NAT и другие манипуляции адресами

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.

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

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

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

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

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

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/port mapping) позволяет перенаправлять запросы из Internet об определенных сервисах на соответствующие отдельные внутренние серверы.

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.

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.

Другие манипуляции адресами:

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).

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.

Экспериментальная комбинация IP aliasing и ICMP redirects известна как

directed ARP (RFC 1433). Это понятие перекликается с понятием directed broadcasts и проявляется при неправильном соотнесением 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.