NAT (от англ. Network Address Translation — «преобразование сетевых адресов») — это механизм в сетях TCP/IP, позволяющий преобразовывать IP-адреса транзитных пакетов.

Механизм NAT определён в RFC 1631, RFC 3022.

Функционирование

Преобразование адресов методом NAT может производиться почти любым маршрутизирующим устройством — маршрутизатором, сервером доступа, межсетевым экраном. Наиболее популярным является SNAT, суть механизма которого состоит в замене адреса источника (англ. source) при прохождении пакета в одну сторону и обратной замене адреса назначения (англ. destination) в ответном пакете. Наряду с адресами источник/назначение могут также заменяться номера портов источника и назначения.

Ваш компьютер может быть подключен к интернету напрямую. Тогда говорят, что у него внешний IP адрес. Обычно это значит, что компьютер подключен сразу к модему (DSL, кабельному или обычному аналоговому).

За NAT означает, что ваш компьютер подключен не к интернету, а к локальной сети. Тогда у него внутренний IP адрес, из интернета сам по себе недоступный.

Доступ к интернету ваш компьютер получает через NAT - процесс трансляции внутренних адресов во внешние и обратно. NAT-устройство обычно называют раутером.

NAT позволяет хостам из intranet прозрачным образом обращаться к хостам в общедоступном пространстве, при этом для внутренних хостов не требуется наличия зарегистрированных (и наиболее дефицитных) сетевых Internet-адресов.

Internet Engineering Task Force вот уже почти десять лет предупреждает о грядущем дефиците адресов в существующем сейчас адресном пространстве, формируемом IPv4. И хотя, по-видимому, новая версия протокола IPv6 будет способна достаточно долгое время удовлетворять требованиям постоянно растущего Internet, предложенные за последние несколько лет меры уже получили законный статус.

Одно из подобных предложений описывает опубликованный в 1994 году стандарт RFC 1631, "Транслятор сетевых IP-адресов" (The IP Network Address Translator). В период становления Internet потребители были вынуждены использовать уникальные в глобальном масштабе сетевые адреса вне зависимости от того, планировали они подключаться к глобальному пространству Internet или нет. Идея состояла в том, чтобы избежать потенциальных конфликтов в том случае, если бывшая ранее частной сеть в конце концов присоединялась к общедоступной Internet.

В условиях продолжающего экспоненциально расширяться Internet присваивание "хороших", то есть уникальных в глобальном смысле сетевых адресов частным сетям стало считаться непроизводительной тратой ценных виртуальных ресурсов. В соответствии со стандартом Network Address Translation определенные IP-адреса выделены для параллельного использования в частных сетях и в общедоступной сети. Как указано в спецификации RFC 1597 "Резервирование адресов для частных Internet" (Address Allocation for Private Internets), разрешено использовать адреса в следующих диапазонах: от 10.0.0.0 до 10.255.255.255; от 172.16.0.0 до 172.31.255.255; от 192.168.0.0 до 192.168.255.255. В соответствии с условиями этого соглашения не предполагается, что по этим адресам маршрутизаторы передают какие-либо пакеты в Internet.

Простейшее устройство NAT имеет два сетевых соединения: одно с Internet и одно с частной сетью. Хосты в частной сети, использующие частные IP-адреса (иногда также называемые адресами Network 10, которые начинаются с 10.0.0.0 в диапазоне, выделенном для частного использования), связываются с Internet, передавая пакеты непосредственно на устройство NAT. В отличие от обычных маршрутизаторов, которые просто считывают адреса отправителя и получателя каждого пакета, прежде чем передать их по назначению, устройства NAT модифицируют заголовки пакетов, заменяя адрес отправителя в частной сети на свой собственный Internet-адрес.

Можно проверить под каким IP адресом вас видят другие компьютеры в интернете. Например на whatsmyip.org ("Your IP Address is x.x.x.x" вверху страницы) или на myipaddress.com.

Если IP адрес вашего компьютера совпадает с показанным одним из этих сайтов, то вы точно подключены к интернету напрямую.

Если, к примеру, один сетевой компьютер обращается к FTP-серверу (порт 21), то при открытии сокета операционная система присваивает сессии любой порт выше 1023. Например, это может быть порт 2153. Тогда IP-пакет, отправляемый со стороны ПК к FTP-серверу, будет содержать IP-адрес отправителя, порт отправителя (2153), IP-адрес получателя и порт назначения (21). IP-адрес и порт отправителя будут использоваться для ответа сервера клиенту. Использование разных портов для различных сетевых сессий позволяет клиентам сети одновременно устанавливать несколько сессий с различными серверами или с сервисами одного сервера.

Теперь рассмотрим процесс установления сессии при использовании NAT-маршрутизатора на границе внутренней сети и сети Интернет.

Когда клиент внутренней сети устанавливает связь с сервером внешней сети, то, как и в случае установки соединения между двумя ПК, открывается сокет, определяемый IP-адресом источника, портом источника, IP-адресом назначения, портом назначения и сетевым протоколом. Когда приложение передает данные через этот сокет, IP-адрес источника и порт источника вставляются в пакет в поля параметров источника. Поля параметров пункта назначения будут содержать IP-адрес сервера и порт сервера. К примеру, компьютер внутренней сети с IP-адресом 192.168.0.1 может обратиться к Web-серверу Глобальной сети с IP-адресом 64.233.188.104. В этом случае операционная система клиента может назначить установленной сессии порт 1251 (порт источника), а порт назначения — это порт Web-сервиса, то есть 80. Тогда в заголовке отправляемого пакета будут указаны следующие атрибуты (рис. 2):

IP-адрес источника: 192.168.0.1;

порт источника: 1251;

IP-адрес получателя: 64.233.183.104;

порт получателя: 80;

протокол: TCP.

Устройство NAT (маршрутизатор) перехватывает исходящий из внутренней сети пакет и заносит в свою внутреннюю таблицу сопоставление портов источника и получателя пакета, используя IP-адрес назначения, порт назначения, внешний IP-адрес устройства NAT, внешний порт, сетевой протокол, а также внутренние IP-адрес и порт клиента.

Предположим, что в рассмотренном выше примере NAT-маршрутизатор имеет внешний IP-адрес 195.2.91.103 (адрес WAN-порта), а для установленной сессии внешний порт NAT-устройства — 3210. В этом случае внутренняя таблица сопоставления портов источника и получателя пакета содержит следующую информацию:

IP-адрес источника: 192.168.0.1;

порт источника: 1251;

внешний IP-адрес

NAT-устройства: 195.2.91.103;

внешний порт NAT-устройства: 3210;

IP-адрес получателя: 64.233.183.104;

порт получателя: 80;

протокол: TCP.

Затем устройство NAT «транслирует» пакет, преобразуя в пакете поля источника: внутренние IP-адрес и порт клиента заменяются внешними IP-адресом и портом устройства NAT. В рассмотренном примере преобразованный пакет будет содержать следующую информацию:

IP-адрес источника: 195.2.91.103;

порт источника: 3210;

IP-адрес получателя: 64.233.183.104;

порт получателя: 80;

протокол: TCP.

Преобразованный пакет пересылается по внешней сети и в итоге попадает на заданный сервер.

Получив пакет, сервер будет направлять ответные пакеты на внешний IP-адрес и порт устройства NAT (маршрутизатора), указывая в полях источника свои собственные IP-адрес и порт (рис. 3). В рассмотренном примере ответный пакет от сервера будет содержать в заголовке следующую информацию:

IP-адрес источника: 64.233.183.104;

порт источника: 80;

IP-адрес получателя: 195.2.91.103;

порт получателя: 3210;

протокол: TCP.

Устройство NAT принимает эти пакеты от сервера и анализирует их содержимое на основе своей таблицы сопоставления портов. Если в таблице будет найдено сопоставление порта, для которого IP-адрес источника, порт источника, порт назначения и сетевой протокол из входящего пакета совпадают с IP-адресом удаленного узла, с удаленным портом и с сетевым протоколом, указанным в сопоставлении портов, то NAT выполнит обратное преобразование: заменит внешний IP-адрес и внешний порт в полях назначения пакета на IP-адрес и внутренний порт клиента внутренней сети. Таким образом, пакет, передаваемый во внутреннюю сеть, для рассмотренного выше примера будет иметь следующие атрибуты:

IP-адрес источника: 64.233.183.104;

порт источника: 80;

IP-адрес получателя: 192.168.0.1;

порт получателя: 1251;

протокол: TCP.

Однако если в таблице сопоставления портов не находится соответствия, то входящий пакет отвергается и соединение разрывается.

Благодаря NAT-маршрутизатору любой ПК внутренней сети получает возможность передавать данные в Глобальную сеть с использованием внешнего IP-адреса и порта маршрутизатора. При этом IP-адреса внутренней сети, как присвоенные сессиям порты, остаются невидимыми со стороны внешней сети.

Помимо source NAT (предоставления пользователям локальной сети с внутренними адресами доступа к сети Интернет) часто применяется также destination NAT, когда обращения извне транслируются межсетевым экраном на сервер в локальной сети, имеющий внутренний адрес и потому недоступный извне сети непосредственно (без NAT).

Существует 3 базовых концепции трансляции адресов: статическая (Static Network Address Translation), динамическая (Dynamic Address Translation ), маскарадная (NAPT, NAT Overload, PAT).

Статический NAT — Отображение незарегистрированного IP адреса на зарегистрированный IP адрес на основании один к одному. Особенно полезно, когда устройство должно быть доступным снаружи сети.

Динамический NAT — Отображает незарегистрированный IP адрес на зарегистрированный адрес от группы зарегистрированных IP адресов. Динамический NAT также устанавливает непосредственное отображение между незарегистрированным и зарегистрированным адресом, но отображение может меняться в зависимости от зарегистрированного адреса, доступного в пуле адресов, во время коммуникации.

Перегруженный NAT (NAPT, NAT Overload, PAT, маскарадинг) — форма динамического NAT, который отображает несколько незарегистрированных адресов в единственный зарегистрированный IP адрес, используя различные порты. Известен также как PAT (Port Address Translation)

При перегрузке, каждый компьютер в частной сети транслируется в тот же самый адрес, но с различным номером порта.

Маскарадинг (англ. Masquerading) — тип трансляции сетевого адреса, при которой адрес отправителя подставляется динамически, в зависимости от назначенного интерфейсу адреса. Разделение SNAT (source NAT) и маскарадинга характерно для iptables, в маршрутизаторах cisco операция трансляции (ip nat source) умеет работать как с заданными адресами, так и с интерфейсами (используя автоматически назначенный интерфейсу адрес).

Ещё одной особенностью маскарадинга (в iptables) является «забывание» про установленные трансляции при остановке (down) интерфейса. Это связано с тем, что после поднятия интерфейса его адрес, вероятнее всего (в случае DHCP/Dialup) будет другим, и записи о ранее выполненных трансляциях не будут иметь смысла.

Преимущества

NAT выполняет три важных функции.

Позволяет сэкономить IP-адреса (только в случае использования NAT в режиме PAT), транслируя несколько внутренних IP-адресов в один внешний публичный IP-адрес (или в несколько, но меньшим количеством, чем внутренних). По такому принципу построено большинство сетей в мире: на небольшой район домашней сети местного провайдера или на офис выделяется 1 «белый» (то есть внешний) IP-адрес, за которым работают и получают доступ вовне все «серые» (то есть внутренние) IP-адреса.

Позволяет предотвратить или ограничить обращение снаружи ко внутренним хостам, оставляя возможность обращения изнутри наружу. При инициации соединения изнутри сети создаётся трансляция. Ответные пакеты, поступающие снаружи, соответствуют созданной трансляции и поэтому пропускаются. Если для пакетов, поступающих снаружи, соответствующей трансляции не существует (а она может быть созданной при инициации соединения или статической), они не пропускаются.

Позволяет скрыть определённые внутренние сервисы внутренних хостов/серверов. По сути, выполняется та же указанная выше трансляция на определённый порт, но возможно подменить внутренний порт официально зарегистрированной службы (например, 80-й порт TCP (HTTP-сервер) на внешний 54055-й). Тем самым, снаружи, на внешнем IP-адресе после трансляции адресов на сайт (или форум) для осведомлённых посетителей можно будет попасть по адресу http://example.org:54055, но на внутреннем сервере, находящимся за NAT, он будет работать на обычном 80-м порту. Повышение безопасности и скрытие «непубличных» ресурсов.

Недостатки

При использовании NAT хосты Internet взаимодействуют напрямую с NAT-устройством, а не с реальным хостом в частной сети. Входящие пакеты передаются по IP-адресу устройства NAT, а устройство меняет адрес назначения в заголовке пакета со своего собственного Internet-адреса на адрес частной сети реального хоста назначения.

В результате - теоретически - единый уникальный в глобальном масштабе IP-адрес может скрывать за собой сотни, тысячи или даже миллионы хостов, чьи адреса определены в частной сети. На практике, однако, такой подход имеет ряд недостатков. Например, многие Internet-протоколы и приложения существенно опираются на тот факт, что сеть в действительности поддерживает соединение из конца в конец, когда пакеты передаются без каких-либо изменений от отправителя получателю. Архитектура IP-защиты, к примеру, не может поддерживать работу через устройство NAT, поскольку оригинальные заголовки, в которых указаны исходные IP-адреса отправителей, защищены с помощью цифровой подписи. Измените исходный адрес - и цифровая подпись потеряет свою целостность.

Использование NAT к тому же усложняет работу администраторов. Конечно, NAT - превосходное решение для организации, филиала или даже подразделения, которые не могут получить достаточное количество уникальных Internet-адресов. Однако при реорганизации, объединении или покупке другой компанией, когда требуется объединение двух или нескольких частных сетей, возникают большие сложности. Даже если структура организации остается достаточно стабильной, системы NAT могут случайно оказаться вложенными структурами, превращая маршрутизацию в настоящий кошмар.

Не все протоколы могут «преодолеть» NAT. Некоторые не в состоянии работать, если на пути между взаимодействующими хостами есть трансляция адресов. Некоторые межсетевые экраны, осуществляющие трансляцию IP-адресов, могут исправить этот недостаток, соответствующим образом заменяя IP-адреса не только в заголовках IP, но и на более высоких уровнях (например, в командах протокола FTP)..

Из-за трансляции адресов «много в один» появляются дополнительные сложности с идентификацией пользователей и необходимость хранить полные логи трансляций.

DoS со стороны узла, осуществляющего NAT — если NAT используется для подключения многих пользователей к одному и тому же сервису, это может вызвать иллюзию DoS-атаки на сервис (множество успешных и неуспешных попыток). Например, избыточное количество пользователей ICQ за NAT приводит к проблеме с подключением к серверу некоторых пользователей из-за превышения допустимой скорости подключений. Частичным решением проблемы является использование пула адресов (группы адресов), для которых осуществляется трансляция.

В некоторых случаях, необходимость в дополнительной настройке при работе с пиринговыми сетями и некоторыми другими программами (Battle.net-игры), в которых необходимо не только инициировать исходящие соединения, но также принимать входящие. Однако, если NAT устройство и ПО требующее дополнительной настройки поддерживает технологию Universal Plug and Play, то в этом случае настройка произойдет полностью автоматически и прозрачно для пользователя.

За границей частной сети

Хосты внутри частной сети обычно без труда связываются с серверами за пределами организации, однако хосты в Internet не всегда могут легко получать доступ к серверам частной сети. Если внешним хостам требуется такой доступ, они взаимодействуют напрямую с одним хостом - самим устройством NAT. Частная сеть, по существу, не видна внешнему миру, и считается, что весь трафик из этой сети - это трафик, отправляемый и присылаемый на устройство NAT.

Network Address Port Translation (NAPT) помогает до некоторой степени решить эту проблему за счет преобразования не только самого IP-адреса, но и порта транспортного уровня. Таким образом, входящий пакет, адресованный на порт Port 80 (как правило, используемый для HTTP-пакетов) на устройстве NAPT, может быть преобразован и передан на Web-сервер частной сети. Без преобразования порта устройство NAT не может "узнать", на какой хост частной сети передавать эти пакеты.

NAT часто рассматривают как решение проблемы защиты. В конце концов, частная сеть действительно как бы скрыта от посторонних глаз. Однако, если злоумышленник получит управление устройством NAT, уязвимой становится вся сеть целиком. Устройство не должно рассматриваться как замена межсетевому экрану, хотя простые устройства, реализующие преобразование адресов, могут оказаться полезны для защиты сетей дома или в небольшом офисе.

Хотя сторонники NAT пропагандируют его как долговременное решение проблемы дефицита адресов IPv4, на самом деле оно способно лишь временно снять остроту ситуации. Если игнорировать архитектурные проблемы и сложности, связанные с развертыванием сетей, само по себе адресное пространство IPv4 тем не менее остается ограниченным и в скором времени может быть исчерпано.

NAT Traversal

NAT Traversal (прохождение или автонастройка NAT) — это набор возможностей, позволяющих сетевым приложениям определять, что они находятся за устройством, обеспечивающим NAT, узнавать внешний IP-адрес этого устройства и выполнять сопоставление портов для пересылки пакетов из внешнего порта NAT на внутренний порт, используемый приложением; все это выполняется автоматически, пользователю нет необходимости вручную настраивать сопоставления портов или вносить изменения в какие-либо другие параметры. Однако существуют меры предосторожности в доверии к таким приложениям — они получают обширный контроль над устройством, появляются потенциальные уязвимости.

Протокол NAT Traversal (NAT-T) инкапсулирует трафик IPSec и одновременно создает пакеты UDP, которые NAT корректно пересылает. Для этого NAT-T помещает дополнительный заголовок UDP перед пакетом IPSec, чтобы он во всей сети обрабатывался как обычный пакет UDP и хост получателя не проводил никаких проверок целостности. После поступления пакета по месту назначения заголовок UDP удаляется, и пакет данных продолжает свой дальнейший путь как инкапсулированный пакет IPSec. Итак, с помощью техники NAT-T возможно установление связи между клиентами IPSec в защищённых сетях и общедоступными хостами IPSec через межсетевые экраны.

Операционные системы с поддержкой NAT

При недостаточном финансировании, либо при наличии уже существующего сервера под управлением серверной ОС возможно организовать трансляцию адресов без необходимости закупки дополнительных устройств. В таком случае, оптимальным будет наличие по крайней мере двух сетевых адаптеров в сервере (возможны варианты с одним, но при наличии trunk-VLAN).

Все существующие и использующиеся серверные ОС поддерживают простейшую трансляцию адресов.

С точки зрения же отказоустойчивости, гибкости и производительности, используются операционные системы UNIX (большинство GNU/Linux, \*BSD-системы, а также OpenSolaris и др.). Во многих из них NAT доступен «из коробки», в других возможна реализация за счёт добавления модулей в сочетании с межсетевыми экранами с функциями трансляции адресов (IPFW, IPtables и др.).

Межсетевой экран или сетевой экран — комплекс аппаратных или программных средств, осуществляющий контроль и фильтрацию проходящих через него сетевых пакетов в соответствии с заданными правилами.

Также сетевые экраны часто называют фильтрами, так как их основная задача — не пропускать (фильтровать) пакеты, не подходящие под критерии, определённые в конфигурации.

Некоторые сетевые экраны также позволяют осуществлять трансляцию адресов — динамическую замену внутрисетевых (серых) адресов или портов на внешние, используемые за пределами ЛВС.

Другие названия

Брандма́уэр (нем. Brandmauer) — заимствованный из немецкого языка термин, являющийся аналогом английского firewall в его оригинальном значении (стена, которая разделяет смежные здания, предохраняя от распространения пожара). Интересно, что в области компьютерных технологий в немецком языке употребляется слово «firewall».

Файрво́лл, файрво́л, файерво́л, фаерво́л — образовано транслитерацией английского термина firewall, эквивалентного термину межсетевой экран.

Основная задача любого брандмауэра в конечном счете сводится к обеспечению безопасности внутренней сети. Для решения этой задачи брандмауэры должны уметь маскировать защищаемую сеть, блокировать все известные типы хакерских атак, блокировать утечку информации из внутренней сети, контролировать приложения, получающие доступ во внешнюю сеть.

Для того чтобы реализовать указанные функции, брандмауэры анализируют весь трафик между внешней и внутренней сетями на предмет его соответствия тем или иным установленным критериям или правилам, определяющим условия прохождения трафика из одной сети в другую. Если трафик отвечает заданным критериям, то брандмауэр пропускает его через себя. В противном случае, то есть если установленные критерии не соблюдены, трафик блокируется брандмауэром. Брандмауэры фильтруют как входящий, так и исходящий трафики, а также позволяют управлять доступом к определенным сетевым ресурсам или приложениям. Они могут фиксировать все попытки несанкционированного доступа к ресурсам локальной сети и выдавать предупреждения о попытках проникновения.

По своему назначению брандмауэры больше всего напоминают контрольно-пропускной пункт (КПП) охраняемого объекта, где производится проверка документов всех входящих на территорию объекта и всех покидающих ее. Если пропуск в порядке — доступ на территорию разрешен. Совершенно аналогично действуют и брандмауэры, только в роли людей, проходящих через КПП, выступают сетевые пакеты, а пропуском является соответствие заголовков этих пакетов предопределенному набору правил.

Можно ли утверждать, что брандмауэр обеспечивает 100-процентную безопасность пользовательской сети или персонального ПК? Безусловно, нет. Хотя бы потому, что вообще ни одна система не дает 100-процентной гарантии безопасности. К брандмауэру стоит относиться как к средству, которое, при правильной его настройке, способно в значительной мере осложнить задачу злоумышленника по проникновению в персональный компьютер пользователя. Подчеркнем: лишь осложнить, но вовсе не гарантировать абсолютной безопасности. Кстати, если речь заходит не о защите локальной сети, а о защите отдельного ПК, имеющего доступ в Интернет, то с обеспечением его персональной безопасности успешно справляется и брандмауэр ICF (Internet Connection Firewall), встроенный в операционную систему Windows XP. Поэтому в дальнейшем мы будем говорить лишь о корпоративных аппаратных брандмауэрах, ориентированных на защиту небольших сетей.

Если брандмауэр, устанавливаемый на входе в локальную сеть, активирован по полной программе (как правило, это соответствует настройкам по умолчанию), то защищаемая им сеть полностью непроницаема и недоступна извне. Однако у столь полной непроницаемости внутренней сети есть и своя оборотная сторона. Дело в том, что в этом случае становится невозможно пользоваться Интернет-сервисами (например, ICQ и тому подобными программами), установленными на ПК. Таким образом, задача настройки брандмауэра заключается в том, чтобы в первоначально глухой стене, которую представляет собой брандмауэр для злоумышленника, проделать окошки, предоставив возможность пользовательским программам отвечать на запросы извне и в конечном счете реализовать подконтрольное взаимодействие внутренней сети с внешним миром. Однако чем больше подобных окон появляется в такой стене, тем более уязвимой становится и сама сеть. Так что еще раз подчеркнем: ни один брандмауэр не может гарантировать абсолютной безопасности защищаемой им локальной сети.

Возможности брандмауэров и степень их интеллектуальности зависят от того, на каком уровне эталонной модели OSI они функционируют. Чем выше уровень OSI, на основе которого построен брандмауэр, тем выше обеспечиваемый им уровень защиты.

Помним, что модель OSI (Open System Interconnection) включает семь уровней сетевой архитектуры. Первый, самый нижний, — это физический уровень. За ним следуют канальный, сетевой, транспортный, сеансовый уровни, уровень представления и прикладной уровень, или уровень приложений. Для того чтобы обеспечивать фильтрацию трафика, брандмауэр должен работать как минимум на третьем уровне модели OSI, то есть на сетевом уровне, где происходит маршрутизация пакетов на основе преобразования MAC-адресов в сетевые адреса. С точки зрения протокола TCP/IP этот уровень соответствует уровню IP (Internet Protocol). Получая информацию сетевого уровня, брандмауэры способны определить адрес источника и получателя пакета и проверить, допустима ли передача трафика между данными адресатами. Однако информации сетевого уровня для анализа содержимого пакета недостаточно. Брандмауэры, функционирующие на транспортном уровне модели OSI, получают несколько больше информации о пакетах и в этом смысле могут предоставлять более интеллектуальные схемы защиты сетей. Что же касается брандмауэров, работающих на уровне приложений, то им доступна полная информация о сетевых пакетах, а значит, такие брандмауэры обеспечивают наиболее надежную сетевую защиту.

В зависимости от уровня модели OSI, на которых функционируют брандмауэры, исторически сложилась следующая классификация этих устройств:

пакетный фильтр (packet filter);

шлюз сеансового уровня (circuit-level gateway);

шлюз прикладного уровня (application-level gateway);

Stateful Packet Inspection (SPI).

Отметим, что данная классификация имеет лишь исторический интерес, поскольку все современные брандмауэры относятся к категории наиболее совершенных (в плане защиты сети) SPI-брандмауэров.

Пакетные фильтры

Брандмауэры типа пакетных фильтров являются наиболее простыми (наименее интеллектуальными). Эти брандмауэры работают на сетевом уровне модели OSI или на IP-уровне стека протоколов TCP/IP. Такие брандмауэры в обязательном порядке присутствуют в каждом маршрутизаторе, поскольку любой маршрутизатор работает как минимум на третьем уровне модели OSI.

Задача пакетных фильтров заключается в фильтрации пакетов на основе информации об IP-адресе источника или получателя, а также о номерах портов.

В брандмауэрах типа пакетных фильтров каждый пакет, до того как он будет передан, анализируется на предмет соответствия критериям передачи или блокировки передачи. В зависимости от пакета и от сформированных критериев передачи брандмауэр может передать пакет, отвергнуть его или послать уведомление инициатору передачи.

Пакетные фильтры просты в реализации и практически не влияют на скорость маршрутизации.

Шлюзы сеансового уровня

Шлюзы сеансового уровня — это брандмауэры, которые работают на сеансовом уровне модели OSI или на TCP (Transport Control Protocol) уровне стека протоколов TCP/IP. Данные брандмауэры отслеживают процесс установления TCP-соединения (организацию сеансов обмена данными между оконечными машинами) и позволяют определить, является ли данный сеанс связи легитимным. Данные, передаваемые к удаленному компьютеру во внешней сети через шлюз на сеансовом уровне, не содержат информации об источнике передачи, то есть все выглядит таким образом, как будто данные отправляются самим брандмауэром, а не компьютером во внутренней (защищаемой) сети. Все брандмауэры на основе NAT-протокола являются шлюзами сеансового уровня (протокол NAT будет описан ниже).

Шлюзы сеансового уровня также не оказывают существенного влияния на скорость маршрутизации. В то же время эти шлюзы не способны осуществлять фильтрацию отдельных пакетов.

Шлюзы прикладного уровня

Шлюзы прикладного уровня, или proxy-серверы, функционируют на прикладном уровне модели OSI. Прикладной уровень отвечает за доступ приложений в сеть. К задачам этого уровня относятся перенос файлов, обмен почтовыми сообщениями и управление сетью. Получая информацию о пакетах на прикладном уровне, шлюзы прикладного уровня могут реализовывать блокировку доступа к определенным сервисам. К примеру, если шлюз прикладного уровня сконфигурирован как Web-proxy, то любой трафик, относящийся к протоколам Telnet, FTP, Gopher, будет заблокирован. Поскольку эти брандмауэры анализируют пакеты на прикладном уровне, они способны осуществлять фильтрацию специфических команд, например, http:post, get и т.д. Данная функция недоступна ни пакетным фильтрам, ни шлюзам сеансового уровня. Шлюзы прикладного уровня могут также использоваться для регистрации активности отдельных пользователей и для установления ими сеансов связи. Эти брандмауэры предлагают более надежный способ защиты сетей по сравнению со шлюзами сеансового уровня и пакетными фильтрами.

SPI-брандмауэры

Последний тип брандмауэров — Stateful Packet Inspection (SPI) — объединяет в себе преимущества одновременно и пакетных фильтров, и шлюзов сеансового уровня, и шлюзов прикладного уровня. То есть фактически речь идет о многоуровневых брандмауэрах, которые работают одновременно на сетевом, сеансовом и прикладном уровнях.

SPI-брандмауэры осуществляют фильтрацию пакетов на сетевом уровне, определяют легитимность установления сеанса связи на основании данных сеансового уровня и анализируют содержимое пакетов, основываясь на данных прикладного уровня.

Эти брандмауэры реализуют наиболее надежный способ защиты сетей и на данный момент являются стандартом де-факто.

Методология и возможности по настройке брандмауэров зависят от конкретной модели. К сожалению, не существует единых правил настройки и тем более единообразного интерфейса. Можно говорить лишь о неких общих правилах, которых стоит придерживаться. Собственно, основное правило довольно простое — необходимо запрещать все, что не требуется для нормального функционирования сети.

Чаще всего возможности по настройке брандмауэров сводятся к активированию некоторых предопределенных правил и к созданию статических правил в виде таблицы.

Другой способ настройки брандмауэра сводится к созданию статических правил, которые позволяют не только защитить сеть снаружи, но и ограничить пользователям локальной сети доступ во внешнюю сеть. Возможности по созданию правил достаточно гибкие и позволяют реализовать практически любую ситуацию. Для создания правила задаются IP-адрес (или диапазон адресов) источника, порты источника, IP-адреса и порты получателя, тип протокола, направление передачи пакета (из внутренней сети во внешнюю или наоборот), а также действие, которое должно предприниматься при обнаружении пакета с означенными свойствами (отбросить или пропустить пакет). К примеру, если требуется запретить пользователям внутренней сети (диапазон IP-адресов: 192.168.1.1-192.168.1.100) обращение к FTP-серверу (порт 21), расположенному по внешнему IP-адресу 64.233.183.104, то правило может быть сформулировано следующим образом:

направление передачи пакетов: LAN-to-WAN;

IP-адреса источника: 192.168.1.1-192.168.1.100;

порт источника: 1-65535;

IP-адрес получателя: 64.233.183.104;

порт получателя: 21;

протокол: TCP;

действие: drop.

все современные маршрутизаторы со встроенными брандмауэрами поддерживают протокол трансляции сетевых адресов NAT (Network Address Translation).

Разновидности сетевых экранов

Сетевые экраны подразделяются на различные типы в зависимости от следующих характеристик:

обеспечивает ли экран соединение между одним узлом и сетью или между двумя или более различными сетями;

происходит ли контроль потока данных на сетевом уровне или более высоких уровнях модели OSI;

отслеживаются ли состояния активных соединений или нет.

В зависимости от охвата контролируемых потоков данных сетевые экраны делятся на:

традиционный сетевой (или межсетевой) экран — программа (или неотъемлемая часть операционной системы) на шлюзе (сервере передающем трафик между сетями) или аппаратное решение, контролирующие входящие и исходящие потоки данных между подключенными сетями.

персональный сетевой экран — программа, установленная на пользовательском компьютере и предназначенная для защиты от несанкционированного доступа только этого компьютера.

Вырожденный случай — использование традиционного сетевого экрана сервером, для ограничения доступа к собственным ресурсам.

В зависимости от уровня, на котором происходит контроль доступа, существует разделение на сетевые экраны, работающие на:

сетевом уровне, когда фильтрация происходит на основе адресов отправителя и получателя пакетов, номеров портов транспортного уровня модели OSI и статических правил, заданных администратором;

сеансовом уровне (также известные как stateful) — отслеживающие сеансы между приложениями, не пропускающие пакеты нарушающих спецификации TCP/IP, часто используемых в злонамеренных операциях — сканировании ресурсов, взломах через неправильные реализации TCP/IP, обрыв/замедление соединений, инъекция данных.

уровне приложений, фильтрация на основании анализа данных приложения, передаваемых внутри пакета. Такие типы экранов позволяют блокировать передачу нежелательной и потенциально опасной информации, на основании политик и настроек.

Некоторые решения, относимые к сетевым экранам уровня приложения, представляют собой прокси-серверы с некоторыми возможностями сетевого экрана, реализуя прозрачные прокси-серверы, со специализацией по протоколам. Возможности прокси-сервера и многопротокольная специализация делают фильтрацию значительно более гибкой, чем на классических сетевых экранах, но такие приложения имеют все недостатки прокси-серверов (например, анонимизация трафика).

.

Прокси-сервер (от англ. proxy — «представитель, уполномоченный») — служба в компьютерных сетях, позволяющая клиентам выполнять косвенные запросы к другим сетевым службам. Сначала клиент подключается к прокси-серверу и запрашивает какой-либо ресурс (например, e-mail), расположенный на другом сервере. Затем прокси-сервер либо подключается к указанному серверу и получает ресурс у него, либо возвращает ресурс из собственного кэша (в случаях, если прокси имеет свой кэш). В некоторых случаях запрос клиента или ответ сервера может быть изменён прокси-сервером в определённых целях. Также прокси-сервер позволяет защищать клиентский компьютер от некоторых сетевых атак и помогает сохранять анонимность клиента.

Использование

Чаще всего прокси-серверы применяются для следующих целей:

Обеспечение доступа с компьютеров локальной сети в Интернет.

Кэширование данных: если часто происходят обращения к одним и тем же внешним ресурсам, то можно держать их копию на прокси-сервере и выдавать по запросу, снижая тем самым нагрузку на канал во внешнюю сеть и ускоряя получение клиентом запрошенной информации.

Сжатие данных: прокси-сервер загружает информацию из Интернета и передаёт информацию конечному пользователю в сжатом виде. Такие прокси-серверы используются в основном с целью экономии внешнего трафика.

Защита локальной сети от внешнего доступа: например, можно настроить прокси-сервер так, что локальные компьютеры будут обращаться к внешним ресурсам только через него, а внешние компьютеры не смогут обращаться к локальным вообще (они «видят» только прокси-сервер).

Ограничение доступа из локальной сети к внешней: например, можно запретить доступ к определённым веб-сайтам, ограничить использование интернета каким-то локальным пользователям, устанавливать квоты на трафик или полосу пропускания, фильтровать рекламу и вирусы.

Анонимизация доступа к различным ресурсам. Прокси-сервер может скрывать сведения об источнике запроса или пользователе. В таком случае целевой сервер видит лишь информацию о прокси-сервере, например, IP-адрес, но не имеет возможности определить истинный источник запроса. Существуют также искажающие прокси-серверы, которые передают целевому серверу ложную информацию об истинном пользователе.

Прокси-сервер, к которому может получить доступ любой пользователь сети интернет, называется открытым.

Виды прокси-серверов

Прозрачный прокси — схема связи, при которой трафик, или его часть, перенаправляется на прокси-сервер неявно (средствами маршрутизатора). При этом клиент может использовать все преимущества прокси-сервера без дополнительных настроек, но с другой стороны, не имеет выбора.

Обратный прокси — прокси-сервер, который в отличие от прямого, ретранслирует запросы клиентов из внешней сети на один или несколько серверов, логически расположенных во внутренней сети. Часто используется для балансировки сетевой нагрузки между несколькими веб-серверами и повышения их безопасности, играя при этом роль межсетевого экрана на прикладном уровне.

Squid — программный пакет, реализующий функцию кэширующего прокси-сервера для протоколов HTTP, FTP, Gopher и (в случае соответствующих настроек) HTTPS. Разработан сообществом как программа с открытым исходным кодом (распространяется в соответствии с GNU GPL). Все запросы выполняет как один неблокируемый процесс ввода/вывода.

Используется в UNIX-like системах и в ОС семейства Windows NT. Имеет возможность взаимодействия с Active Directory Windows Server путём аутентификации через LDAP, что позволяет использовать разграничения доступа к интернет ресурсам пользователей, которые имеют учётные записи на Windows Server, также позволяет организовать «нарезку» интернет трафика для различных пользователей.

Сервер Squid развивается в течение уже многих лет. Обеспечивает совместимость с большинством важнейших протоколов Интернета, а также с операционными системами.

Описание архитектуры

Списки контроля доступа

Для контроля доступа к ресурсам и определения ряда действий используются списки контроля доступа[2] (англ. access control list, acl). Каждый ACL может состоять из нескольких критериев (но только одного типа):

адрес (сеть) источника запроса, цели запроса

имя (доменное имя) источника запроса, имя цели запроса

части URL запроса

протокол

порт (получателя, отправителя, самого squid’а)

метод (PUT или GET) при передаче данных по HTTP

браузер (User-agent)

ident (запрос к рабочей станции)

Номер автономной системы отправителя/получателя (не для всех случаев)

Авторизация на прокси-сервере (см. ниже)

Номер соединения (чаще всего используется для ограничения количества соединений)

SNMP

сертификаты пользователя

параметры запроса

внешние обработчики

Идентификация

Squid поддерживает несколько видов идентификации пользователей:

По IP-адресу (или доменному имени узла)

По переданным реквизитам (логин/пароль)

По идентификатору пользовательского агента (браузера)

Для идентификации по логину/паролю возможно использовать:

Обычные логин/пароль

NTLM-авторизацию

Внешние программы авторизации (определяющие формат авторизации)

Squid имеет возможность переписывать запрашиваемые URL. Squid может быть сконфигурирован так, чтобы пропускать входящие URL через процесс редиректора выполняемого как внешний процесс (подобно dnsserver), который возвращает новый URL или пустую строку, обозначающую отсутствие изменений. Редиректор — не является стандартной частью пакета Squid. Редиректор предоставляет администратору контроль за передвижениями пользователей. Использование редиректора в сочетании с прозрачным проксированием дает простой, но эффективный контроль, над доступом к нежелательным ресурсам( например к развлекательным ресурсам и социальным сетям в корпоративной сети, порнографии). Программа-редиректор должна читать URL (один на строку) со стандартного входа и записывать измененные URL или пустые строки на стандартный выход. Нужно заметить, что программа-редиректор не может использовать буферизированный ввод/вывод. Squid дописывает дополнительную информацию после URL, которую редиректор может использовать для принятия решения.

SAMS (SQUID Account Management System) — программное средство для администрирования доступа пользователей к прокси-серверу Squid.

На данный момент SAMS настраивает работу редиректоров:

• Редиректор SAMS — редиректор, работающий напрямую с базами SAMS

• SquidGuard — очень мощный редиректор.

• Стандартный SQUID — простейший редиректор, описанный в документации к SQUID

Виды редикторов:

Редиректор SAMS

Написан специально для SAMS, напрямую использует информацию, содержащуюся в базе данных. Позволяет включить различное перенаправление запросов для пользователей (регулируется шаблонами пользователей). Редиректор SAMS обеспечивает:

• ограничение доступа пользователей к SQUID

• контроль времени доступа пользователей к SQUID

• ограничение доступа пользователей к запрещенным ресурсам (или доступ пользователей только к разрешенным ресурсам)

• перенаправление запросов пользователей к баннерам, счетчикам и т. п.

Редиректор SquidGuard

Мощный редиректор с большими возможностями. В состав редиректора входят списки баннерных, порно и пр. доменов. SAMS добавляет в файл конфигурации SquidGuard squidguard.conf настройки на списки запрещенных доменов и перенаправления доступа SAMS. Настройки на списки, идущие с SquidGuard не изменяются и не удаляются. При использовании редиректора SquidGuard в файл squid.conf заносятся acl, разрешающие доступ всех пользователей к SQUID. Ограничение доступа пользователей организовано средствами редиректора.

Стандартный SQUID

Этот редиректор описан в документации на SQUID. Написан на perl. Редиректор создается после подачи команды на реконфигурирование SQUID, на основе списков перенаправления запросов. Быстрый и легкий редиректор, но не различает пользователей. При использовании этого редиректора, ограничение доступа пользователей по спискам запрета доступа организовано с использованием ACL SQUID. При использовании редиректора SQUID или если редиректор не используется вовсе, то в существует возможность — при отключении пользователей за превышение трафика у них остается доступ к URL и IP адресам, прописанным в списке «Локальные домены».

Ограничение максимальной скорости соединения

Ограничение максимальной скорости получения пользователем (пользователями) в squid реализовано с помощью механизма англ. delay pools (дословно — «пулы задержки»). Механизм ограничения скорости работает по принципу бассейна (откуда и название pool (бассейн)), в который «втекает» и «вытекает» информация. Отдельные конфигурируемые подобным образом области памяти называются англ. bucket (ведро). У ведра есть параметры: «ёмкость», «скорость наполнения». Если пользователь (пользователи) получают информацию на скорости ниже, чем «скорость наполнения», то ведро всегда полно. Если пользователь кратковременно поднимает скорость получения информации выше скорости наполнения, то до момента, пока ведро не пусто, он не ограничивается по скорости, как только ведро становится пустым, клиент получает информацию со скоростью наполнения ведра. В случае наличия групповых и индивидуальных ведёр, они включаются последовательно.

Существует три типа (класса) delay pools[3]:

Единое ведро (англ. aggregate bucket, class 1) ограничение на общую потребляемую полосу для всей группы. (параметры: ёмкость бассейна, скорость наполнения).

Единое ведро с автоматическим формированием индивидуальных вёдер (англ. single aggregate bucket as well as an "individual" bucket, class 2). Индивидуальные вёдра формируются из битов IP-адреса (c 25 по 32).

Единое ведро, сетевые вёдра и индивидуальные вёдра (англ. single aggregate bucket as well as a "network" bucket and a "individual" bucket, class 3). Сетевое ведро формируется по битам 17-24 IP-адреса.

Для каждого ведра указываются два параметра: ёмкость и скорость наполнения. −1 означает «без ограничения».

Попадание пользователей в то или иное ведро определяется списками доступа к вёдрам, они просматриваются в порядке упоминания в файле конфигурации до первого совпадения. Пользователи, не попадающие ни в одно из вёдер, в скорости не ограничиваются.

Обратное кэширование

Одной из особенностей squid является возможность работать в режиме «обратного прокси» («reverse proxy»), также известного как «ускоритель» («HTTP accelerator»). В этом случае вместо кэширования запросов нескольких пользователей к множеству сайтов, кешируются запросы множества пользователей к нескольким сайтам. В этом режиме принятый запрос проверяется на «динамичность» (нужно ли каждый раз обрабатывать запрос с нуля) и «возраст» (актуальны ли ещё данные). Если данные ещё актуальны и не поменялись, то запрос не передаётся серверу, а отдаётся из кеша squid’а. Таким образом существенно снижается нагрузка на серверы (например, в Википедии запросы к страницам кешируются, так как от просмотра их содержимое не меняется, при этом нагрузка на серверы существенно меньше — обработка запроса к кешу много проще, чем обработка запроса к базе данных SQL, обработка вики-разметки и формирование веб-страницы).

Кроме того, «обратный прокси» способен распределять запросы между несколькими серверами, балансируя нагрузку и/или обеспечивая отказоустойчивость, то есть фактически предоставляет функциональность, аналогичную кластеру.

Режим прозрачного прокси-сервера

Основная статья: прозрачный прокси

В сочетании с некоторыми межсетевыми экранами и маршрутизаторами squid может работать в режиме прозрачного прокси (англ. transparent proxy). В этом режиме маршрутизатор вместо того, чтобы сразу пересылать HTTP-запросы пользователя HTTP-серверу в Интернете, перенаправляет их прокси-серверу, который может работать как на отдельном хосте, так и на самом маршрутизаторе. Прокси-сервер обрабатывает запрос (с возможной отдачей содержимого из кеша), это содержимое направляется к запросившему пользователю, для которого оно выглядит как «ответ» сервера, к которому адресовался запрос. Таким образом, пользователь может даже не знать, что все запросы и ответы прошли через прокси-сервер.

При таком подходе проксирования аутентификация не предусмотрена, так как прозрачность проксирования это и подразумевает.

Достоинства

Администратору прозрачного прокси-сервера не нужно настраивать каждую клиентскую машину для работы с прокси.

Недостатки

В режиме прозрачности не проксируются FTP- и HTTPS-запросы

3proxy — бесплатный кроссплатформенный прокси сервер. Основными отличительными особенностями являются небольшой размер (несколько сотен килобайт) и переносимость (есть версии для x86 и x64 платформ). Программа не имеет графического интерфейса, её настройка производится путём написания конфигурационного файла. Существует возможность запуска программы как в консольном режиме, так и в фоновом режиме в виде службы.

Основные возможности

Поддержка протоколов HTTP, HTTPS, SOCKS, POP3, SMTP;

Переадресация TCP и UDP трафика (англ. Port mapping);

Объединение прокси серверов в цепочку (англ. Proxy chaining);

Ограничение пропускной способности канала;

Контроль доступа (ACL);

Ведение журналов;

Недостатки

Отсутствие поддержки кеширования веб-страниц;

Сложность настройки;

Последнее время некоторые антивирусные пакеты некорректно распознают 3proxy как вирус (троянский конь). Разработчики антивирусов обычно мотивируют это тем, что «хакеры» без ведома пользователей иногда устанавливают его на компьютеры пользователей и используют для рассылки спама. Тем не менее, программа сама по себе не является вредоносной.

Microsoft Internet Security and Acceleration Server (обычно сокращается до Microsoft ISA Server) — прокси-сервер для защиты сети от атак извне, а также контроля интернет-трафика.

Пришел на смену прокси-серверу Proxy 2.0 от Microsoft. Позволяет организовать защиту локальной сети от вмешательств из сети Интернет и безопасно публиковать различные виды серверов, дает возможность распределять доступ пользователей локальной сети к ресурсам Интернет. Оснащен средствами для анализа посещаемых ресурсов, учета трафика, а также защиты против атак из сети Интернет. Имеет различные виды аутентификации и авторизации, в том числе поддерживает аутентификацию Active Directory. Поддерживает как рабочие группы, так и домены Windows NT. Имеет множество плагинов для отслеживания исходящего и входящего трафика.

Не совместим с Windows Server 2008 и Windows Server 2008 R2.

Веб-прокси (англ. «web-based proxy») — это прокси-сервер и анонимайзер особого вида, представляющий собой веб-приложение (чаще всего PHP или Perl скрипт) установленное на веб-сервере, выступающее в роли посредника для загрузки контента различных веб-сайтов.

Веб-прокси могут быть использованы для:

ускорения загрузки веб-сайтов;

тестирования онлайн сервисов;

обхода ограничений Администратора локальной сети на доступ к определенным адресам веб-сайтов;

сокрытия реального IP-адреса и анонимного доступа к веб-сайтам;

получения доступа к веб-сайтам закрытым для просмотра пользователей определенных стран;

и многих других целей.

Использование веб-прокси

Веб-прокси устанавливается на веб-сервере и обеспечивает возможность его пользователям загружать произвольные веб-ресурсы от IP адреса этого сервера, что и обеспечивает анонимность веб-сёрфинга. Для использования веб-прокси, необходимо ввести адрес интересующего ресурса и, при желании указав дополнительные настройки, активировать переход. К дополнительным настройкам относятся:

запрет/разрешение на обработку JavaScript;

использование Cookie;

кодирование веб-адреса и страницы;

замена или очистка заголовка;

и ряд других, зависящий от конкретного приложения.

При кодирование веб-адреса загружаемого ресурса применяются алгоритмы Base64 и Rot13. Для кодирования веб-страниц используется преобразование в HEX код всего текстового содержания веб-страницы. Применение этих методов позволяет получить доступ к страницам с запрещенным контентом и перейти по веб-адресу заблокированному Администратором. Административными методами противодействия использованию в локальных сетях веб-прокси могут стать: усиление фильтрации контента и блокировка адресов публичных веб-прокси.

Стоит отметить, что большинство хостеров запрещают устанавливать на дешевых тарифах виртуального хостинга веб-прокси по причине большой нагрузки на сервер и активного потребления входящего трафика. Несмотря на это ограничение, в Интернет широко распространено создание целых сетей веб-прокси, они представляют собой большое количество различных доменых имен, размещенных на одном или нескольких IP-адресах выделенных серверов. Такие сети используются их владельцами для получения прибыли от рекламы, размещаемой на этих ресурсах.

Популярные скрипты веб-прокси

CGIProxy

CGI прокси-сервер с поддержкой HTTP и FTP протоколов. CGIProxy обладает одной из полных и лучших среди аналогичных веб-прокси поддержек JavaScript, Flash и SSL (реализованной на OpenSSL и Net::SSLeay). Для своей работы веб-скрипт требует подключенный mod\_perl или Perl 5.6.1 и выше. Настройка веб-прокси производится в текстовом формате и обладает большой гибкостью конфигурирования. Система фильтрации CGIProxy позволяет блокировать не только пользователей веб-ресурса, но и веб-сайты запрашиваемые пользователями. Опции конфигурации позволяют ограничить типы загружаемого контента с целью сокращения нагрузки на веб-сервер, возможен переход в полностью текстовый режим изменением всего одного параметра настроек.

Glype Proxy

PHP прокси-сервер, который подобно PHProxy позволяет обходить внутрисетевые ограничения путем использования внешнего веб-сервера. Glype Proxy обладает улучшенной поддержкой JavaScript, фильтром для блокировки как определенных веб-сайтов, так и адресов с заданными ключевыми словами, еще одной полезной функцией Glype Proxy является встроенная система кэширования, которая экономит ресурсы сервера, тем самым повышает производительность веб-сайта при малых потреблениях ресурсов системы, даже при работе с видео, аудио и flash. Упрощенная установка скрипта на веб-сервер позволит без знания PHP уже после загрузки на сервер начать использование Glype Proxy, а поддержка «тем» оформления обеспечит возможность быстрого выбора веб-дизайна для веб-прокси.

PHProxy

PHP прокси-сервер (с поддержкой HTTP протокола), предоставляющий возможность обхода различных ограничений, установленных локальным прокси-сервером. Для примера, часто на университетских прокси-серверах закрыты наиболее популярные развлекательные веб-ресурсы, чтобы получить доступ к таким ресурсам, возможно использовать PHProxy. Для использования необходим внешний веб-сервер с поддержкой PHP, на котором установлен PHProxy.

Zelune

PHP прокси-сервер, который во многом отличается от аналогичных веб-скриптов. Основными отличительными чертами Zelune являются: необязательное, но возможное использование MySQL позволит администратору веб-прокси применить блокировку пользователей по странам. Встроенная система кэширования позволяет загружать ранее запрошенные веб-страницы из кэша веб-сервера, а не получать их заново с удаленных ресурсов. Полезной опцией для администраторов публичных веб-серверов будет возможность заменять блоки рекламы с запрошенных страниц на свои собственные, на данный момент поддерживаются рекламные блоки Google AdSense и Yahoo! Publisher Network. Панель администрирования Zelune обладает большим количеством настроек, которые позволят настроить веб-прокси в соответствии с индивидуальными требованиями владельца веб-сайта, контролировать размер кэша и блокировать доступ к веб-сайту по заданным IP адресам.

Проблемы использования публичных веб-прокси и их решение

Адрес веб-прокси не доступен

Прежде всего это может быть связано с невозможностью зайти на веб-прокси из-за того, что он тоже является заблокированным ресурсом. Заблокированным как внутри локальной сети Администратором, так и владельцем веб-прокси. Очевидное решение данной проблемы — попробовать другой ресурс.

Невозможность соединиться с желаемым ресурсом

С данной проблемой можно столкнуться в нескольких случаях:

Маленькая скорость соединения между веб-прокси и введенным адресом ресурса не позволяет обработать и загрузить его при помощи используемого веб-прокси.

Адрес введенного ресурса заблокирован на данном веб-прокси по каким-либо причинам.

Веб-прокси может быть заблокирован владельцем сайта, адрес которого необходимо загрузить.

Данная проблема, вне зависимости от случая может быть решена только путем использования другого адреса веб-прокси.

Фильтрация сильнее применяемых средств обхода

Невозможность просмотра из-за ограничений накладываемых на содержание (контент) веб-сайта. Решением данной проблемы может стать изменение настроек веб-прокси. Обычно, настройки скрыты и могут быть показаны после нажатие ссылки Options, находящейся в непосредственной близости от поля ввода адреса ресурса. Там можно установить различные параметры, наиболее полезной может оказать активация следующих опций: «Encode URL», «Encode Page», «Strip page title», «Use base64 encoding on the address», наличие той или иной опции зависит от типа используемого веб-прокси.

Некоторые функции веб-сайта недоступны

Многие динамические ресурсы представляют пользователям ряд функций, выполнение которых должно производиться непосредственно на компьютере пользователя. Так как, для доступа к этим ресурсам используется веб-прокси, то именно веб-прокси в данный момент является пользователем веб-сайта, доступ к которому необходимо получить. Поэтому из-за ряда технических проблем такие функции как авторизация, отправка сообщений, поиск и прочие могут оказаться недоступными. Решение данной проблемы тоже заключается в небольшой дополнительной настройке веб-прокси. Обычно эти настройки скрыты и могут быть показаны после нажатие ссылки Options, находящейся в непосредственной близости от поля ввода адреса ресурса. Там можно изменить различные параметры, для решения этой проблемы рекомендуется отключить (снять галочки) опции: «Remove Scripts», «Remove Objects» и проверить установлена ли галочка напротив пункта «Allow Cookies».

WebWrapper: web-proxy, сжимает трафик, кэширует, зашита о вирусов, отключение баннеров, рекламы и прочее.

Ремейлер (англ. remailer) — это сервер, получающий сообщение электронной почты и переправляющий его по адресу, указанному отправителем. В процессе переадресации вся информация об отправителе уничтожается, поэтому конечный получатель лишён возможности выяснить, кто является автором сообщения.[1] Некоторые из ремейлеров позволяют также шифровать письма и указывать фиктивный адрес отправителя, но большинство из них прямо указывают в заголовке, что электронное сообщение анонимно. В качестве ремейлеров могут выступать специализированные веб-сайты, открытые SMTP-серверы и анонимные сети вроде Mixminion.

Виды ремейлеров

Ремейлеры делятся на анонимные и псевдо-анонимные.

При использовании псевдо-анонимного ремейлера, его оператор знает адрес электронной почты, который необходим для получения ответа на письмо. Тайна связи полностью зависит от оператора, который может стать жертвой угроз, шантажа или социальной инженерии. Преимуществом псевдо-анонимных ремейлеров является их юзабилити, за которое пользователь расплачивается меньшей защищённостью.

Анонимные ремейлеры обеспечивают гораздо более высокую секретность, но при этом они и сложнее в использовании. Их операторы не могут знать, какие данные пересылаются через них, а поэтому нет гарантии своевременной доставки сообщения, которое может и вовсе затеряться.[2] В обмен на высокое время ожидания анонимные ремейлеры достаточно надёжно скрывают от посторонних глаз реальный адрес и содержимое сообщения.

Mixminion — стандарт реализации третьего типа протокола анонимной пересылки электронной почты. Mixminion может отсылать и принимать анонимные сообщения электронной почты.

Mixminion использует архитектуру «перемешанных сетей» (англ. Mix network) для предоставления очень высокой степени анонимности, а также для предотвращения прослушивания и других видов атак при пересылке сообщений. Серверы, названные «миксерами», которые запускают волонтёры, принимают сообщения, расшифровывают их, реформируют и передают в истинный пункт назначения. Каждое письмо электронной почты проходит через несколько серверов так, что ни один из серверов не может быть связан как отправитель для получателя.

Отсылая анонимные сообщения, Mixminion разбивает его на одинаковые по размеру участки, приводя пакеты в одинаковый вид и выбирая путь для отправления через смешанную сеть для каждого пакета. Программа шифрует каждый пакет публичным ключом для каждого сервера на его пути, один за другим. При отправлении пакета, Mixminion посылает его в первый «микс-сервер» на пути. Первый сервер расшифровывает пакет, читая, какой следующий сервер должен принять пакет, и передаёт его. В итоге, пакет прибывает к конечному «микс-серверу», который посылает его выбранному получателю. Так как ни один сервер не знает более чем одного смежного сервера, они не могут связать получателя с отправителем.

Mixminion поддерживает технологию «одноразовых блоков ответа» (англ. «Single-Use Reply Blocks», сокр. «SURB») для опознания анонимных адресатов. SURB кодирует половину пути к адресату, далее, каждый «микс-сервер» в очереди «разворачивает» один слой в пути, а после этого зашифровывает сообщение для получателя-адресата. Когда сообщение достигает адресата, он может расшифровать сообщение и прочитать, какой SURB был использован для его отправки, но отправитель не знает, какой получатель принял анонимное сообщение.