

ПРОКСИ

6.0.1.1

Совокупность инструментальных средств (программного и аппаратного обеспечения), предназначенную для контроля доступа к сетевым ресурсам принято называть *прокси* (proxy).

6.0.2.1

Задачи, выполняемые прокси:

1. *Аутентификация* (authentication) -- определение круга пользователей, имеющих права доступа к сетевым ресурсам.

Если рассматривать более подробно, то в рамках аутентификации, можно выделить:

1. *Идентификацию* (identification) -- назначение пользователям (субъектам) и ресурсам (объектам) уникальных символьных или цифровых обозначений, то есть имен и названий, или идентификаторов в ОС.
2. Собственно *аутентификацию* -- обеспечение гарантии, что пользователи являются теми, за кого они себя выдают.
3. *Авторизацию* (authorization) -- назначение аутентифицированным пользователям прав, что обычно неотделимо от аутентификации.

6.0.2.2

Аутентификация обычно выполняется по *учетной записи* (account), то есть совокупности имени пользователя и пароля.

В общем же случае, может выполняться как более сложно, например, по карте доступа или биометрически, так и более просто, например, по IP-адресу или MAC-адресу.

Аутентификация может проводиться:

1. *Локально* -- запрос обрабатывается на том же устройстве, которое обеспечивает доступ, или к которому требуется доступ.
2. *Удаленно* -- запрос перенаправляется на внешний выделенный для этого сервер по специальным протоколам, таким как RADIUS (Remote Authentication Dial In User Services) и TACACS+ (Terminal Access Controller Access Control System Plus).

6.0.2.3

Если прокси «не виден» для клиентского ПО, то его называют **прозрачным (transparent)**.

6.0.2.4

2. **Фильтрация (filtering)** -- запрет или разрешение прохождения входящих или исходящих пакетов по выбранным критериям.

В качестве объекта фильтрации выступает пакет.

Может выполняться по IP-адресам, по портам, по содержимому и так далее.

6.0.2.5

3. Сетевой (межсетевой) экран (firewall, по-немецки brandmauer) -- запрет или разрешение доступа к определенным категориям сетевых ресурсов (как правило централизованным или внешним).

Сетевой экран в основном выполняет фильтрацию, но это более общее понятие.

6.0.2.6

Классификация сетевых экранов:

1. Packet Firewalls -- просто пропускают или отбрасывают пакеты. Работают на третьем уровне (очень редко на втором).
2. Stateful Firewalls -- способны следить (tracking) за состоянием TCP-соединений, то есть выполнять *инспекцию* (inspection) трафика. Работают на третьем и четвертом уровнях.
3. Application Gateway Firewalls -- способны следить за сообщениями протоколов прикладного уровня (например, HTTP), то есть выполнять *глубокую инспекцию* -- DPI (Deep Packet Inspection). Работают на третьем -- седьмом уровнях.

6.0.2.7

4. *Безопасность* (security) или, точнее, кибербезопасность (cybersecurity) -- в данном контексте, обеспечение защиты информации, передаваемой по открытым для прослушивания сетям.

Еще более общее чем сетевой экран понятие.

Задачи, решаемые в рамках обеспечения безопасности:

1. *Аутентификация* -- в данном контексте, обеспечение гарантии, что сообщение пришло от того, от кого его ожидают. Как правило, заключается в манипулировании с ключами. Алгоритмы: PSK, RSA и другие.

2. *Целостность* (integrity) -- обеспечение гарантии, что при пересылке сообщение не было повреждено и не было подменено. Как правило, заключается в подсчете значений хэш-функций. Алгоритмы: MD5 и SHA-256 и другие.

3. *Конфиденциальность* (confidentiality) -- обеспечение гарантии, что перехваченное сообщение не может быть прочитано. Как правило, заключается в шифровании данных. Алгоритмы: 3DES, AES и другие.

6.0.2.8

Как вариант, возможна маскировка конфиденциальных данных под неконфиденциальные, выражаяющаяся в различных алгоритмах *стеганографии* (steganography).

6.0.2.9

Во многие алгоритмы для формирования доверительных отношений между абонентами (security associations) заложено использование *цифровых подписей* (digital signatures) или *цифровых сертификатов* (digital certificates).

При этом гарантом ответственности (nonrepudiation) может выступать третья сторона, которой доверяют обе взаимодействующие стороны.

6.0.2.10

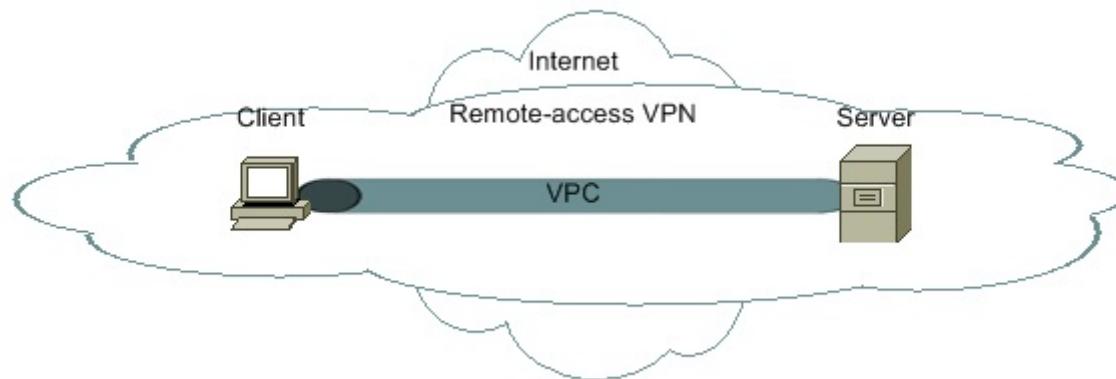
Современные тенденции в области безопасности компьютерных сетей сводятся к формированию так называемых *виртуальных частных сетей* -- VPNs (Virtual Private Networks), охватывающих взаимодействующие станции.

При этом взаимодействие осуществляется по формируемым особым образом (с целью максимальной защиты) *виртуальным частным каналам* -- VPCs (Virtual Private Channels), которые на практике обычно представляют собой защищенные туннели, проложенные через открытые для прослушивания сети.

6.0.2.11

Все VPNs можно разделить на два типа:

1. Site-to-site -- в рядовом случае связывают одноранговые шлюзы и являются статическими (например, IPsec VPNs).
2. Remote-access -- в рядовом случае обеспечивают подключение удаленных пользователей, создаются динамически и базируются на клиент-серверной модели (например, TLS VPNs).



6.0.2.12

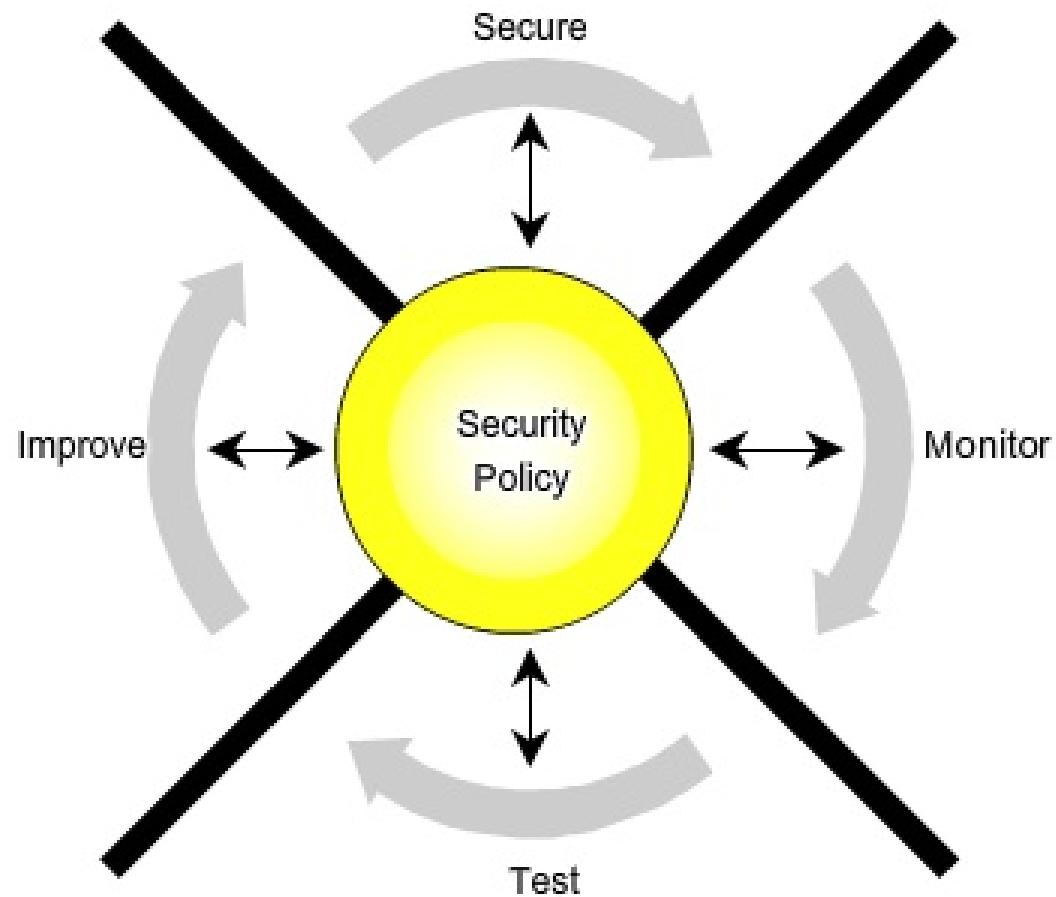
При администрировании, существуют две основополагающие политики обеспечения безопасности:

1. Разрешено все (по умолчанию), что не запрещено (явно).
2. Запрещено все, что не разрешено.

В настоящее время наиболее оправданным признан второй вариант.

6.0.2.13

Network Security Wheel



Security wheel [Cisco]

6.0.2.14а

Вопросы компьютерной безопасности, в том числе сетевой, как известно, находятся под контролем государства.

Ключевые стандарты Беларуси:

СТБ 34.101.1-2004 (ИСО/МЭК 15408-9) «Информационные технологии и безопасность. Критерии оценки безопасности информационных технологий. Часть 1. Введение и общая модель».

СТБ 34.101.2-2004 (ИСО/МЭК 15408-2:1999) «Информационные технологии и безопасность. Критерии оценки безопасности информационных технологий. Часть 2. Функциональные требования безопасности».

СТБ 34.101.3-2004 (ИСО/МЭК 15408-3:1999) «Информационные технологии и безопасность. Критерии оценки безопасности информационных технологий. Часть 3. Гарантийные требования безопасности».

СТБ 34.101.4-2004 «Информационные технологии. Методы и средства безопасности. Профиль защиты электронной почты предприятия».

СТБ П 34.101.8-2003 «Информационные технологии. Методы и средства безопасности. Программные средства защиты от воздействия вредоносных программ и антивирусные программные средства. Общие требования».

6.0.2.14b

СТБ 34.101.11 -2004 «Информационные технологии. Методы и средства безопасности. Критерии оценки безопасности информационных технологий. Профиль защиты операционной системы сервера для использования в доверенной зоне корпоративной сети».

СТБ П 34.101.14-2004 «Информационные технологии. Методы и средства безопасности. Критерии оценки безопасности информационных технологий. Профиль защиты программных средств маршрутизатора для использования в демилитаризованной зоне корпоративной сети»

СТБ 34.101.30-2007 «Информационные технологии. Методы и средства безопасности. Объекты информатизации. Классификация».

СТБ 1176.1-99 «Информационная технология. Криптографическая защита информации. Функция хэширования».

СТБ 1176.2-99 «Информационная технология. Криптографическая защита информации. Процедуры выработки и проверка электронной цифровой подписи».

СТБ ГОСТ Р 50922-2000 «Защита информации. Основные термины и определения»

6.0.2.15

5. *Ведение журналов* (logging, auditing) -- протоколирование различных событий, связанных с доступом к сетевым ресурсам.

Может выполняться по самым разным критериям.

6.0.2.16

6. Отслеживание угроз (threat monitoring).

Стоит немного особняком в отношении задач обеспечения защиты информации и ведения журналов.

Если сделать упор на реакцию при возникновении угроз, то можно выделить два типа прокси:

1. IDSes (Intrusion Detection Systems) -- своеобразные сенсоры, которые отслеживают вредоносный трафик по сигнатурам и различными способами оповещают при его обнаружении. Обычно трафик через них не «пропускается», а копируется в их сторону для параллельного анализа (promiscuous).

2. IPSes (Intrusion Prevention Systems) -- не просто отслеживают вредоносный трафик, а и способны самостоятельно его заблокировать. Обычно трафик «пропускается» через них (inline).

6.0.2.17a

Cisco выделяет три типа вредоносных атак:

1. Reconnaissance Attacks -- разведывательные, целью которых является несанкционированный сбор информации.

Примеры: просмотр содержимого пакетов снifferами (packet sniffing), сканирование адресов в поисках станций (ping sweeping), сканирование портов в поисках сервисов (port scanning), ловля на доверие (phishing), социальная инженерия (social engineering), поиск информации в Internet (Internet information queries).

6.0.2.17b

2. Access Attacks -- связанные с доступом, целью которых является получение несанкционированного доступа к информации или подмена информации.

Примеры: подбор паролей методом «грубой силы» (brute-force password search), использование имеющихся прав не по назначению (trust exploitation), перенаправление пользовательских запросов на ложные серверы (port redirection), различные варианты подмены информации в каналах (man-in-the-middle), использование уязвимостей ПО (buffer overflow).

6.0.2.17c

3. DoS (Denial of Service) Attacks -- связанные с сервисами, целью которых является отказ в обслуживании по тому или иному протоколу.

DDos (Distributed DoS) отличается тем, атаку проводят множество станций.

Примеры: ping с длиной пакета 65535 Byte с целью «завешивания» некоторых старых ОС (ping of dead), порождение с помощью особенностей SNMP-запросов многочисленных станций-«зомби» с целью «забрасывания» SNMP-ответами выбранной станции-«жертвы» (smurf), последовательное создание многочисленных полуоткрытых TCP-соединений (TCP SYN flooding).

6.0.2.18а

Cisco выделяет следующие традиционные типы компьютерных преступников:

1. Hackers (хакеры) -- наиболее общий термин, но характерной чертой является наличие знаний в области компьютерных технологий.
2. Black Hats -- злоумышленники, которые могут и не обладать большими знаниями.
3. White Hats -- выполняют несанкционированные атаки, но из благих побуждений (например, сообщают администратору об обнаруженных проблемах).



6.0.2.18b

4. Crackers (взломщики) -- специализируются на взломе защиты КС, или ПО, или еще чего-либо.
5. Spammers -- массово рассылают электронную почту.
6. War Drivers -- путешествуют в поисках «халявы» (обычно незащищенных беспроводных сетей).
7. Phishers (от phone fishers) -- пытаются под различными предлогами «выудить» конфиденциальную информацию.
8. Phrickers (от phone freakers) -- используют особенности телефонных сетей для совершения преступлений.

6.0.2.19

Cisco выделяет три типа вредоносных программ:

1. Viruses (вирусы) -- наиболее общий термин, но характерной чертой является распространение с помощью внедрения вредоносного кода в пользовательские программы.

2. Worms (черви) -- характерной чертой является самостоятельное распространение посредством СПД, протекающее в три фазы: поиск или создание известных вирусу заранее уязвимостей (*enabling vulnerability*), внедрение путем копирования через сеть (*penetration*), вредоносное проявление (*payload*).

3. Trojan horses (троянские кони) -- характерной чертой является маскировка под «безобидные» программы.

6.0.2.20

7. Акселерация (acceleration) -- ускорение доступа к сетевым ресурсам за счет определенных оптимизаций.

Основные способы:

1. Кэширование.
2. Многопоточность.
3. Поддержка «докачки».

6.0.2.21

8. *Формирование трафика (traffic shaping)* -- распределение приоритетов при доступе к сетевым ресурсам по определенным критериям.

Может быть программным или аппаратным, статическим или динамическим.

Может осуществляться по разным критериям, например, по времени.

6.0.2.22

9. Преобразование адресов (address translation).

Особую проблему при организации коллективного доступа в Internet представляет собой «невидимость» приватных адресов.

Первоначально задачу можно сформулировать так: требуется, чтобы несколько пользовательских станций из внутренней подсети могли совместно пользоваться одним публичным адресом.

NAT (Network Address Translation) -- наиболее общий стандарт (RFC 3032 плюс RFC 2663), решающий задачу путем прозрачной подмены адресов на маршрутизаторах.

Обобщенный алгоритм работы IP NAT:

1. Клиент передает пакет прокси с поддержкой NAT.
2. Прокси запоминает IP-адрес назначения, IP-адрес источника, подставляет в качестве IP-адреса источника свой адрес и передает пакет серверу, запрашиваемому клиентом.
3. После получения ответного пакета от сервера выполняются обратные преобразования на основе запомненной информации.
4. Ответный пакет передается клиенту.

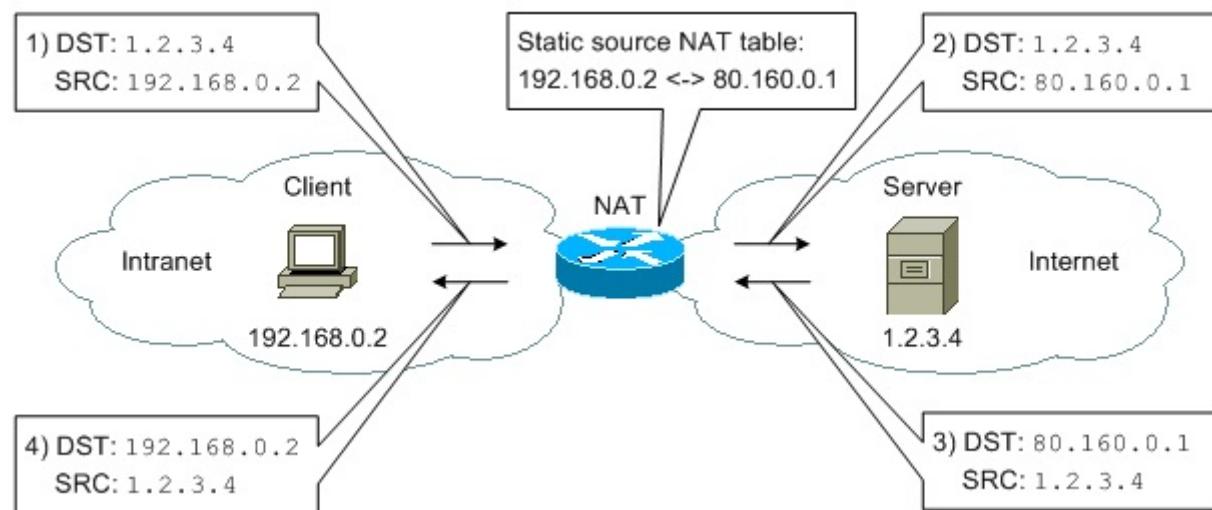
6.0.2.23

Для обеспечения правильности выполнения преобразований строится таблица, то есть NAT работает по табличному принципу.

6.0.2.24а

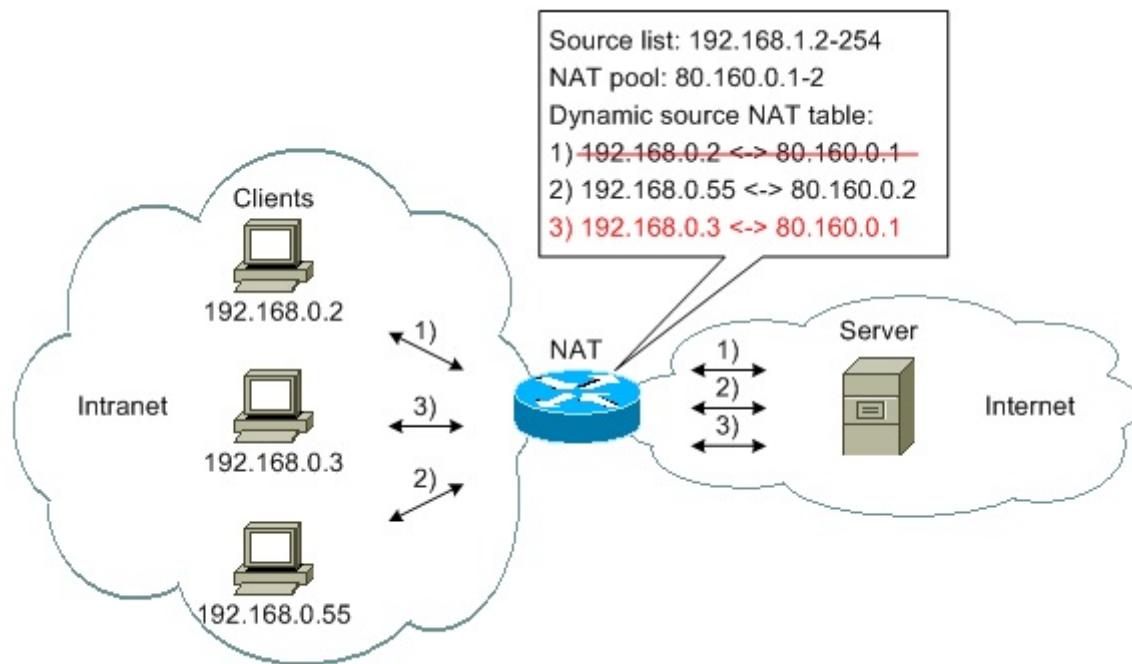
Все реализации NAT, в первую очередь, делят на два типа:

1. Статические (static) -- преобразования осуществляется исходя из строгого соответствия пар адресов.



6.0.2.24b

2. Динамические (dynamic) -- при преобразованиях адреса' по мере надобности выделяются из пулов по определенным критериям.



6.0.2.25а

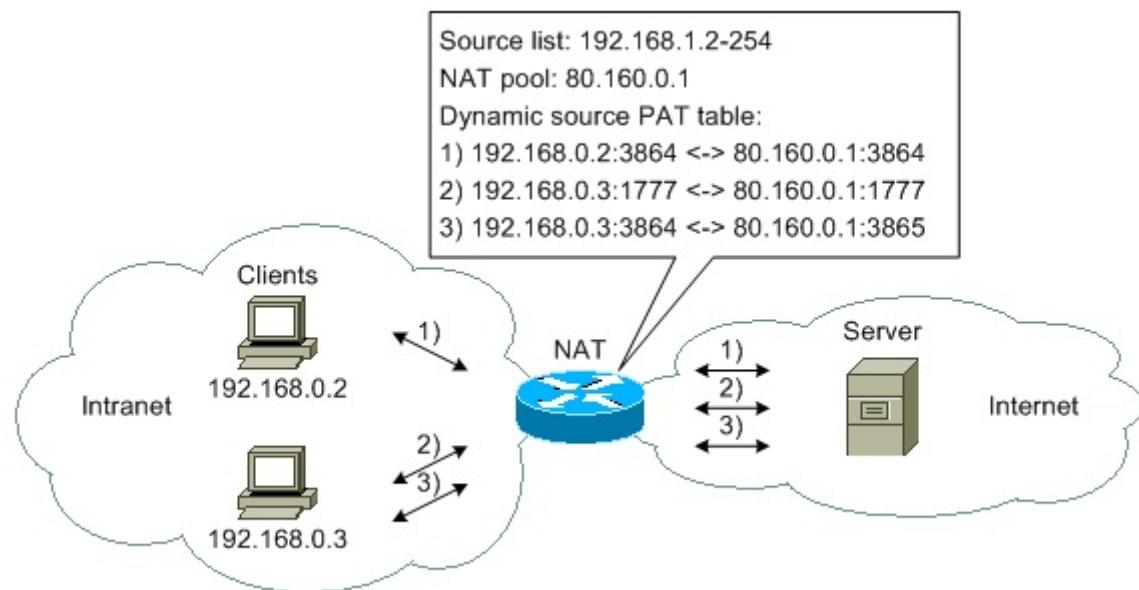
Наиболее сложной, но и наиболее полноценной формой NAT, позволяющей различать L4-соединения, является PAT (Port Address Translation) -- в дополнение к IP-адресам запоминаются номера программных портов. Согласно терминологии RFCs это называют NAPT (Network Address Port Translation), у Cisco это NAT Overload, но обычно используют аббревиатуру PAT.

PAT-дополнение совместимо и со статическим, и с динамическим вариантами NAT.

При этом, порты учитываются, но не подменяются -- за исключением конфликтных ситуаций, когда пара замененный IP-адрес источника плюс порт уже имеется в другой, еще активной, строке таблицы. Конфликт устраняется путем подмены и порта источника, например, на следующий либо случайно сгенерированный (незанятый) порт.

Поддержка PAT в отношении ICMP имеет существенные отличия, так как аналогом пары порт источника и порт назначения является лишь один идентификатор.

6.0.2.25b



6.0.2.26

Первоначальная постановка задачи предполагает, что подменяется IP-адрес источника (source NAT), но возможна подмена IP-адреса назначения (destination NAT) либо обоих адресов (twice NAT).

Первоначальная постановка задачи так же предполагает, что приватный IP-адрес замещается публичным, но, в общем случае, возможна произвольная комбинация.

Наконец, первоначальная постановка задачи предполагает, что запросы порождаются клиентами во внутренней сети, но, поскольку «правильная» NAT-таблица работает в двух направлениях (преобразуются и IP-адреса назначения в ответных пакетах), открыта возможность обслуживания запросов со стороны внешних клиентов (two-way NAT).

6.0.2.27

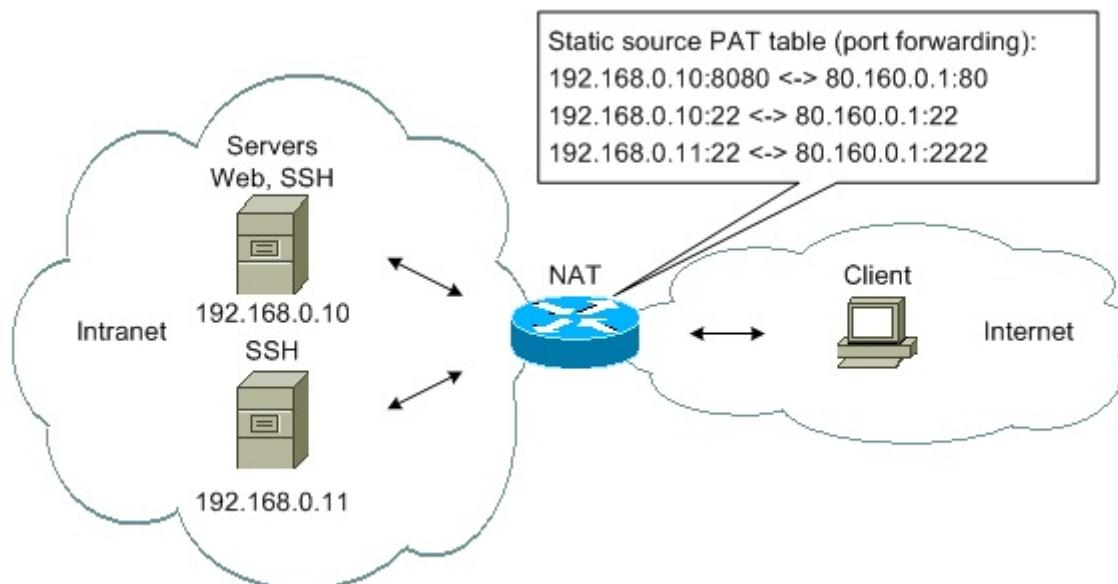
Какой вариант NAT позволит обеспечить доступ из Internet к web-серверу во внутренней сети?

Как обеспечить доступ и к web-серверу, и к маршрутизатору?

6.0.2.28

Так, статический вариант NAT позволяет разместить во внутренней сети сервер и адресовать его из Internet.

Более того, статический вариант PAT под названием port forwarding (в системах UNIX port redirecting плюс port mapping) позволяет перенаправлять запросы из Internet об определенных сервисах на соответствующие отдельные внутренние серверы.



6.0.2.29

Существуют особые реализации NAT -- NAT traversals -- наборы возможностей, позволяющие сетевым приложениям управлять NAT (определять публичный IP-адрес, назначать порты и так далее).

На NAT могут накладываться различные ограничения (restricted NAT), например, связанные с «происхождением» пакетов.

Все варианты NAT совместимы с туннелированием.

NAT полностью противоречит идеологии IPv6, поэтому, касательно IPv6, его поддержка не рекомендуется.

Экзотическим гибридом IPv6 и IPv4 (в том числе альтернативой туннелям IPv6-over-IPv4), позволяющим вместо IPv6-адресов подставлять IPv4-адреса, является NAT64. Stateless NAT64 (RFC 6145) можно рассматривать как аналог IPv4 «просто» NAT, stateful NAT64 (RFC 6146) -- как аналог IPv4 PAT.

6.0.2.30

С NAT связано еще несколько понятий, некоторые из которых можно рассматривать как компоненты NAT.

Согласно идеологии IPv4, в нормальной ситуации, в течение сеанса работы сетевой интерфейс должен иметь один IP-адрес.

Термин IP masquerading (IP-маскарад) обобщенно означает что IP-адрес сетевого интерфейса можно менять «на лету».

В Linux-трактовке, это комбинация source NAT с динамически назначаемым IP-адресом, на который выполняется замена (заранее неизвестен).

6.0.2.31

Термин IP aliasing (IP-псевдонимы) обобщенно означает что сетевой интерфейс может иметь несколько IP-адресов.

В более «приземленной» трактовке, это возможность непосредственного присвоения сетевому интерфейсу сразу нескольких IP-адресов из разных подсетей либо из одной подсети.

При IP aliasing в UNIX, в отличие от других ОС, на базе аппаратного сетевого интерфейса создают дополнительные, выраженные явно, логические интерфейсы, каждый из которых соответствует отдельной подсети.

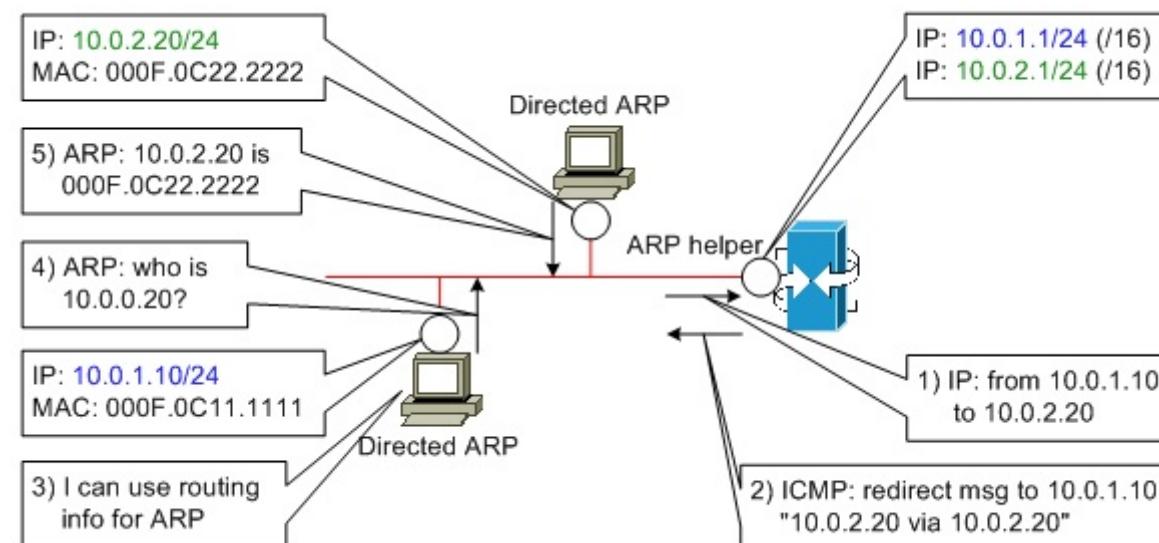
IP aliasing позволяет на один сегмент «наложить» несколько подсетей без разделения на канальном уровне, что не совсем правильно, но часто используется в различных экспериментах (таковые подсети и интерфейсы можно называть частично виртуальными). (Не путать с виланами и подинтерфейсами, которые будут рассмотрены в дальнейшем.)

Среди IP-псевдонимов выделяют главный (по аналогии с главным сетевым интерфейсом IP-шлюза). Выделение главного IP-псевдонима позволяет решить проблему выбора адреса источника, если запрос сформирован на самой станции.

6.0.2.32

Экспериментальная комбинация IP aliasing и ICMP redirects известна как directed ARP (RFC 1433). Это понятие перекликается с понятием directed broadcasts и проявляется при неправильном соотнесении подсетей и сегментов.

Directed ARP разрешает формировать ARP-запрос в отношении IP-адреса из другой подсети если обе подсети «наложены» на один сегмент -- на основании маршрутной информации от ARP helper.



6.0.2.33

10. Прочие задачи, связанные с преобразованием передаваемой информации и, как правило, не требующие обеспечения конфиденциальности.

Можно упомянуть, например, прозрачное сжатие данных, балансировку нагрузки и вполне легальное перенаправление прикладных сервисов.

6.0.3.1a

Все перечисленные выше задачи могут решаться самыми разными устройствами на различных уровнях модели OSI. Все способы можно объединить в три направления:

1. Host-based (server-based + personal) -- сугубо программные решения на базе универсальных серверных, настольных и мобильных ОС, таких как Windows, Linux, iOS и так далее. Эти ОС, в свою очередь, «круятся» на обычных серверах, ПК и смартфонах. Здесь можно сразу выделить два уровня, на которых можно решать упомянутые задачи: ядро ОС и прикладное ПО.

2. Integrated -- программно-аппаратные решения на базе специализированных сетевых ОС, таких как Cisco IOS, Juniper Junos OS, Alcatel-Lucent SR OS и так далее. Эти ОС «круются» на маршрутизаторах и другом сетевом оборудовании. Они хоть и специализированы как сетевые, но в своей области универсальны, то есть не полностью адаптированы к упомянутым задачам. Степень адаптации повышается за счет специальных аппаратных модулей.

6.0.3.1b

3. Appliance-based -- полностью специализированные аппаратные решения, такие как Cisco ASA, Fortinet FortiGate, SafeNet eSafe и так далее. Эти устройства «сосредоточены» на обеспечении безопасности и их называют *аппаратными сетевыми экранами* (security appliances).

6.0.3.2



Fortinet FortiGate 100E [Fortinet]

6.0.3.3

Аппаратные сетевые экраны, которые «заточены» для контроля доступа по протоколам прикладного уровня, известны как NACs (Network Admission Controls).

Примером может служить Cisco Ironport.

6.0.3.4



Cisco Ironport C160 [Cisco]

6.0.3.5

Новые условные графические обозначения.



-- обобщенный прокси-сервер



-- обобщенный сетевой экран



-- аппаратный сетевой экран



-- защищенный маршрутизатор

(аналогично обозначают остальные защищенные устройства)



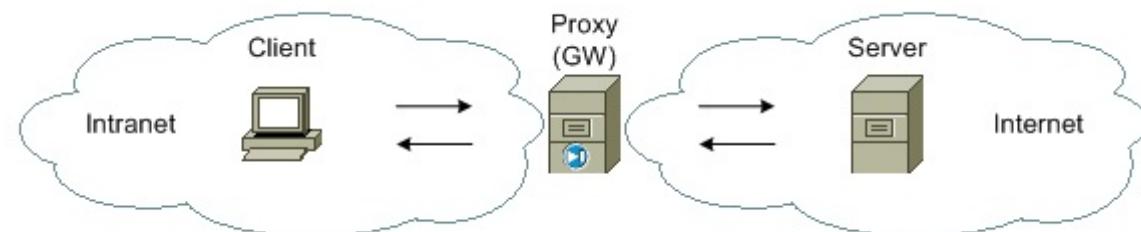
-- IPS/IDS



-- NAC

6.0.4.1

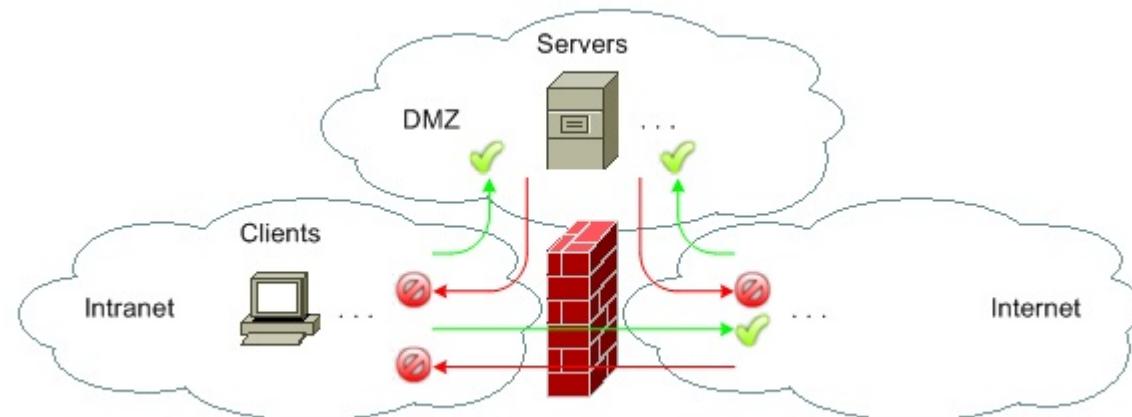
Типичное место расположения прокси -- это граница между внутренней сетью и сетью публичного доступа.



Задачи, связанные с обеспечением безопасности обычно решают во внутренней сети. Провайдеры, в идеале, обеспечивают беспрепятственное прохождение трафика.

6.0.4.2

Концепция демилитаризованной зоны -- DeMilitarization Zone (DMZ) -- предполагает размещение серверов, которые должны быть доступны публично, в отдельно выделенной области.



Зеленым цветом показаны разрешенные запросы (сетевой экран пропускает), красным -- запрещенные (сетевой экран блокирует).

Данная концепция не является догмой и может быть адаптирована к потребностям конкретного предприятия либо организации.

6.0.5.1

Как один из компонентов обеспечения безопасности Cisco предлагает комплексную технологию под названием AAA (Authentication, Authorization, and Accounting).

Под **аккаунтингом** понимают учет ресурсов, потребленных авторизированными пользователями, например, времени или объема трафика.

6.0.6.1

Две основные проблемы прокси.

При использовании следует избегать каскадирования большого числа прокси, а также перегрузки ядра ОС.

6.0.7.1

Касательно протоколов прикладного уровня, подавляющее большинство прокси -- это HTTP-прокси, причем двух вариантов:

1. Get Method.
2. Connect Method (при TLS-туннелировании).

На втором месте находятся прокси электронной почты.

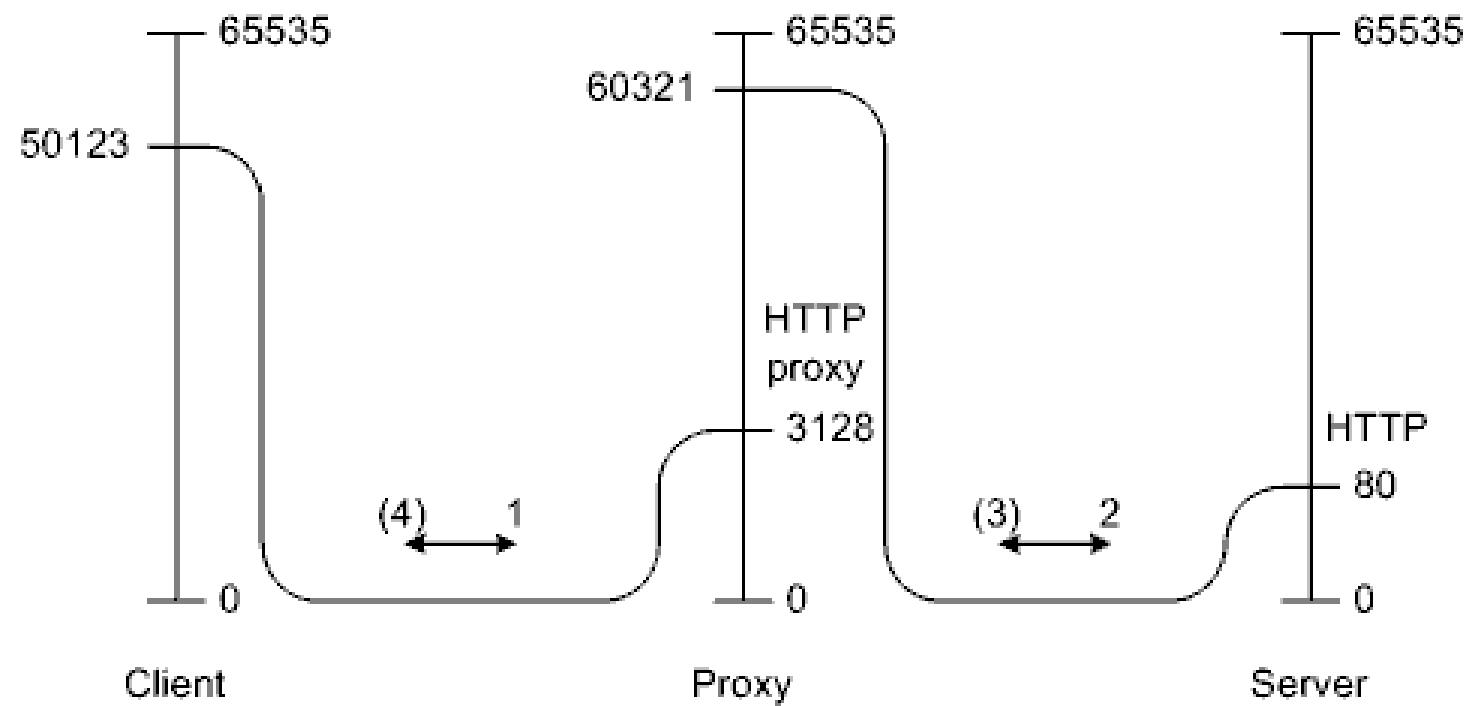
Все остальные встречаются крайне редко.

FTP-прокси, в настоящее время, обычно отдельно не выделяют.

Поддержку данного протокола включают в HTTP-прокси.

6.0.7.2

Обобщенная последовательность действий при взаимодействии клиента и сервера на примере HTTP-прокси.



6.0.7.3

Соединяемся с 192.168.251.1 (192.168.251.1:8080)
GET http://www.navitel.com.ua/files/knipex/knipex_catalogue.pdf HTTP/1.0 Accept: */* Referer: http://www.navitel.com.ua/files/knipex/ Proxy-Authorization: Basic Z2xY2V2aWNoOmJzdWly Host: www.navitel.com.ua
HTTP/1.0 200 OK Date: Fri, 11 Mar 2011 09:27:40 GMT Server: Apache/1.3.37 (Unix) mod_throttle/3.1.2 FrontPage/5.0.2.2635 mod_psoft_traffic/0.2 mod_ssl/2.8.28 OpenSSL/0.... Last-Modified: Wed, 04 Feb 2009 07:22:56 GMT ETag: "bac01d-a98695-49894250" Accept-Ranges: bytes Content-Length: 11110037 Content-Type: application/pdf X-Cache: MISS from proxy1.bsuir.by X-Cache-Lookup: MISS from proxy1.bsuir.by:8080 Via: 1.1 proxy1.bsuir.by:8080 (squid/2.7.STABLE6) Connection: close
Состояние закачки - [Закачка]
Еще одна секция запущена
Соединяемся с 192.168.251.1 (192.168.251.1:8080)
GET http://www.navitel.com.ua/files/knipex/knipex_catalogue.pdf HTTP/1.0 Accept: */* Range: bytes=5571018- Referer: http://www.navitel.com.ua/files/knipex/ Proxy-Authorization: Basic Z2xY2V2aWNoOmJzdWly Host: www.navitel.com.ua
HTTP/1.0 206 Partial Content Date: Fri, 11 Mar 2011 09:27:44 GMT Server: Apache/1.3.37 (Unix) mod_throttle/3.1.2 FrontPage/5.0.2.2635 mod_psoft_traffic/0.2 mod_ssl/2.8.28 OpenSSL/0.... Last-Modified: Wed, 04 Feb 2009 07:22:56 GMT ETag: "bac01d-a98695-49894250" Accept-Ranges: bytes Content-Length: 5539019 Content-Range: bytes 5571018-11110036/11110037 Content-Type: application/pdf X-Cache: MISS from proxy1.bsuir.by X-Cache-Lookup: HIT from proxy1.bsuir.by:8080 Via: 1.1 proxy1.bsuir.by:8080 (squid/2.7.STABLE6) Connection: close

Пример журнала многопоточного скачивания через HTTP-прокси

6.0.8.1

Для поддержки работы прокси-серверов разработано несколько вспомогательных протоколов:

ICP (Internet Cache Protocol),

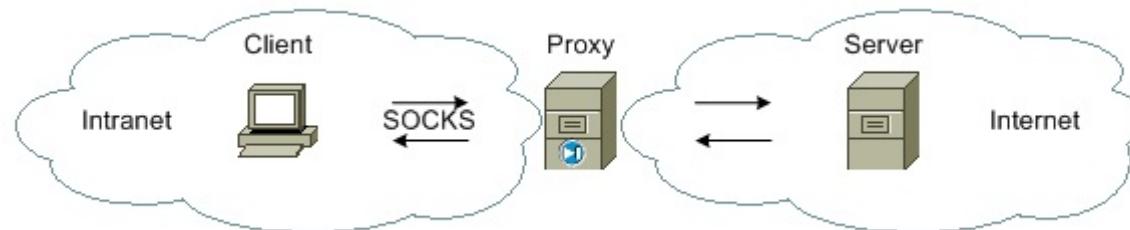
HTCP (Hyper Text Caching Protocol),

CARP (Common Address Redundancy Protocol)

и некоторые другие.

6.0.8.2

Более старой альтернативой HTTP Proxy и NAT является SOCKS (SOCKet Secure) -- протокол, использующий клиент-серверную модель.



Существуют две основные версии: SOCKSv4 (открытый стандарт от ряда разработчиков) и SOCKSv5 (RFC 1928).

В четвертой версии предусмотрены функции формирования запросов к прокси-серверу, установки TCP-соединений, передачи данных между приложениями.

В пятой версии дополнительно введена поддержка аутентификации.

6.0.9.1

Различные задачи, решаемые прокси, соответствующим наиболее уместным образом воплотились как на различных уровнях модели OSI, так и в различных категориях ПО:

1. Ядро ОС.
2. Системное ПО, включаемое в дистрибутив.
3. Прикладное ПО, включаемое в дистрибутив.
4. Стороннее (third party) ПО, не включаемое в дистрибутив.

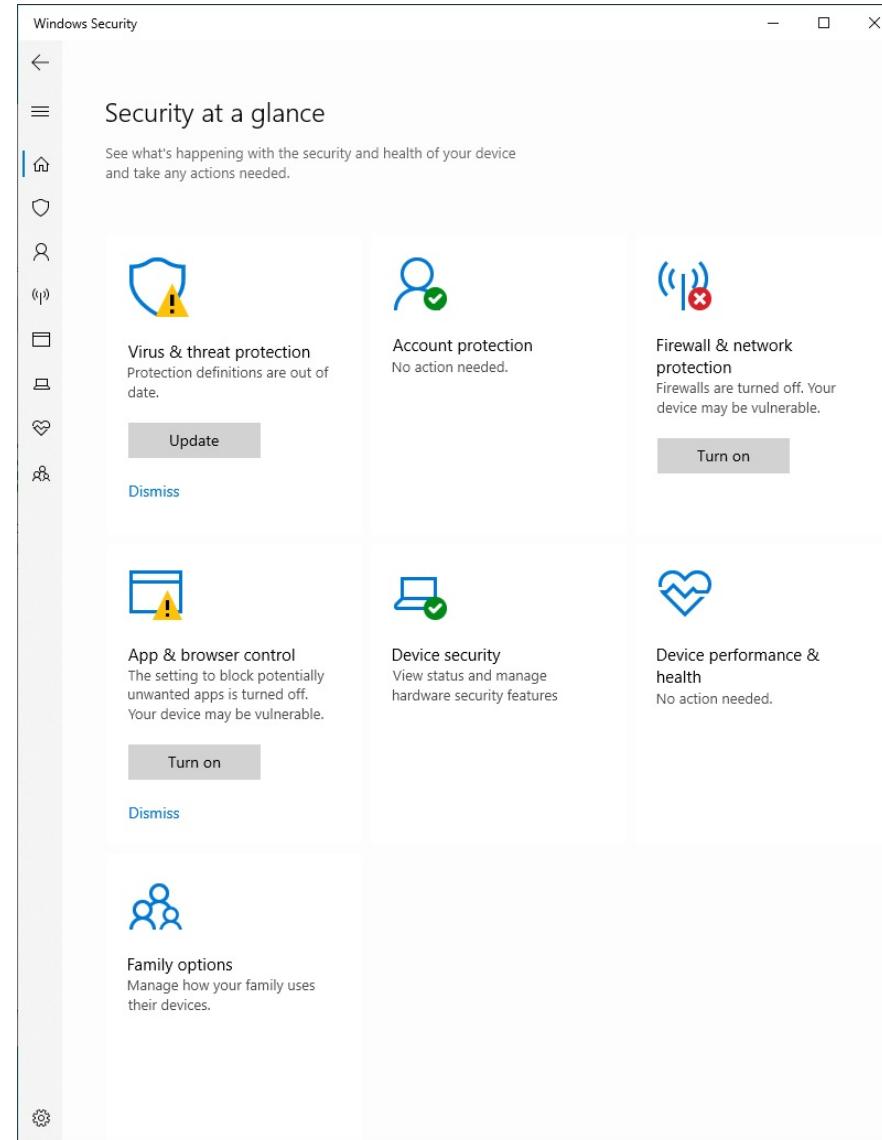
6.0.10.1

Начиная с Windows XP SP2 и Server 2003 SP1, в ядро интегрирован Windows Firewall.

Позиционируют как базовый персональный сетевой экран.

В Windows 10, 11, Server 2016/2019/2022 Windows Firewall, в связке с Windows Defender и Windows Action Center, постепенно «переносят» в отдельное системное приложение под названием Windows Security.

6.0.10.2a



Windows Security | Windows 10 21H2

6.0.10.2b

The screenshot shows the Windows Security interface under the Firewall & network protection section. On the left, a sidebar lists various security categories: Home, Virus & threat protection, Account protection, Firewall & network protection (which is selected and highlighted in blue), App & browser control, Device security, Device performance & health, and Family options. At the bottom of the sidebar is a Settings icon.

The main content area is titled '(P) Firewall & network protection' and describes who and what can access your networks. It includes a warning message: 'Windows Defender Firewall is using settings that may make your device unsafe.' Below this is a 'Restore settings' button.

The interface then details three network profiles:

- Domain network:** Firewall is on. A 'Turn off' button is present.
- Private network:** Firewall is off. A 'Turn on' button is present.
- Public network (active):** Firewall is off. A 'Turn on' button is present.

At the bottom of the main content area, there are links for further action: Allow an app through firewall, Network and Internet troubleshooter, Firewall notification settings, Advanced settings, and Restore firewall to default.

On the right side of the window, there are additional links: Windows Community videos, Learn more about Firewall & network protection, Have a question? (with a Get help link), Who's protecting me? (with a Manage providers link), Help improve Windows Security (with a Give us feedback link), Change your privacy settings, View and change privacy settings for your Windows 10 device, Privacy settings, Privacy dashboard, and Privacy Statement.

Firewall & network protection в Windows 10 21H2

6.0.10.2c

The screenshot shows the Windows Defender Firewall settings in the Control Panel. The left sidebar lists options like 'Allow an app or feature through Windows Defender Firewall', 'Change notification settings', 'Turn Windows Defender Firewall on or off', 'Restore defaults', 'Advanced settings', and 'Troubleshoot my network'. The main pane displays information about protecting the PC and allows updating firewall settings. It shows two network profiles: 'Private networks' (Not connected) and 'Guest or public networks' (Connected). The 'Windows Defender Firewall state' is set to 'Off'. Under 'Incoming connections', it says 'Block all connections to apps that are not on the list of allowed apps'. Under 'Active public networks', it lists 'Network 2'. The 'Notification state' is set to 'Do not notify me when Windows Defender Firewall blocks a new app'.

Windows Defender Firewall

Control Panel Home

Allow an app or feature through Windows Defender Firewall

Change notification settings

Turn Windows Defender Firewall on or off

Restore defaults

Advanced settings

Troubleshoot my network

Help protect your PC with Windows Defender Firewall

Windows Defender Firewall can help prevent hackers or malicious software from gaining access to your PC through the Internet or a network.

Update your Firewall settings

Windows Defender Firewall is not using the recommended settings to protect your computer.

Use recommended settings

What are the recommended settings?

Private networks Not connected

Guest or public networks Connected

Networks in public places such as airports or coffee shops

Windows Defender Firewall state: Off

Incoming connections: Block all connections to apps that are not on the list of allowed apps

Active public networks: Network 2

Notification state: Do not notify me when Windows Defender Firewall blocks a new app

See also

Security and Maintenance

Network and Sharing Center

Windows Defender Firewall в Windows 10 (11, Server 2016/2019/2022) (то же что Windows Firewall в предыдущих версиях)

6.0.10.2d

The screenshot shows the Windows Defender Firewall with Advanced Security interface. The left sidebar has navigation links: Inbound Rules, Outbound Rules, Connection Security Rules, and Monitoring. The main area displays the 'Inbound Rules' table with the following columns: Name, Group, Profile, Enabled, Action, Override, Program, Local Address, Remote Address, Protocol, Local Port, Remote Port, Authorized Users, and Authz. The table lists numerous rules, mostly from the 'Core Networking' group, including entries for Firefox, Microsoft Lync, and various Windows services like Edge Traversal and ICMPv6. The right sidebar contains an 'Actions' section with options like New Rule..., Filter by Profile, Refresh, and Export List... .

Name	Group	Profile	Enabled	Action	Override	Program	Local Address	Remote Address	Protocol	Local Port	Remote Port	Authorized Users	Authz
Firefox (C:\Program Files\Mozilla Firefox)	Private	Yes	Allow	No	C:\Prog...	Any	Any	Any	UDP	Any	Any	Any	Any
Firefox (C:\Program Files\Mozilla Firefox)	Private	Yes	Allow	No	C:\Prog...	Any	Any	Any	TCP	Any	Any	Any	Any
Microsoft Lync	Public	Yes	Allow	No	C:\Prog...	Any	Any	Any	UDP	Any	Any	Any	Any
Microsoft Lync UcMapi	Public	Yes	Allow	No	C:\Prog...	Any	Any	Any	UDP	Any	Any	Any	Any
@FirewallAPI.dll,-80201	@FirewallAPI.dll,-80200	All	Yes	Allow	No	%System...	Any	Local subnet	TCP	554, 8554...	Any	Any	Any
@FirewallAPI.dll,-80206	@FirewallAPI.dll,-80200	All	Yes	Allow	No	%System...	Any	Local subnet	UDP	5000-5020	Any	Any	Any
AllJoyn Router (TCP-In)	AllJoyn Router	Domai...	Yes	Allow	No	%System...	Any	Any	TCP	9955	Any	Any	Any
AllJoyn Router (UDP-In)	AllJoyn Router	Domai...	Yes	Allow	No	%System...	Any	Any	UDP	Any	Any	Any	Any
BranchCache Content Retrieval (HTTP-In)	BranchCache - Content Retr...	All	No	Allow	No	SYSTEM	Any	Any	TCP	80	Any	Any	Any
BranchCache Hosted Cache Server (HTTP...	BranchCache - Hosted Cach...	All	No	Allow	No	SYSTEM	Any	Any	TCP	80, 443	Any	Any	Any
BranchCache Peer Discovery (WSD-In)	BranchCache - Peer Discov...	All	No	Allow	No	%System...	Any	Local subnet	UDP	3702	Any	Any	Any
Cast to Device functionality (qWave-TCP...	Cast to Device functionality	Private..	Yes	Allow	No	%System...	Any	PlayTo Renderers	TCP	2177	Any	Any	Any
Cast to Device functionality (qWave-UDP...	Cast to Device functionality	Private..	Yes	Allow	No	%System...	Any	PlayTo Renderers	UDP	2177	Any	Any	Any
Cast to Device SSDP Discovery (UDP-In)	Cast to Device functionality	Public	Yes	Allow	No	%System...	Any	Any	UDP	PlayTo Dis...	Any	Any	Any
Cast to Device streaming server (HTTP-St...	Cast to Device functionality	Private	Yes	Allow	No	System	Any	Local subnet	TCP	10246	Any	Any	Any
Cast to Device streaming server (HTTP-St...	Cast to Device functionality	Domain	Yes	Allow	No	System	Any	Any	TCP	10246	Any	Any	Any
Cast to Device streaming server (HTTP-St...	Cast to Device functionality	Public	Yes	Allow	No	System	Any	PlayTo Renderers	TCP	10246	Any	Any	Any
Cast to Device streaming server (RTCP-Str...	Cast to Device functionality	Public	Yes	Allow	No	%System...	Any	PlayTo Renderers	UDP	Any	Any	Any	Any
Cast to Device streaming server (RTCP-Str...	Cast to Device functionality	Domain	Yes	Allow	No	%System...	Any	Any	UDP	Any	Any	Any	Any
Cast to Device streaming server (RTCP-Str...	Cast to Device functionality	Private	Yes	Allow	No	%System...	Any	Local subnet	UDP	Any	Any	Any	Any
Cast to Device streaming server (RTSP-Str...	Cast to Device functionality	Private	Yes	Allow	No	%System...	Any	Local subnet	TCP	23554, 235...	Any	Any	Any
Cast to Device streaming server (RTSP-Str...	Cast to Device functionality	Domain	Yes	Allow	No	%System...	Any	Any	TCP	23554, 235...	Any	Any	Any
Cast to Device streaming server (RTSP-Str...	Cast to Device functionality	Public	Yes	Allow	No	%System...	Any	PlayTo Renderers	TCP	23554, 235...	Any	Any	Any
Cast to Device UPnP Events (TCP-In)	Cast to Device functionality	Public	Yes	Allow	No	System	Any	PlayTo Renderers	TCP	2869	Any	Any	Any
Citrix Provisioning Services (UDP-In)	Citrix PVS Rule Group	All	Yes	Allow	No	System	Any	Any	UDP	6902	Any	Any	Any
Connected Devices Platform - Wi-Fi Dire...	Connected Devices Platform	Public	Yes	Allow	No	%System...	Any	Any	TCP	Any	Any	Any	Any
Connected Devices Platform (TCP-In)	Connected Devices Platform	Domai...	Yes	Allow	No	%System...	Any	Any	TCP	Any	Any	Any	Any
Connected Devices Platform (UDP-In)	Connected Devices Platform	Domai...	Yes	Allow	No	%System...	Any	Any	UDP	Any	Any	Any	Any
Core Networking - Destination Unreacha...	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any	Any	Any
Core Networking - Destination Unreacha...	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv4	Any	Any	Any	Any
Core Networking - Dynamic Host Config...	Core Networking	All	Yes	Allow	No	%System...	Any	Any	UDP	68	67	Any	Any
Core Networking - Dynamic Host Config...	Core Networking	All	Yes	Allow	No	%System...	Any	Any	UDP	546	547	Any	Any
Core Networking - Internet Group Mana...	Core Networking	All	Yes	Allow	No	System	Any	Any	IGMP	Any	Any	Any	Any
Core Networking - HTTPS (TCP-In)	Core Networking	All	Yes	Allow	No	System	Any	Any	TCP	IPHTTPS	Any	Any	Any
Core Networking - IPv6 (IPv6-In)	Core Networking	All	Yes	Allow	No	System	Any	Any	IPv6	Any	Any	Any	Any
Core Networking - Multicast Listener Do...	Core Networking	All	Yes	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any	Any	Any
Core Networking - Multicast Listener Qu...	Core Networking	All	Yes	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any	Any	Any
Core Networking - Multicast Listener Rep...	Core Networking	All	Yes	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any	Any	Any
Core Networking - Multicast Listener Rep...	Core Networking	All	Yes	Allow	No	System	Any	Local subnet	ICMPv6	Any	Any	Any	Any
Core Networking - Neighbor Discovery A...	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any	Any	Any
Core Networking - Neighbor Discovery S...	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any	Any	Any
Core Networking - Packet Too Big (ICMP...	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any	Any	Any
Core Networking - Parameter Problem (I...	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any	Any	Any
Core Networking - Router Advertisement...	Core Networking	All	Yes	Allow	No	System	Any	fe80::/64	ICMPv6	Any	Any	Any	Any
Core Networking - Router Solicitation (I...	Core Networking	All	Yes	Allow	No	System	Any	Any	ICMPv6	Any	Any	Any	Any
Core Networking - Teredo (UDP-In)	Core Networking	All	Yes	Allow	No	%System...	Any	Any	UDP	Edge Trave...	Any	Any	Any

Windows Defender Firewall Advanced Settings b Windows 10 21H2

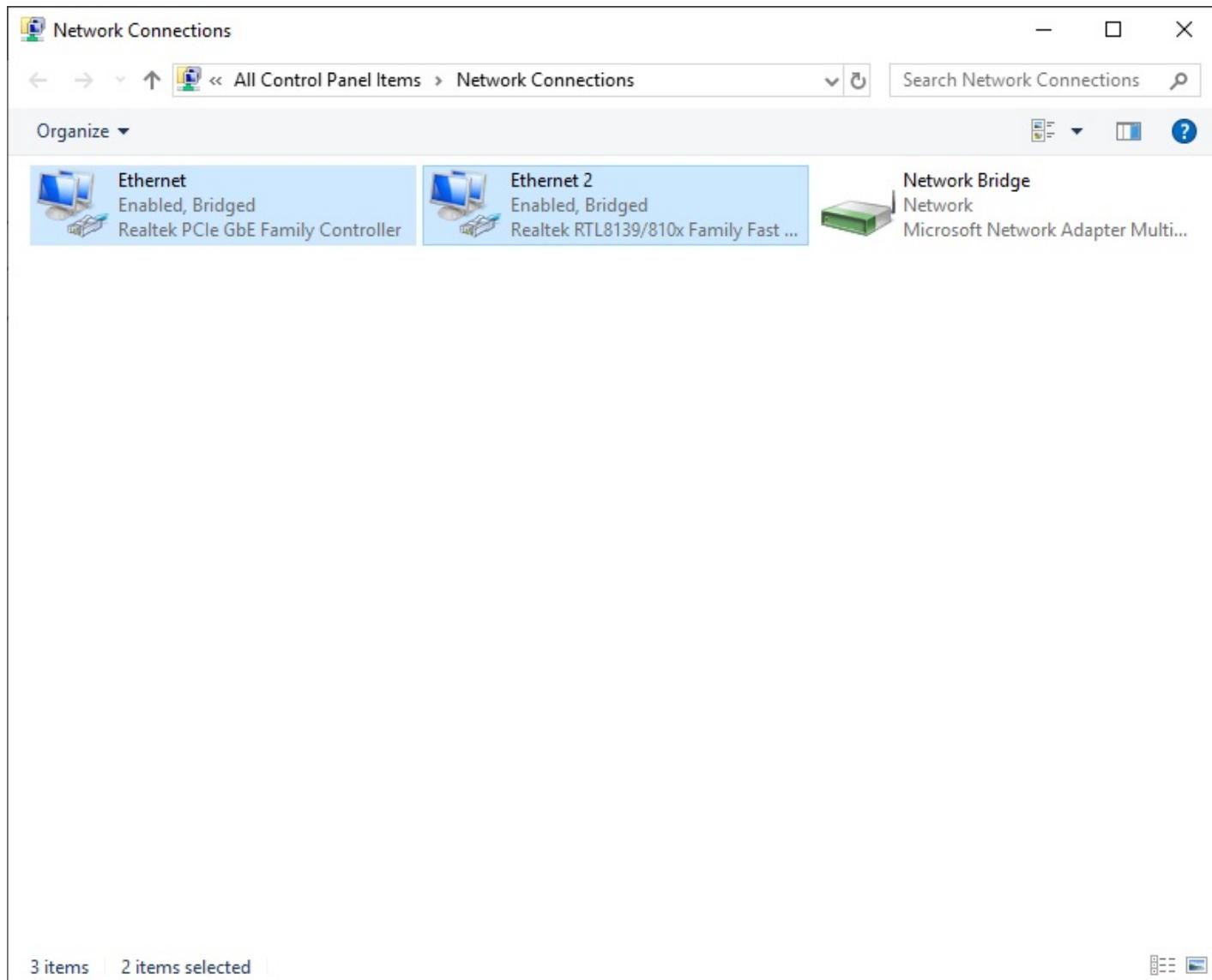
6.0.10.3

Упрощенными для удобства пользователя специфическими формами прокси являются Network Bridge и Internet Connection Sharing («вытесняют» сервис Routing and Remote Access).

Network Bridge позволяет объединить два возможно разнородных сегмента с целью эмуляции одного сегмента.

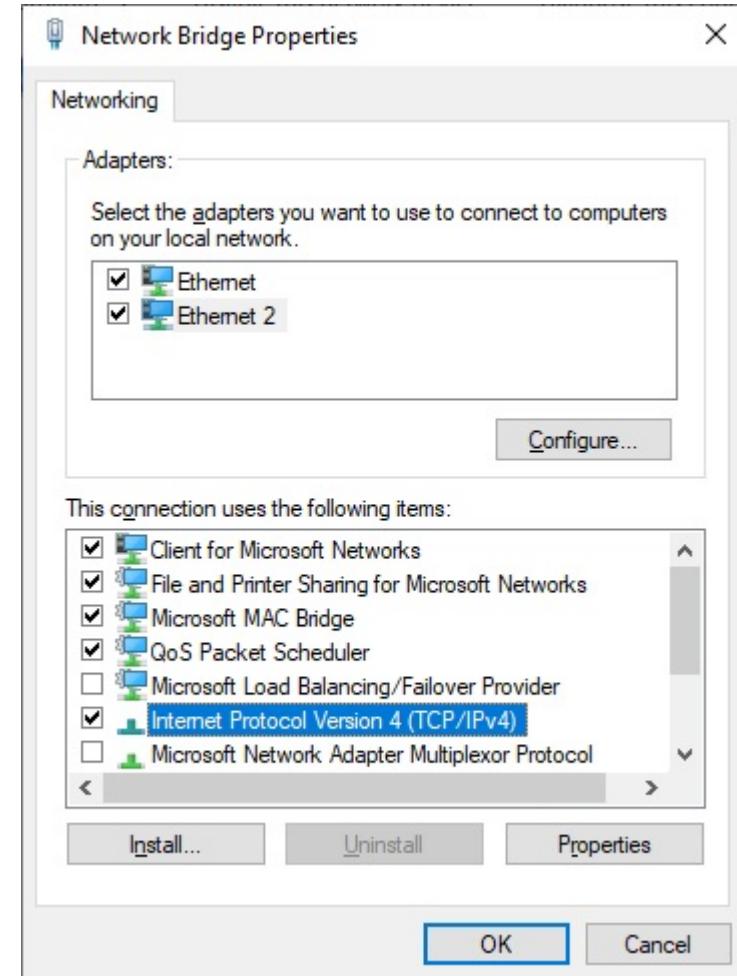
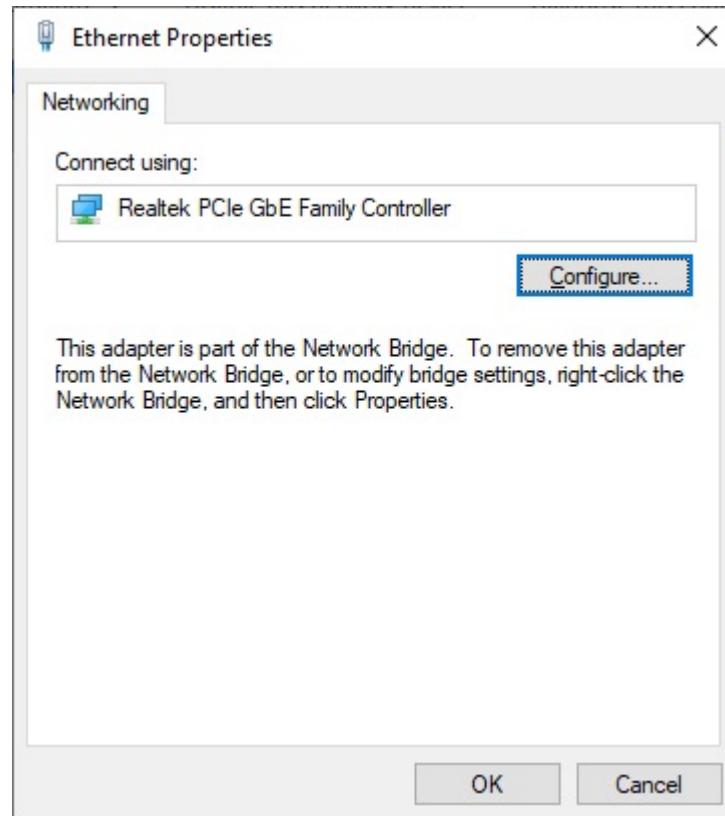
Internet Connection Sharing позволяет нескольким пользователям из одной подсети совместно использовать (в режиме разделения времени) один сетевой интерфейс из другой подсети, обычно с целью доступа в Internet.

6.0.10.4a



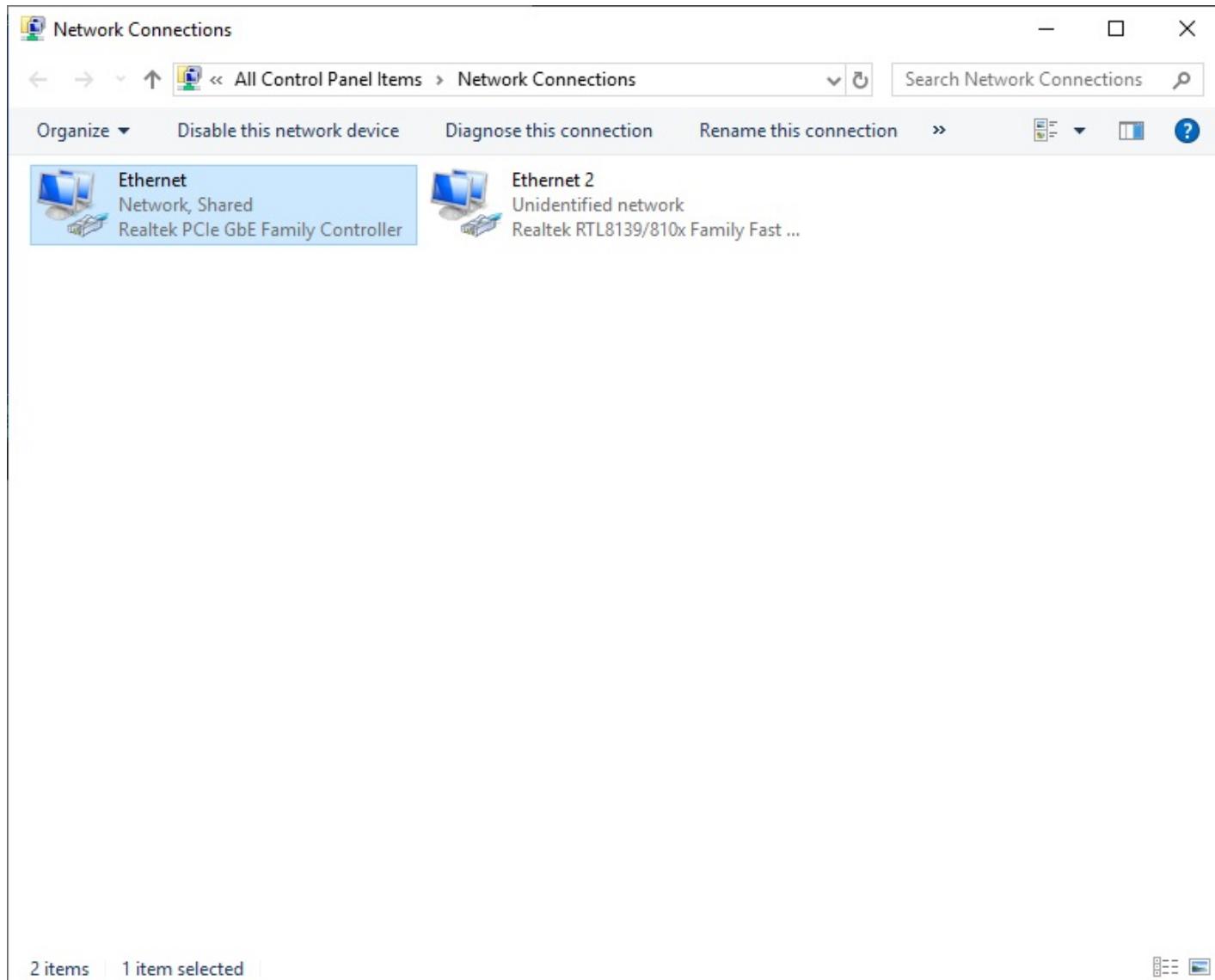
Пример Network Bridge в Windows 10 (11, Server 2016/2019/2022)

6.0.10.4b



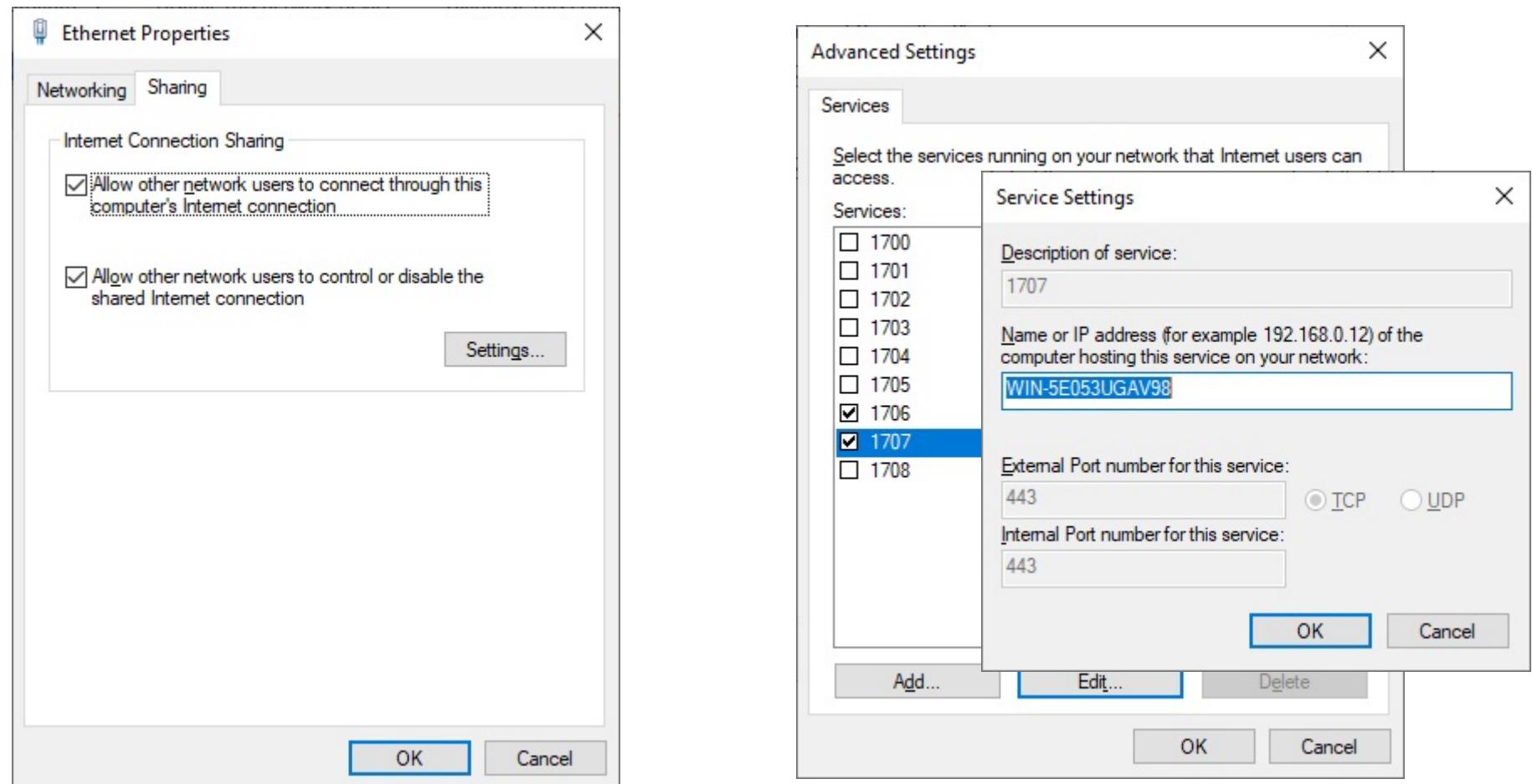
Пример Network Bridge в Windows 10 (11, Server 2016/2019/2022)

6.0.10.5a



Пример Internet Connection Sharing в Windows 10 (11, Server 2016/2019/2022)

6.0.10.5b



Пример Internet Connection Sharing в Windows 10 (11, Server 2016/2019/2022)

6.0.10.6

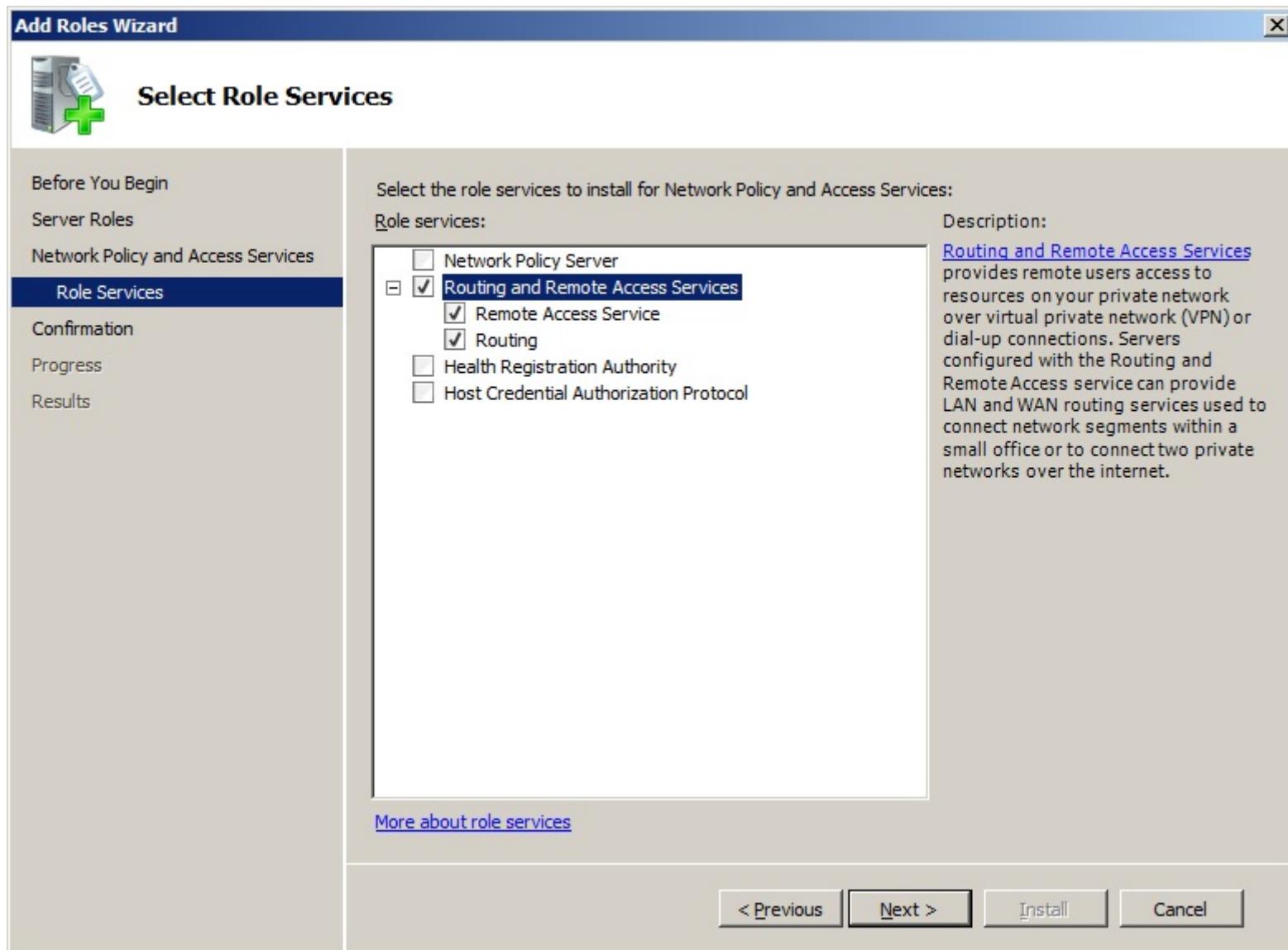
Полноценная поддержка NAT с графическим интерфейсом доступна в серверных редакциях -- в составе Routing and Remote Access.

В Server 2003 R2 был интегрированный компонент Routing and Remote Access, в Server 2008 R2 -- опциональная роль с тем же названием.

Начиная с Server 2012, роль имеет название Remote Access.

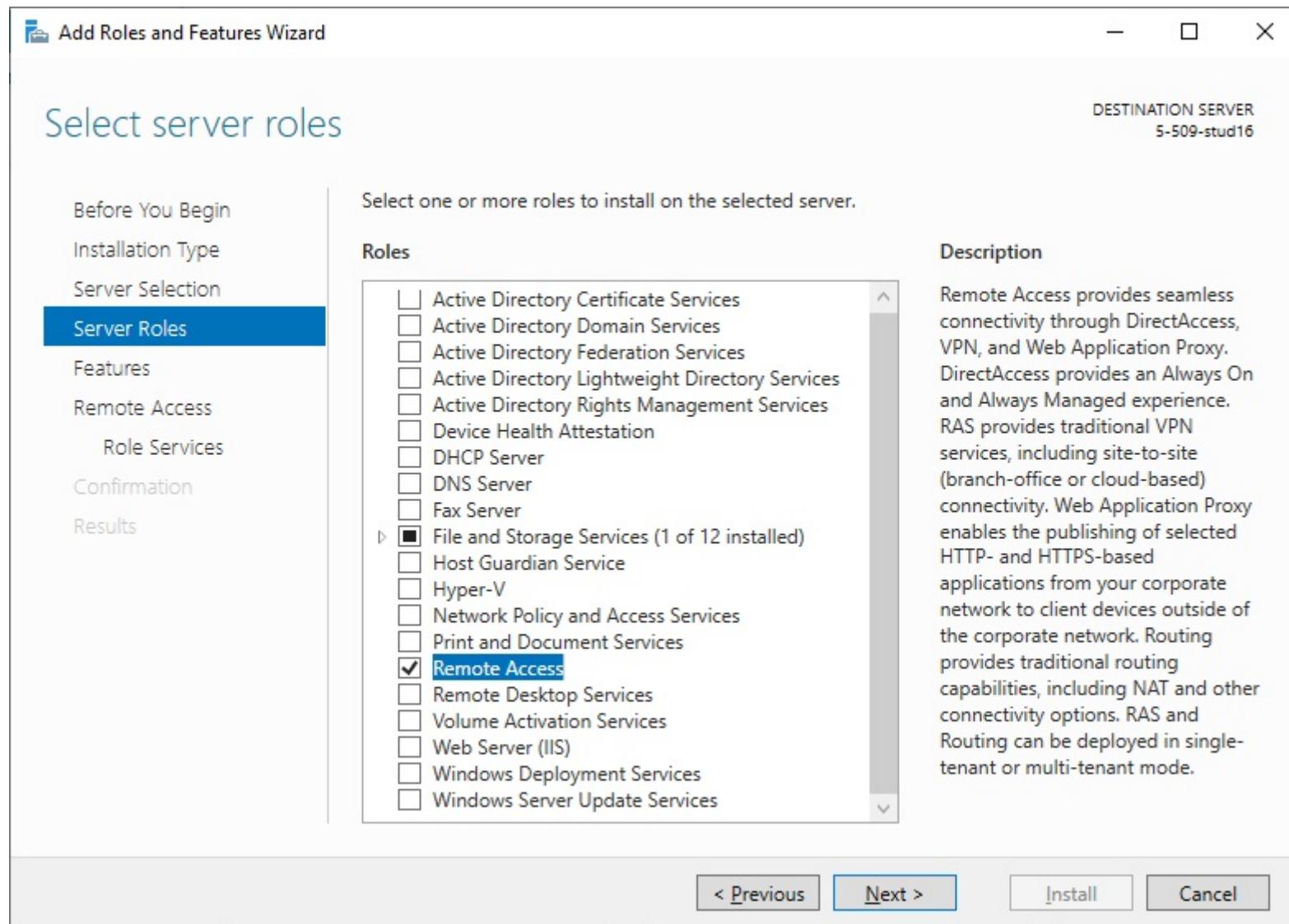
Для конфигурирования используют оснастку Routing and Remote Access (`rrasmgmt.msc`).

6.0.10.7a



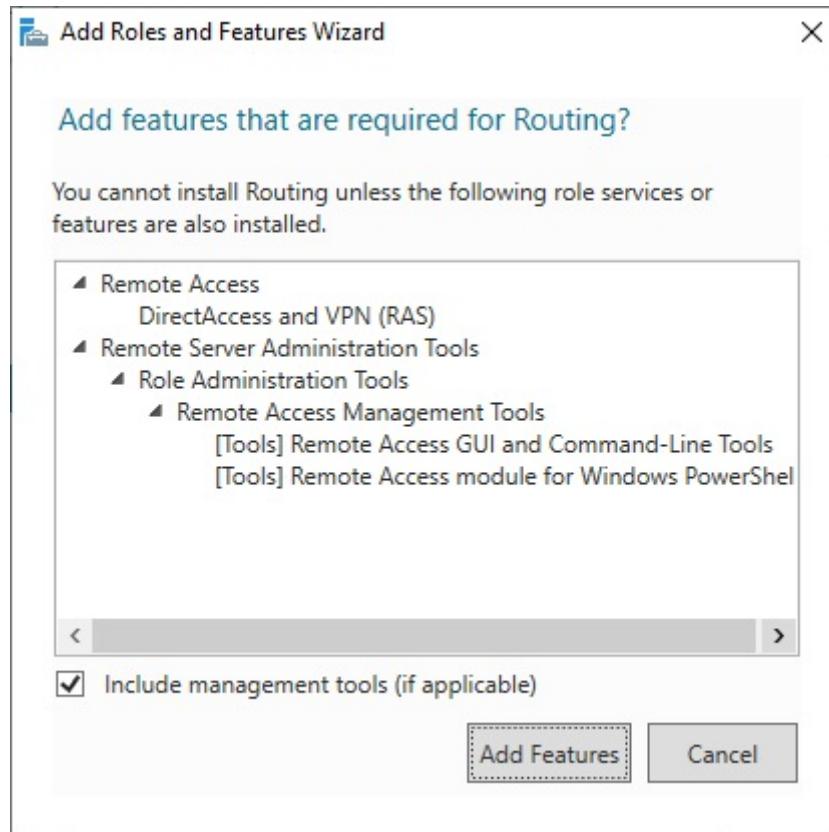
Установка роли RRAS в Windows Server 2008 R2

6.0.10.7b



Установка сервиса удаленного доступа в Windows Server 2016/2019/2022

6.0.10.7c



Установка сервиса удаленного доступа в Windows Server 2016/2019/2022

6.0.10.7d

Routing and Remote Access Server Setup Wizard

Configuration

You can enable any of the following combinations of services, or you can customize this server.

Remote access (dial-up or VPN)
Allow remote clients to connect to this server through either a dial-up connection or a secure virtual private network (VPN) Internet connection.

Network address translation (NAT)
Allow internal clients to connect to the Internet using one public IP address.

Virtual private network (VPN) access and NAT
Allow remote clients to connect to this server through the Internet and local clients to connect to the Internet using a single public IP address.

Secure connection between two private networks
Connect this network to a remote network, such as a branch office.

Custom configuration
Select any combination of the features available in Routing and Remote Access.

< Back **Next >** Cancel

Routing and Remote Access Server Setup Wizard

Custom Configuration

When this wizard closes, you can configure the selected services in the Routing and Remote Access console.

Select the services that you want to enable on this server.

VPN access

Dial-up access

Demand-dial connections (used for branch office routing)

NAT

LAN routing

< Back **Next >** Cancel

Пример шагов мастера конфигурирования сервиса удаленного доступа в Windows Server 2016/2019/2022

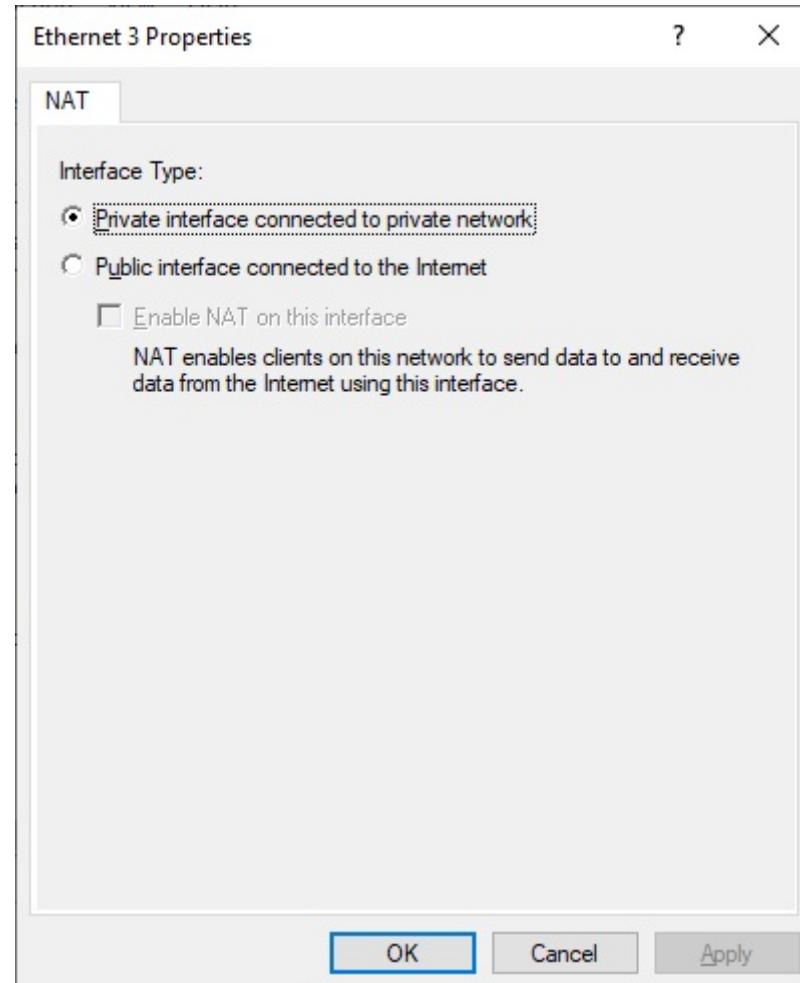
6.0.10.7e

The screenshot shows the Windows Server 2016/2019/2022 Routing and Remote Access Management Console. The left pane displays a tree view of network configurations under '5-509-STUD16 (local)'. Under 'IPv4', the 'NAT' node is selected. The right pane shows a table titled 'NAT' with the following data:

Interface	Total mappings	Inbound packets translated	Inbound packets rejected	Outbound packets translated	Outbound packets rejected
Ethernet 3	0	0	0	0	0
Ethernet 2	5	3 148	0	1 802	0

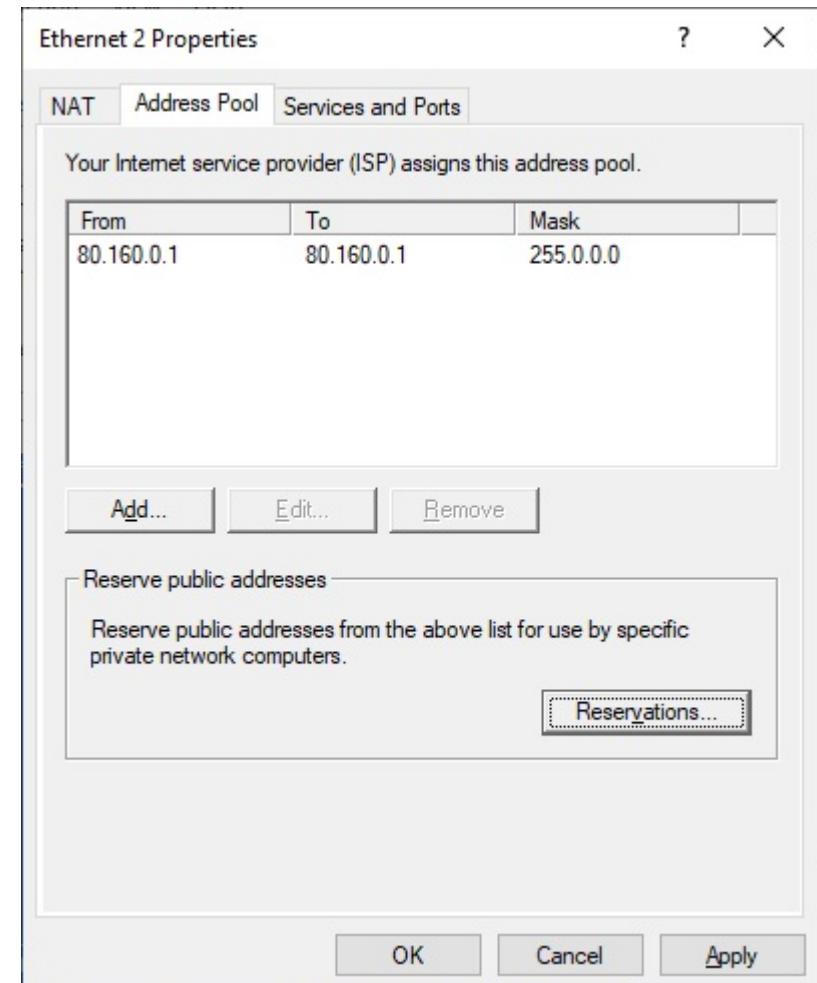
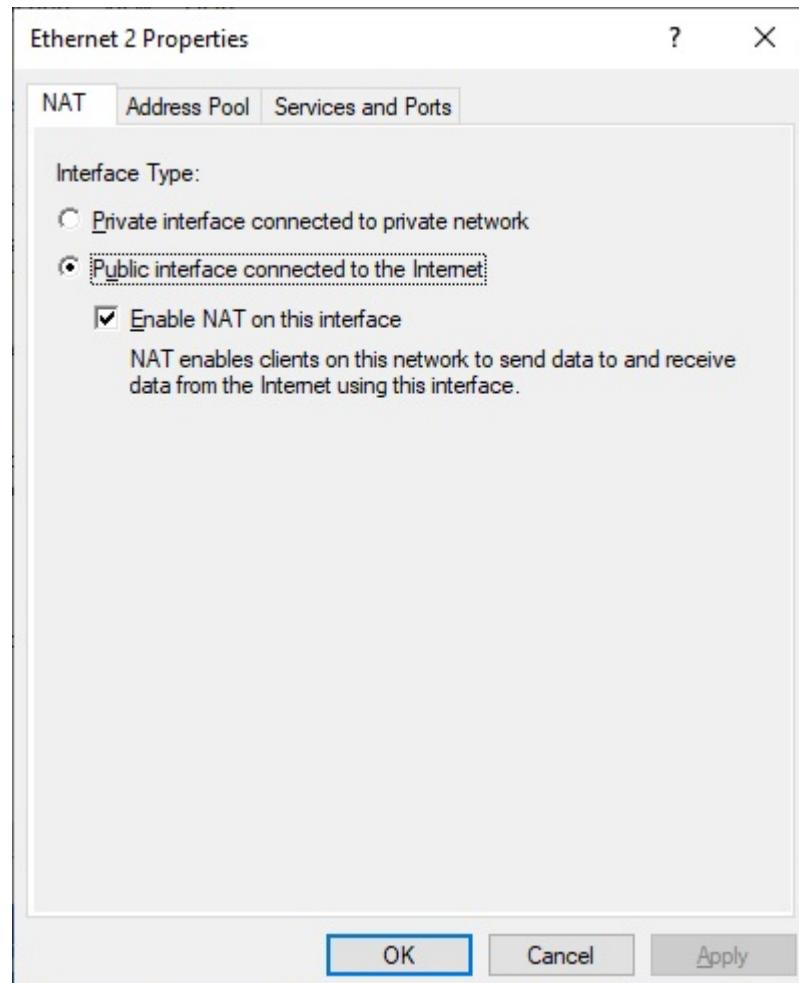
Пример NAT в Windows Server 2016/2019/2022

6.0.10.7f



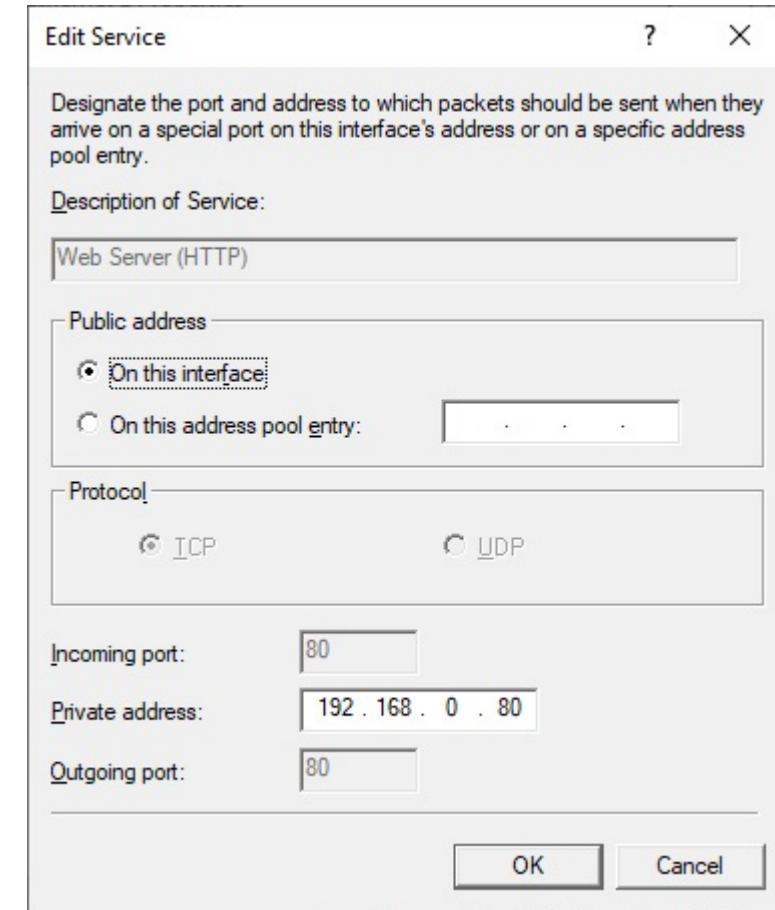
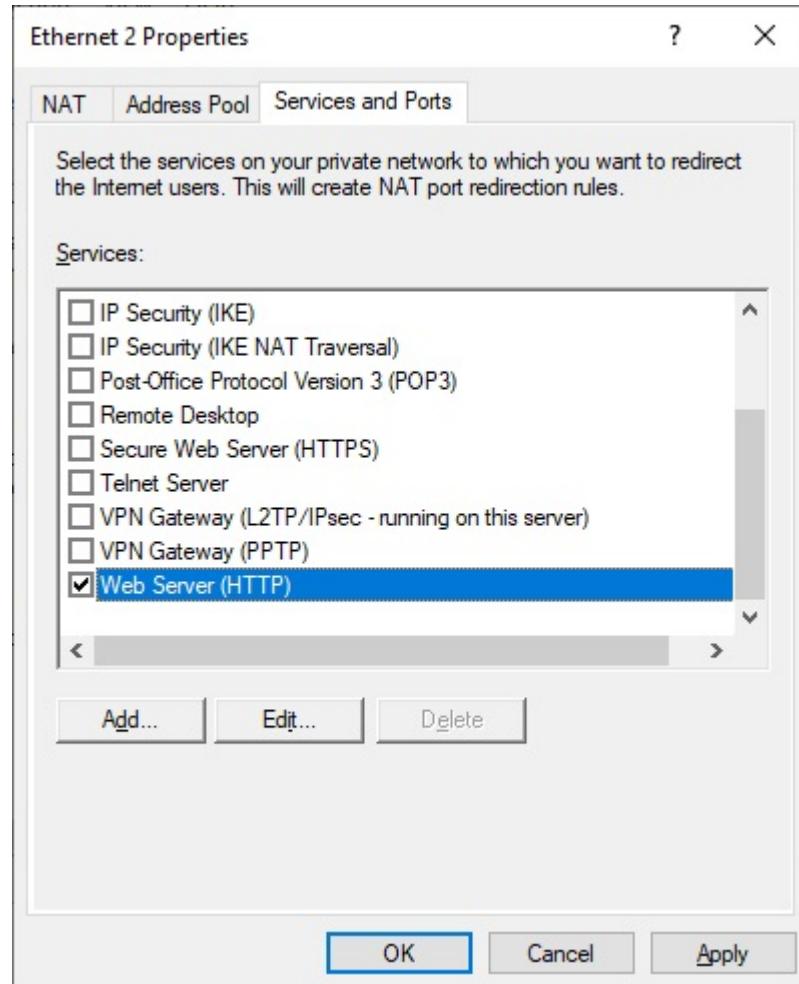
Пример NAT в Windows Server 2016/2019/2022

6.0.10.7g



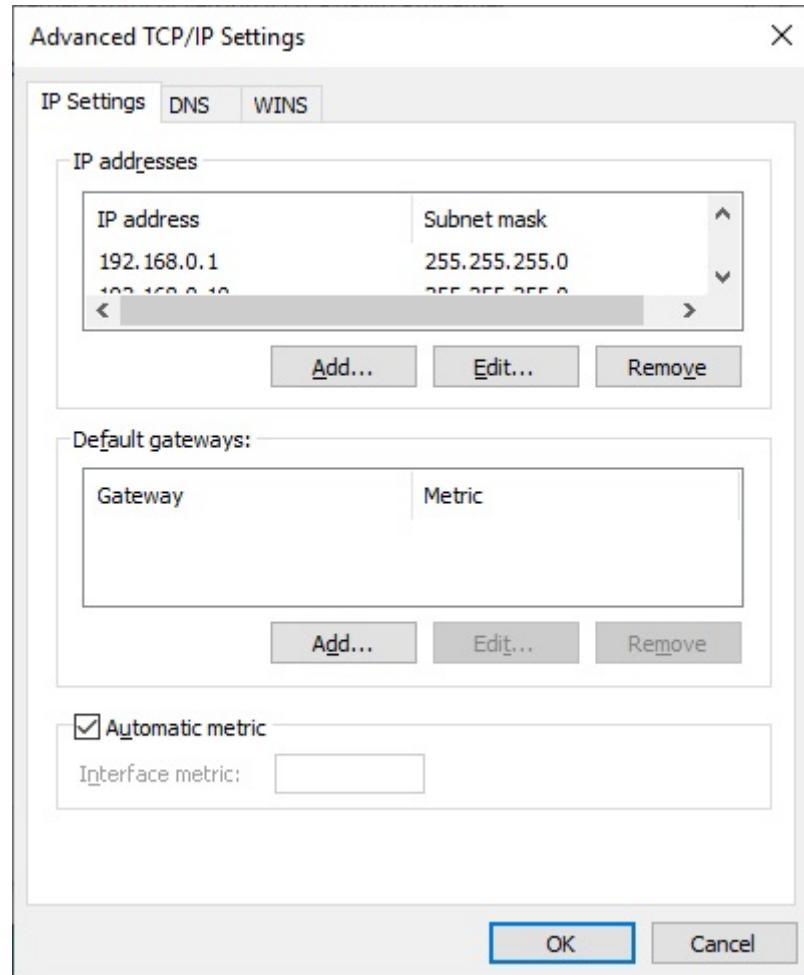
Пример NAT в Windows Server 2016/2019/2022

6.0.10.7h



Пример NAT в Windows Server 2016/2019/2022

6.0.10.8



Network Connection Details:	
Property	Value
Connection-specific DN...	
Description	Intel(R) PRO/1000 GT Desktop Adapter
Physical Address	00-1B-21-5E-E9-A9
DHCP Enabled	No
IPv4 Address	192.168.0.1
IPv4 Subnet Mask	255.255.255.0
IPv4 Address	192.168.0.10
IPv4 Subnet Mask	255.255.255.0
IPv4 Address	192.168.2.1
IPv4 Subnet Mask	255.255.255.0
IPv4 Default Gateway	
IPv4 DNS Server	
IPv4 WINS Server	
NetBIOS over Tcpip En...	Yes
Link-local IPv6 Address	fe80::c953:a642:d49f:56ca%7
IPv6 Default Gateway	

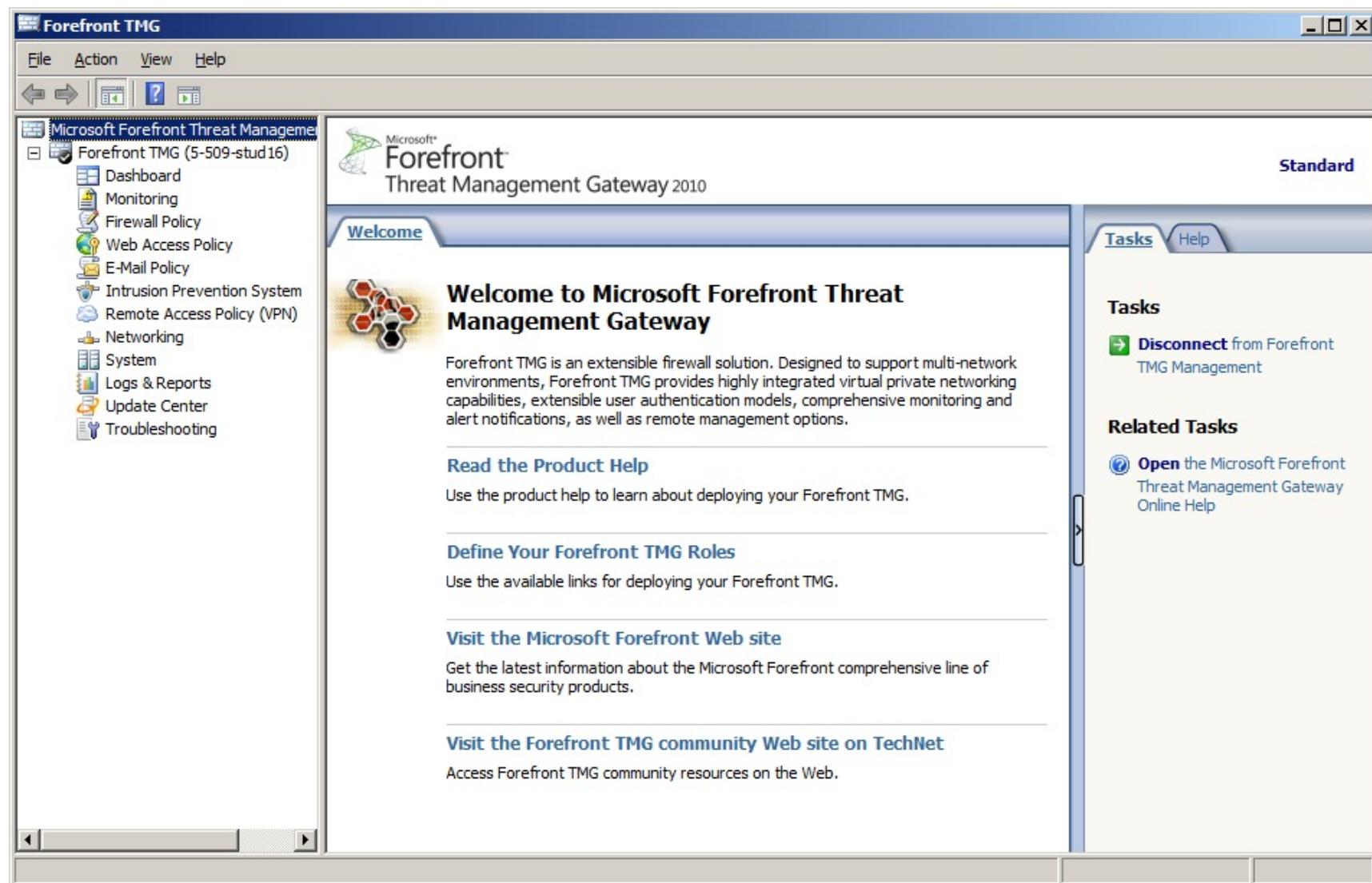
Buttons: Close

Пример IP aliasing в Windows

6.0.11.1

В случае масштабного применения для серверов Windows предлагали стандартный пакет ISA (Internet Security and Acceleration) Server, который позже назвали Forefront Threat Management Gateway. Позиционировали как кэширующий прокси-сервер с возможностями сетевого экрана. Но в 2012 г. разработка прекращена.

6.0.11.2



Microsoft Forefront TMG 2010

6.0.11.3

Кроме того, очень широко применяют пакеты сторонних производителей, среди которых следует выделить Qbik WinGate и Kerio (GFI Software) KerioControl.

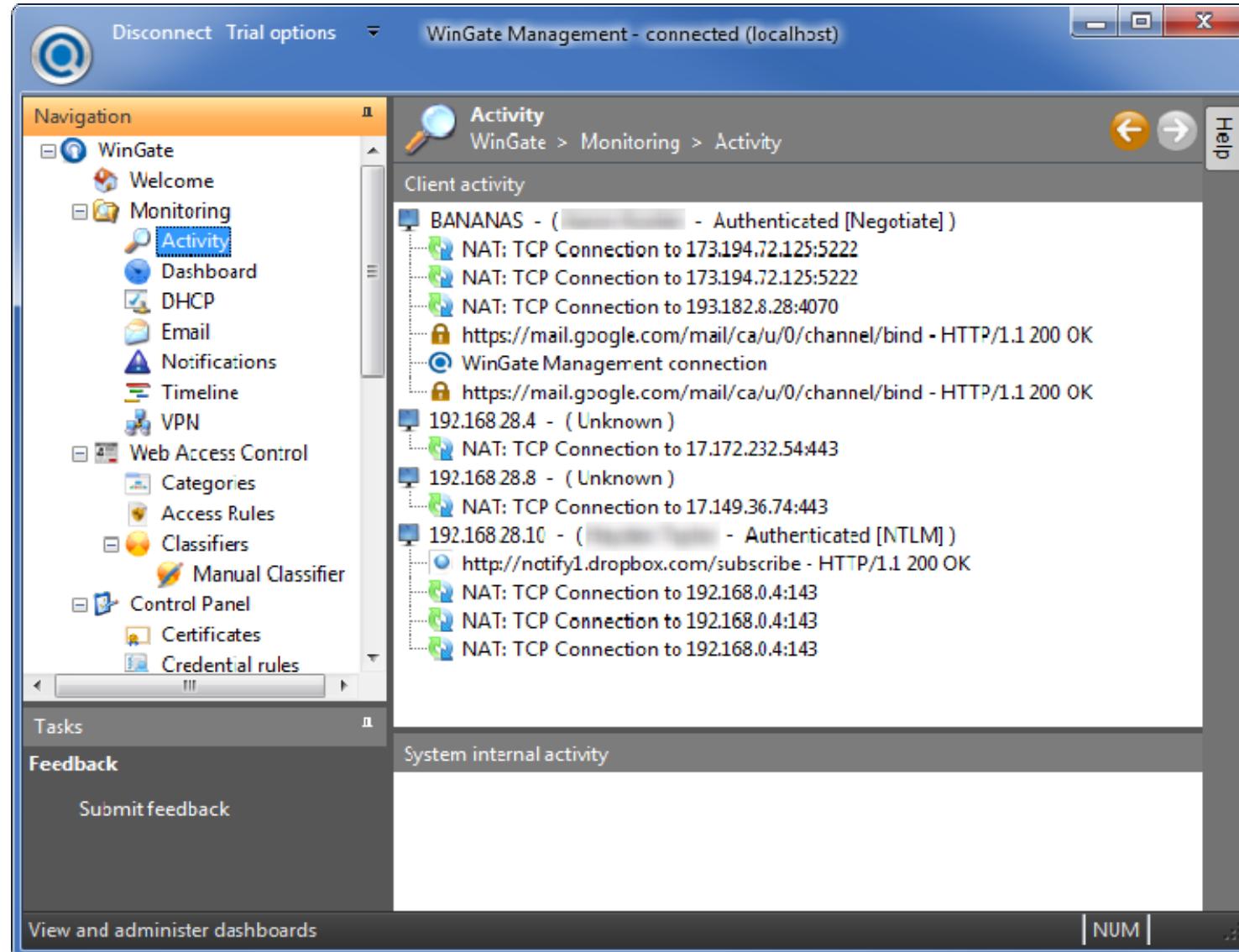
6.0.11.4a

The screenshot shows a web browser window with the following details:

- Title Bar:** File, Edit, View, History, Bookmarks, Tools, Help. The active tab is "WinGate Proxy Server. Web Proxy".
- Address Bar:** https://www.wingate.com
- Page Content:**
 - Header:** QBIK, CREATORS OF WINGATE, LOG IN | Register, Shopping Cart.
 - Main Banner:** WinGate 9, download your free trial now! (with screenshots of the software interface).
 - About us:** Since 1995 Qbik has developed capable & user friendly software specialising in Internet connectivity and security. Our products allow users to manage their Internet connections (WinGate), connect remote offices together (WinGate VPN), and combat malware (Kaspersky AV for WinGate).
 - Free 30 day trials:** All our products are available for a free 30 day trial, and are unlocked with an encoded key. Whether you are purchasing, upgrading, or wish to trial our software, the downloaded software is the same.
 - Unlimited support:** We provide unlimited free support to help you get the best out of our products, whether you are evaluating or have purchased our software.
 - WinGate:** Our flagship advanced caching HTTP proxy, SOCKS server and multi-protocol proxy server, email server and internet gateway system for Windows.
 - Intercept, cache and scan web content.
 - Scan https traffic also, with https inspection.
 - Restrict and log user web access.
 - Solve headaches with legacy TLS versions
 - And a whole lot more....
 - New version: 9.4.1 (1 February 2020) FREE for 10 users**
 - Kaspersky AntiVirus FOR WINGATE:** Scan your WinGate traffic for malware using Kaspersky Labs highly respected scanning engine. Also scans web browsing for exploits.
 - Lumen:** Scan your WinGate traffic for malware using Kaspersky Labs highly respected scanning engine. Also scans web browsing for exploits.

Qbik WinGate

6.0.11.4b



WinGate Management [Qbik]

6.0.11.5

The screenshot shows a web browser window displaying the GFI Software KerioControl product page. The URL in the address bar is <https://www.gfi.com/products-and-solutions/network-security-solutions/kerio-control>. The page features a dark blue header with the GFI Software logo and navigation links for Products, Company, Partners, Support, Login, Free Trials, and English language selection. Below the header, the breadcrumb navigation shows Home / Products and solutions / KerioControl. A star rating of 4.4 / 5 (82 Reviews) is displayed. The main content area includes the KerioControl logo, a brief description of it as a next-generation firewall for small and medium-sized businesses, and a 'Try KerioControl for free' button. To the right, there is a video player showing a thumbnail for 'Kerio Control - Essential security for S...'. Below the video player, there is a 'Watch on YouTube' button. At the bottom of the main content area, there is a navigation bar with tabs for KerioControl (selected), Overview, Features, Pricing, Resources, and Support, along with a 'Buy now' button. A note about the Kerio VPN Client is also present.

Detect threats, block viruses, control traffic and more

KerioControl is a next-generation firewall and unified threat management product for small and medium-sized businesses (SMBs) that are looking for a comprehensive solution for their security needs. With KerioControl, businesses gain:



A firewall that connects



Intrusion protection



Web content and



Virtual private

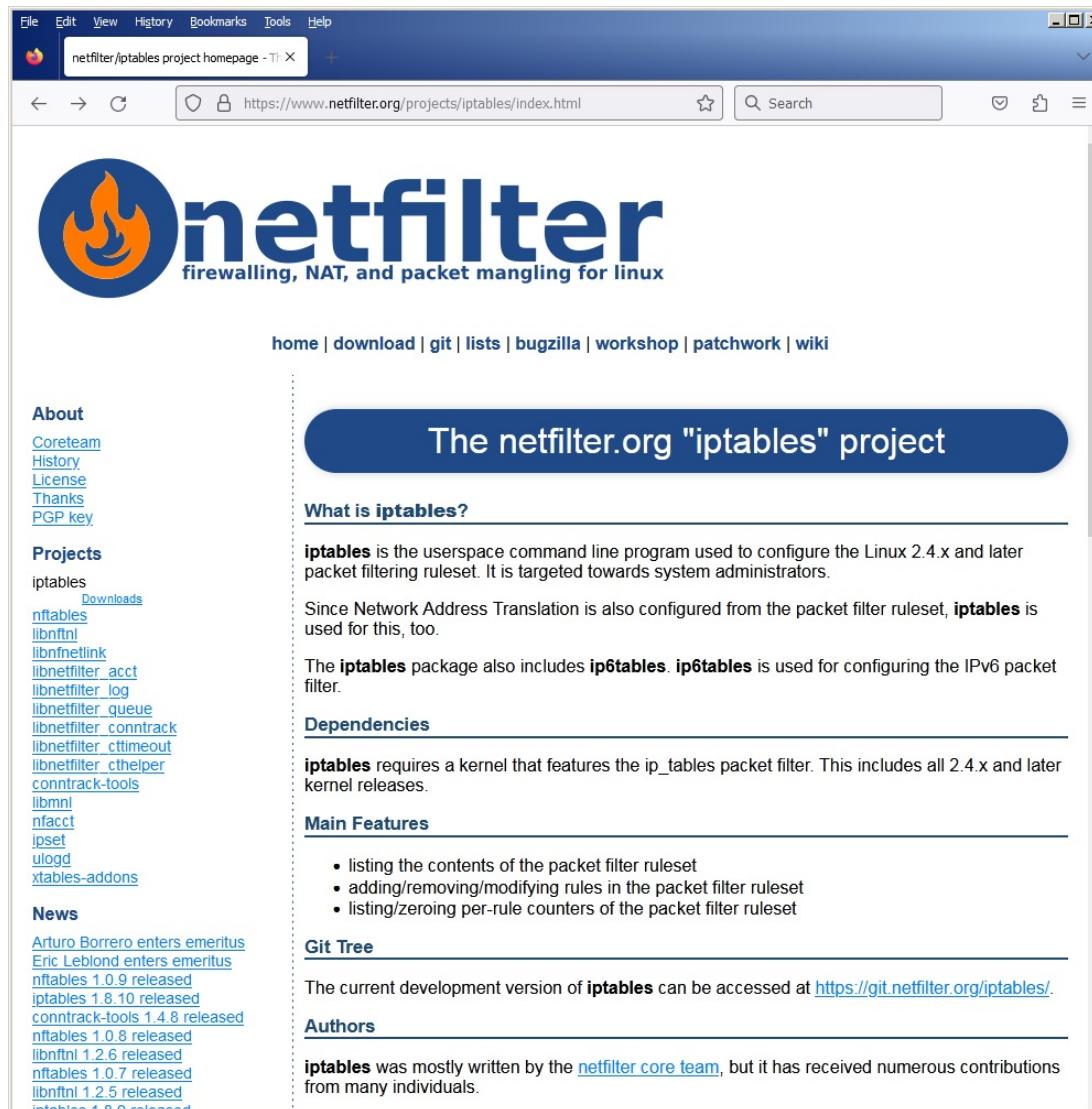
Kerio (GFI Software) KerioControl

6.0.12.1

В большинстве систем UNIX широко применяют пакет IP Filter -- в основном для целей фильтрации и NAT.

В Linux эту роль выполняет пакет IP Tables (пришел на смену Ipfwadm и IP Chains).

6.0.12.2



The screenshot shows a Firefox browser window with the title bar "netfilter/iptables project homepage - Ti X". The address bar displays the URL <https://www.netfilter.org/projects/iptables/index.html>. The page content is the netfilter.org "iptables" project homepage. It features a large logo with a blue flame icon and the text "netfilter" in blue, with the subtitle "firewalling, NAT, and packet mangling for linux". Below the logo is a navigation menu with links to "home", "download", "git", "lists", "bugzilla", "workshop", "patchwork", and "wiki". On the left side, there is a sidebar with sections for "About" (links to Coreteam, History, License, Thanks, PGP key), "Projects" (links to iptables, nftables, libnftnl, libnftnetlink, libnetfilter_acct, libnetfilter_log, libnetfilter_queue, libnetfilter_conntrack, libnetfilter_cttimeout, libnetfilter_cthelper, conntrack-tools, libmnl, nfact, ipset, ulogd, xtables-addons), and "News" (links to Arturo Borrero enters emeritus, Eric Leblond enters emeritus, nftables 1.0.9 released, iptables 1.8.10 released, conntrack-tools 1.4.8 released, nftables 1.0.8 released, libnftnl 1.2.6 released, nftables 1.0.7 released, libnftnl 1.2.5 released, iptables 1.8.0 released). The main content area has a blue header "The netfilter.org "iptables" project". Below it, a section titled "What is iptables?" states: "iptables is the userspace command line program used to configure the Linux 2.4.x and later packet filtering ruleset. It is targeted towards system administrators." It also notes that since Network Address Translation is configured from the packet filter ruleset, "iptables" is used for this. The "iptables" package includes "ip6tables", which is used for IPv6 packet filtering. A "Dependencies" section states that "iptables" requires a kernel with the ip_tables packet filter. A "Main Features" section lists: listing the contents of the packet filter ruleset, adding/removing/modifying rules in the packet filter ruleset, and listing/zeroing per-rule counters of the packet filter ruleset. A "Git Tree" section provides a link to the current development version of "iptables" at <https://git.netfilter.org/iptables/>. An "Authors" section notes that "iptables" was mostly written by the [netfilter core team](#), with contributions from many individuals.

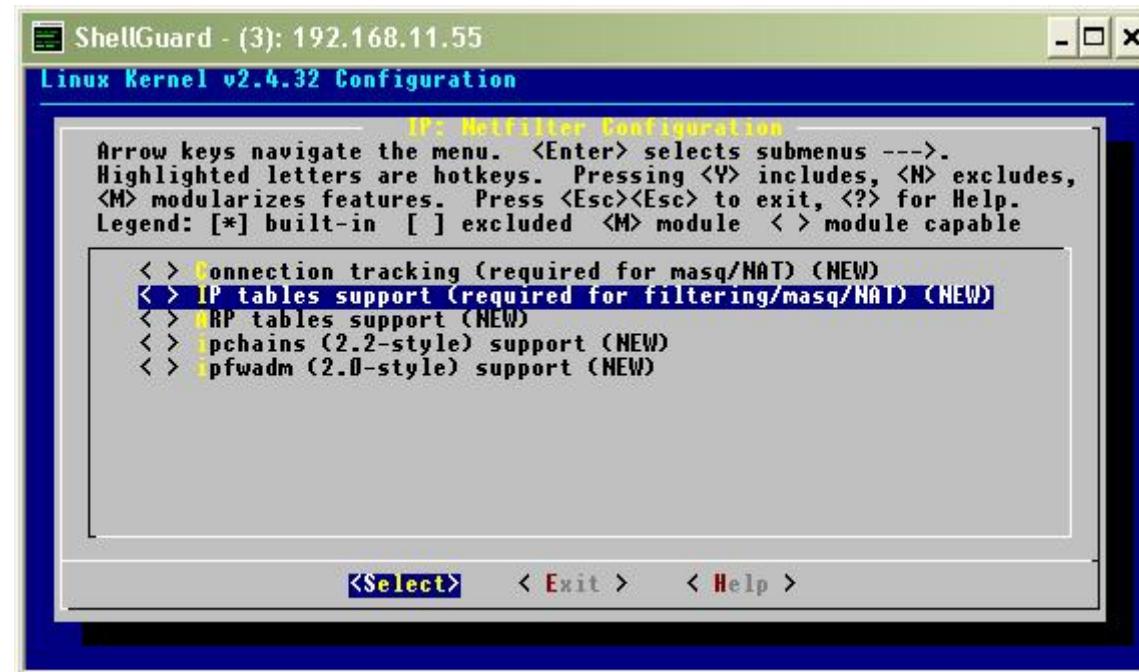
IP Tables project site

6.0.12.3

Для нормальной работы IP Tables должны быть включены некоторые опции ядра.

Для обеспечения возможности управления введен одноименный сервис iptables.

6.0.12.4



Раздел IP tables support при конфигурировании ядра Linux

6.0.12.5

Фильтры строятся на основе правил (rules).

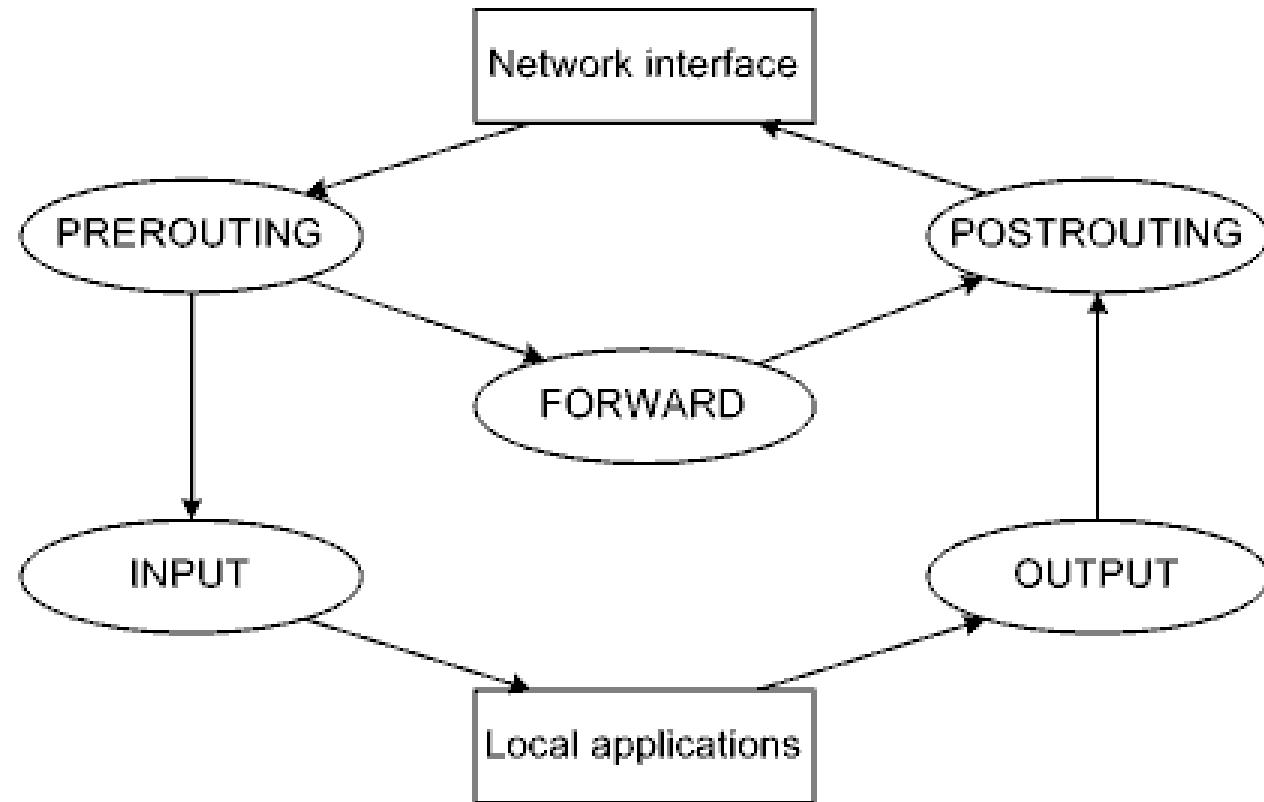
Каждое правило -- это строка, содержащая в себе условия, определяющие подпадает ли пакет под правило, и действие, которое необходимо осуществить в случае выполнения условий.

Правила могут объединяться в цепочки (chains) и образовывать сложную иерархию.

Следовательно, при работе с IP Tables необходимо внимательно проверять содержимое и последовательность правил.

6.0.12.6

Стандартные цепочки IP Tables.



6.0.12.7a

Общий формат правила:

```
iptables [-t table] command [match] [target/jump]
```

6.0.12.7b

Примеры таблиц (tables):

`filter` -- нужна для фильтрации пакетов;

`mangle` -- нужна для внесения изменений в заголовки пакетов (например, в поле TTL);

`nat` -- нужна для преобразования адресов.

6.0.12.7с

Примеры команд (commands):

- A (--append) -- добавить новое правило в конец цепочки;
- D (--delete) -- удалить правило из цепочки;
- F (--flush) -- удалить все правила из цепочки;
- I (--insert) -- вставить новое правило в цепочку;
- L (--list) -- вывести на экран список правил в цепочке;
- N (--new-chain) -- создать новую цепочку с названием в таблице;
- P (--policy) -- определить политику по умолчанию для цепочки;
- R (--replace) -- заменить одно правило другим в цепочке;
- X (--delete-chain) -- удалить цепочку из таблицы.

6.0.12.7d

Примеры критериев (matches):

- d (--destination) -- нужен для указания адреса назначения;
- f (--fragment) -- нужен для включения поддержки фрагментации;
- i (--in-interface) -- нужен для указания сетевого интерфейса, принимающего пакеты;
- o (--out-interface) -- нужен для указания сетевого интерфейса, передающего пакеты;
- p (--protocol) -- нужен для указания протокола;
- s (--source) -- нужен для указания адреса источника.

6.0.12.7e

Примеры действий (targets) :

ACCEPT -- пакет прекращает движение по цепочке (и всем цепочкам, приведшим к текущей) и считается пропущенным, но он может быть отброшен следующими цепочками;

DNAT -- подмена адреса назначения;

DROP -- пакет отбрасывается (окончательно);

LOG -- протоколирование пакета или связанных с его прохождением событий;

MASQUERADE -- подмена адреса источника без явного указания заменяющего адреса;

REJECT -- равно DROP плюс посылка ответного ICMP-сообщения о недостижимости;

SNAT -- подмена адреса источника.

Переходы (jumps) позволяют передавать пакет другим цепочкам.

6.0.12.8

```
#iptables -t nat -A PREROUTING -p tcp -d 10.10.10.1 -dport 80  
-j DNAT --to-destination 192.168.0.2
```

Пример правила IP Tables

6.0.12.9

В Linux существуют два основных варианта создания мостов «на лету»: более старый -- с помощью команды `brctl`, и более новый -- с помощью команды `ip`.

6.0.12.10

```
#brctl addbr br0
#ifconfig br0 192.168.11.11 netmask 255.255.255.224
#ifconfig br0 up

#brctl addif br0 eth0
#brctl addif br0 eth1

#ip link add name br0 type bridge
#ip link set dev br0 up
#ip address add 192.168.11.11/27 dev br0

#ip link set eth0 master br0
#ip link set eth1 master br0
```

6.0.12.11

/etc/sysconfig/network-scripts/ifcfg-br0:

```
DEVICE=br0
ONBOOT=yes
BOOTPROTO=static
IPADDR=192.168.11.11
NETMASK=255.255.255.224
```

/etc/sysconfig/network-scripts/ifcfg-eth0 (eth1 и так далее) :

```
DEVICE=eth0
ONBOOT=yes
BOOTPROTO=none
BRIDGE=br0
```

Примеры постоянных мостов (ветви Red Hat и SUSE)

6.0.12.12

```
/etc/network/interfaces:  
...  
iface eth0 inet manual  
...  
iface eth1 inet manual  
...  
  
auto br0  
iface br0 inet static  
    address 192.168.11.11  
    netmask 255.255.255.224  
    bridge_ports eth0 eth1
```

Примеры постоянных мостов (ветвь Debian)

6.0.12.13

В Linux, на примере eth0, логические сетевые интерфейсы при IP aliasing -- это eth0:0 (не то же самое что eth0), eth0:1, eth0:2 и так далее. Логический сетевой интерфейс может иметь только один IP-адрес из соответствующей подсети. IP-адреса логических сетевых интерфейсов могут сосуществовать с IP-адресом eth0.

6.0.12.14

```
#ifconfig
eth0      Link encap:Ethernet HWaddr 00:08:C7:2A:FD:EC
          inet addr:192.168.0.1 Bcast:192.168.0.255 Mask:255.255.255.0
                  UP BROADCAST MULTICAST MTU:1500 Metric:1
                  RX packets:0 errors:0 dropped:0 overruns:0 frame:0
                  TX packets:0 errors:0 dropped:0 overruns:0 carrier:0
                  collisions:0 txqueuelen:1000
                  RX bytes:0 (0.0 b) TX bytes:0 (0.0 b)

eth0:0    Link encap:Ethernet HWaddr 00:08:C7:2A:FD:EC
          inet addr:192.168.3.1 Bcast:192.168.3.255 Mask:255.255.255.0
                  UP BROADCAST MULTICAST MTU:1500 Metric:1

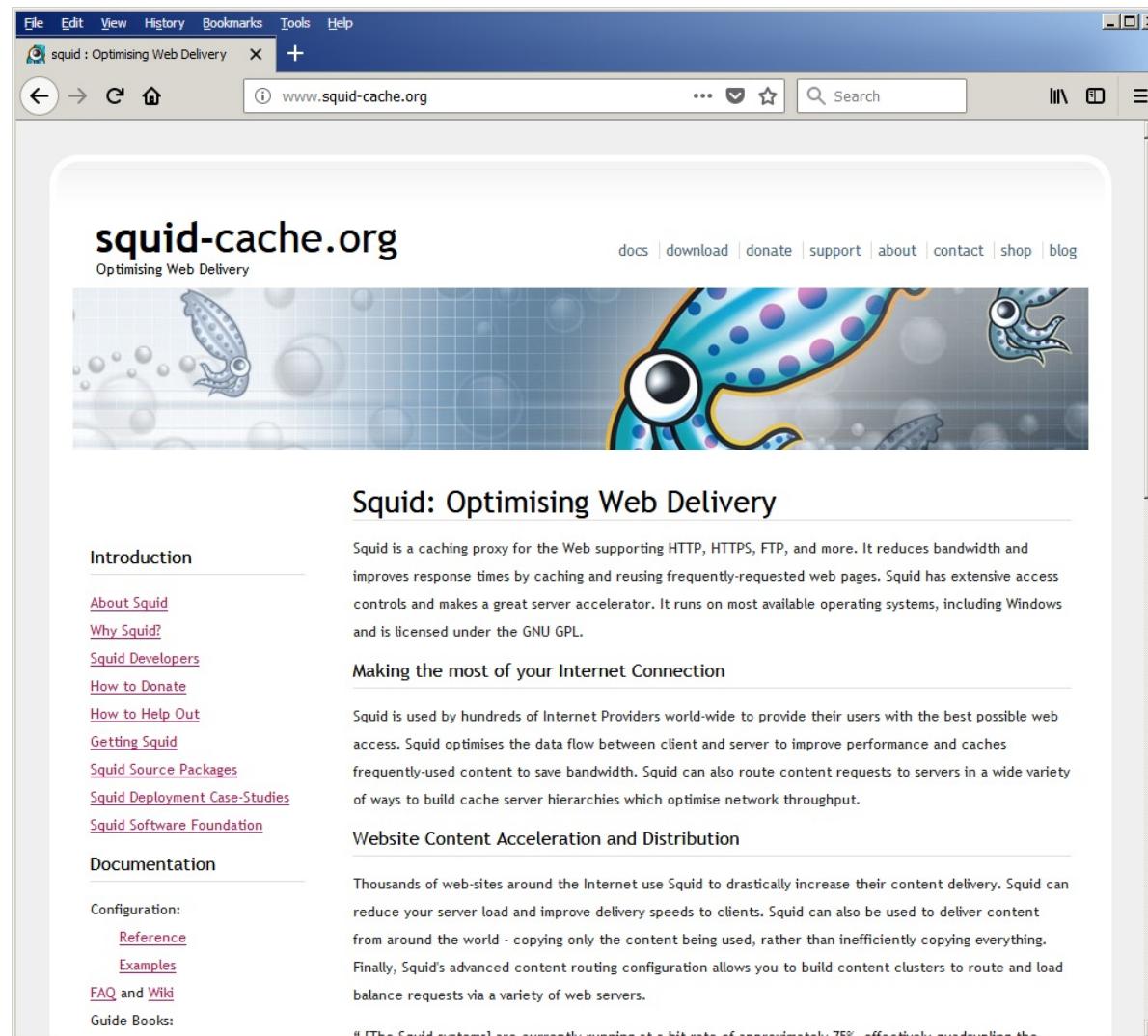
eth0:1    Link encap:Ethernet HWaddr 00:08:C7:2A:FD:EC
          inet addr:192.168.2.1 Bcast:192.168.2.255 Mask:255.255.255.0
                  UP BROADCAST MULTICAST MTU:1500 Metric:1
```

Пример IP aliasing в Linux

6.0.13.1

Де facto стандартным прокси-сервером для систем UNIX, в том числе и Linux, является пакет Squid со стандартным конфигурационным файлом /etc/squid/squid.conf.

6.0.13.2



The screenshot shows a web browser window with the title "squid : Optimising Web Delivery". The address bar displays "www.squid-cache.org". The main content area features the "squid-cache.org" logo with the tagline "Optimising Web Delivery". Below the logo is a cartoon illustration of a squid swimming in bubbles. The page title is "Squid: Optimising Web Delivery". The main text describes Squid as a caching proxy for the Web supporting HTTP, HTTPS, FTP, and more, noting its ability to reduce bandwidth and improve response times through caching and reusing frequently-requested web pages. It also mentions extensive access controls and its use as a server accelerator across various operating systems, including Windows, and its licensing under the GNU GPL. The page includes sections for "Introduction", "Documentation", and "Configuration", each with a list of links. A small note at the bottom states: "The Squid system is currently running at a bit-rate of approximately 75%, effectively quadrupling the".

Squid project site

6.0.13.3

```
http_port 3128

cache_mem 512 MB

cache_dir ufs /var/cache/squid/ 512 16 256

access_log /var/log/squid/access.log
cache_store_log /var/log/squid/store.log
logfile_rotate 0

cache_log /var/log/squid/cache.log

ftp_user anonymous@
ftp_passive off

url_rewrite_program /usr/bin/squidGuard

request_timeout 1 minute

error_directory /usr/share/squid/errors/English

auth_param basic program /usr/lib/squid/ncsa_auth /etc/shadow
auth_param basic children 5
auth_param basic realm Squid Authentication
auth_param basic credentialsttl 2 hours
auth_param basic casesensitive on

acl EVM src 192.168.11.0/24
acl NCSA-USERS proxy_auth REQUIRED

http_access allow localhost
http_access allow EVM
http_access allow NCSA-USERS
http_access deny all
```

Пример файла squid.conf

6.0.14.1

Фильтрация в IOS (ACLs) будет рассмотрена в дальнейшем -- в рамках дополнительных возможностей активного сетевого оборудования.

6.0.14.2

IOS поддерживает все теоретические варианты NAT. Статические и динамические преобразования совместимы (даже по несколько в одном направлении), но статические нужно конфигурировать раньше динамических.

Отдельно взятое правило преобразований задают командой `ip nat`.

Если правило относится к динамическим преобразованиям, аргумент `overload` разрешает совместное использование пула адресов и включает РАТ. Если правило относится к статическим преобразованиям, РАТ включает аргумент `extendable`.

Обязательно нужно привязать сетевые интерфейсы к внутренней сети (`inside`) и сети публичного доступа (`outside`) командами `ip nat inside` и `ip nat outside` соответственно (можно и не по одному), что указывает направления преобразований.

Вполне закономерно, что правила, относящиеся к статическим преобразованиям, активны постоянно -- при запросах как со стороны `inside`, так и со стороны `outside`, а правила, относящиеся к динамическим преобразованиям, активируются -- только при запросах со стороны `inside`, хотя и те, и другие, работают в обе стороны.

6.0.14.3

Важно правильно понимать Cisco-терминологию, связанную с адресацией при NAT-преобразованиях:

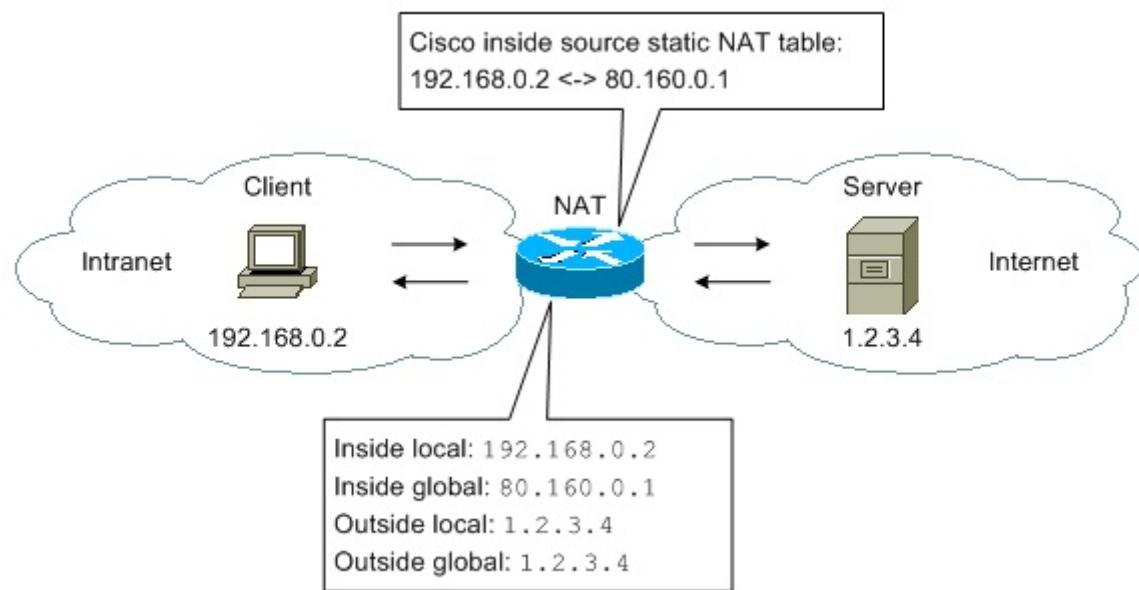
1. Inside local -- адрес расположенной в сети `inside` станции как он «виден» в сети `inside`.
2. Inside global -- адрес расположенной в сети `inside` станции как он «виден» в сети `outside`.
3. Outside local -- адрес расположенной в сети `outside` станции как он «виден» в сети `inside`.
4. Outside global -- адрес расположенной в сети `outside` станции как он «виден» в сети `outside`.

6.0.14.4

Cisco-варианты NAT:

1. `ip nat inside source static` -- в направлении из сети `inside` в сеть `outside` подменяется IP-адрес источника (и IP-адрес назначения в обратном направлении), то есть адрес `inside local` заменяется адресом `inside global` (классический статический source NAT).
2. `ip nat inside source list` -- замены аналогично `ip nat inside source static`, плюс позволяет задействовать список адресов `inside local` и пул адресов `inside global` (динамический source NAT).
3. `ip nat inside destination list` -- в направлении из сети `outside` в сеть `inside` подменяется IP-адрес назначения, то есть адрес `inside global` заменяется адресом `inside local` -- фактически то же самое, что и `ip nat inside source`, но позволяет задействовать список адресов `inside global` и пул адресов `inside local` (используется редко).
4. `ip nat outside source static` -- в направлении из сети `outside` в сеть `inside` подменяется IP-адрес источника, то есть адрес `outside global` заменяется адресом `outside local` (статический destination NAT, используется редко).
5. `ip nat outside source list` -- замены аналогично `ip nat outside source static`, плюс позволяет задействовать список адресов `outside global` и пул адресов `outside local` (динамический destination NAT, используется редко).

6.0.14.5



По умолчанию строка NAT-таблицы считается валидной 24 часа -- TCP, 5 минут -- UDP, 1 минуту -- ICMP.

6.0.14.6

Если адрес `inside global` совпадает с адресом сетевого интерфейса, относящегося к сети `outside`, то вместо адреса `inside global` можно указать название соответствующего сетевого интерфейса (однако переносимость конфигурации снижается).

6.0.14.7

```
Router(config)#ip nat inside source static 192.168.0.207 191.0.0.207  
  
Router(config)#ip nat pool EXAMPLE-NAT-POOL 191.0.0.209 191.0.0.212 netmask 255.255.255.240  
Router(config)#access-list 1 permit 192.168.0.208 0.0.0.15  
Router(config)#ip nat inside source list 1 pool EXAMPLE-NAT-POOL  
  
Router(config)#ip nat inside source static tcp 192.168.0.206 22 191.0.0.213 2222  
  
Router(config)#interface fa0/0  
Router(config-if)#ip nat inside  
  
Router(config)#interface fa0/1  
Router(config-if)#ip nat outside
```

6.0.14.8

Для просмотра NAT-таблицы (активных преобразований) используют команду `show ip nat translations`, для просмотра статистики -- `show ip nat statistics`.

6.0.14.9

```
Router#show ip nat translations
Pro Inside global      Inside local        Outside local      Outside global
tcp 80.94.160.200:8005 192.168.11.5:22    ---               ---
tcp 80.94.160.200:80   192.168.11.5:80    ---               ---
tcp 80.94.160.200:443  192.168.11.5:443   37.9.113.186:38398 37.9.113.186:38398
tcp 80.94.160.200:443  192.168.11.5:443   ---               ---
tcp 80.94.160.200:2929 192.168.11.29:22   ---               ---
tcp 80.94.160.200:2222 192.168.11.49:22   ---               ---
tcp 80.94.160.200:2411 192.168.11.51:22   ---               ---
tcp 80.94.160.200:41998 192.168.11.51:41998 80.94.162.85:993 80.94.162.85:993
tcp 80.94.160.200:44652 192.168.11.51:44652 80.94.162.85:993 80.94.162.85:993
tcp 80.94.160.200:2401 192.168.11.55:22   ---               ---
tcp 80.94.160.200:2402 192.168.11.179:22  ---               ---
tcp 80.94.160.200:1180 192.168.11.180:22  ---               ---
tcp 80.94.160.200:7272 192.168.59.145:22  ---               ---
tcp 80.94.160.200:2727 192.168.59.147:22  ---               ---
tcp 80.94.160.200:8888 192.168.59.147:80   ---               ---
tcp 80.94.160.200:8880 192.168.59.147:443 ---               ---
tcp 80.94.160.200:8180 192.168.59.147:8080 ---               ---
tcp 80.94.160.200:8190 192.168.59.147:8090 ---               ---
```

6.0.14.10

При любых NAT-преобразованиях автоматически создаются и конфигурируются (только автоматически) виртуальные сетевые интерфейсы NVIs (NAT Virtual Interfaces).

Но действуют эти интерфейсы в особых случаях. Если необходимо чтобы сетевой интерфейс относился к сети `inside` одного NAT-преобразования и одновременно относился к сети `outside` другого NAT-преобразования, то необходимо использовать другие команды -- чтобы маршрутизатор определял направления автоматически.

Таковые NAT-преобразования описывают командами `ip nat source static` и `ip nat source list`, а на соответствующих сетевых интерфейсах просто включают поддержку NAT командой `ip nat enable`.

6.0.14.11

Мосты в IOS будут рассмотрены в дальнейшем -- в рамках возможностей беспроводного оборудования.

6.0.14.12

Для присвоения IP-псевдонимов в Cisco IOS используют команду `ip address` с аргументом `secondary`.

6.0.14.13

```
Router(config)#interface se0/0/0
Router(config-if)#ip address 192.168.0.1 255.255.255.0
Router(config-if)#ip address 192.168.0.10 255.255.255.0 secondary
Router(config-if)#ip address 192.168.2.1 255.255.255.0 secondary
Router(config-if)#exit
```