом интерфейсе**, а затем полностью его удалить и настроить заново.

Именованный список **требует** использовать **названия** списков **вместо их номеров**. Все введенные команды нумеруются **с шагом 10**, что позволяет легко добавлять и удалять команды.

Для стандартных именованных списков:

Router(config)# ip access-list standard название
Router(config-std-nacl)# deny IP_адрес отправителя инверт_маска

Чтобы удалить ненужную команду достаточно узнать ее номер. Для этого нужно ввести команду:

Router# show ip access-list название *затем ввести команду* ***удаление строки (no)***
Router(config-ext-nacl)# no 10

РЕДАКТИРОВАНИЕ ИМЕНОВАННЫХ СПИСКОВ УПРАВЛЕНИЯ ДОСТУПОМ *NACL*

У именованных списках ACL можно удалять конкретные записи, а также добавлять записи между имеющимися правилами с присвоением им номера между номерами правил, между которыми добавляется новое правило.

Например, имеется ACL с темя записями:

R1# show access-lists

Standard IP access-list WEBSERVER

10 permit 192.168.10.10

20 deny 192.168.10.0, wildcard bits 0.0.0.255

30 deny 192.168.12.0, wildcard bits 0.0.0.255

Пусть надо добавить еще одно правило:

R1(config)# access-list standard WEBSERVER

R1(config-std-nacl)# 15 permit 192.168.10.13

Вот, что получилось:

R1# show access-lists

Standard IP access-list WEBSERVER

10 permit 192.168.10.10

15 permit 192.168.10.13

20 deny 192.168.10.0, wildcard bits 0.0.0.255

30 deny 192.168.12.0, wildcard bits 0.0.0.255

ПРОСМОТР И ПРОВЕРКА СПИСКОВ УПРАВЛЕНИЯ ДОСТУПОМ

Используется команда:
Router# show access-lists *{access-list-mumber | name}*

Например,
R1# show access-lists - выводит все ACL
R1# show access-lists 10 - выводит ACL с номером 10
R1# show access-lists NAM - выводит ACL с именем NAM