# Атаки на протокол TCP и его участников.

Слабость TCP в том, что реализация протокола предполагает «честное» поведение всех участников сети. В результате злоумышленник может получить доступ к передаваемым данным, выдать себя за другую сторону, привести систему в нерабочее состояние.

1. **Passive scan** - пассивное сканирование портов SYNc-(SYNs-ACKc)-RSTc и SYNc-RSTs. При достаточно умном поведении сканера (например, сканирование с низкой скоростью или проверка лишь конкретных портов) детектировать пассивное сканирование невозможно, поскольку оно ничем не отличается от обычных попыток установить соединение.

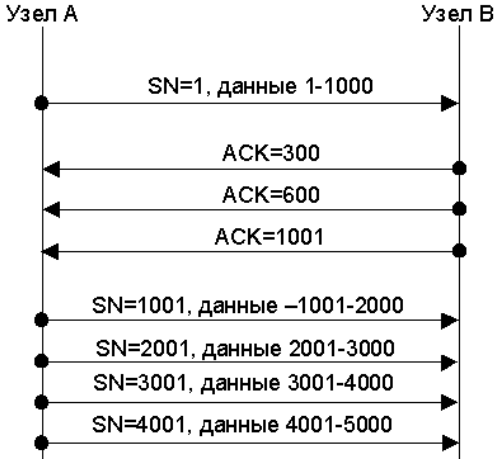
**Защита** - закрыть на firewall все сервисы, доступ к которым не требуется извне.

2. **TCP Reset –** если бит RST=1, то получатель должен немедленно прекратить использовать данное соединение.

   Атакующий используя IP-spoffing и сфальсифицированный RST-сегмент с номером SN, находящимся в рамках доступного окна, может разрывать чужие сессии. Например, сбросить TCP-соединение между BGP-соседями, чтобы каждый из соседей удалил маршруты, полученные от другого, и распространил информацию о недостижимости этих маршрутов другим своим соседям.

**Защита** – шифрование на уровне IP или BGP.

3. **Принуждение к ускорению/замедлению передачи** - злоумышленник отбирает себе ресурсы сервера или замедляет соединения прочих участников сети.




**Варианты реализации атаки:**
a) ложные дубликаты подтверждений;
b) преждевременные подтверждения;
**c) расщепление подтверждений:**

Пусть сервер A начинает медленный старт с B. Окно перегрузки cwnd=1, поэтому клиенту B высылается один полноразмерный сегмент (например, 1000).

Нарушитель B вместо 1 подтверждения о получении сегмента (ACK, SN=1001), имитирует получение сегмента по частям и высылает серию подтверждений (ACK, SN=300, 600 и 1001).

Это воздействует на алгоритм медленного старта и вынудит A необоснованно увеличить cwnd до 4 (отправить 4 сегмента вместо 2). На k шаге узел A будет отправлять не V=N*2^k байт, а V=N*M^k байт (где, V – скорость потока, N – размер сегмента, M - количество расщеплений).

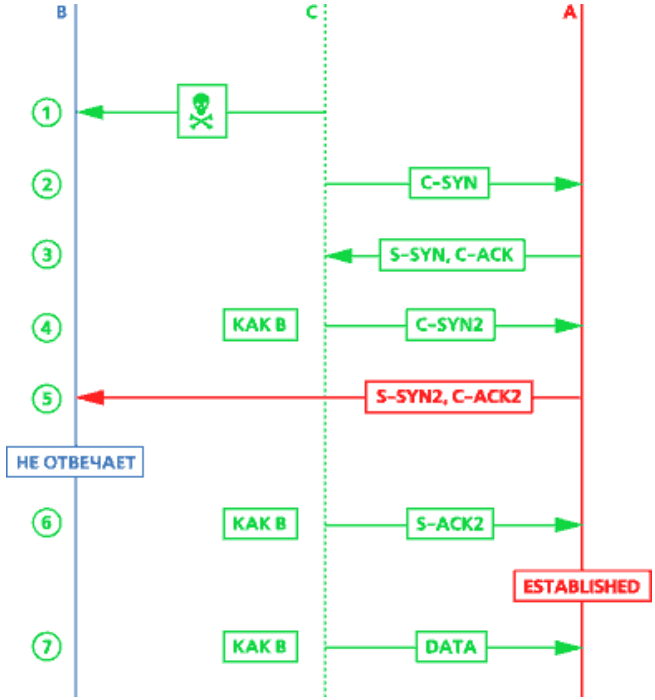
При агрессивной атаке (1000 подтверждений на сегмент) уже на 4 шаге V=1 TiB/sec. Скорость ограничена лишь возможностями сети и узлов. Другие узлы диагностируют состояние затора и уменьшают скорость передачи данных, фактически освобождая канал для злоумышленника.

**Защита** – пока нет, нужны изменения алгоритмов регулировки потока в реализации стека TCP.

**4. IP-spoofing + TCP sequence prediction** - предсказание TCP ISN**.**

Например, для захвата rlogin/rsh (доверительный удалённый логин) порядок атаки:

1. Через легитимный сервис (например web) провоцируем сервер (A) на TCP-связь (шлём SYN) от своего имени и узнаём алгоритм образования ISN;
2. Выводим жертву-клиента (B) из сети (например, SYN Flooding) на пару минут;
3. От имени клиента (IP-spoofing) формируем ложный запрос rlogin/rsh к серверу;
4. Клиент не сможет сделать RST на неожиданный SYN-ACK, т.к. выведен нами из строя на шаге 2;
5. Ответ мы не получим, его сервер отправит не нам, а настоящему клиенту, поэтому предсказываем диапазон возможных ISN-сервера и формируем один (или серию) ответов от имени клиента об успешном «тройном рукопожатии»;
6. Внутри этих ложных ответов уже будут содержаться telnet команды реконфигурации сервиса rlogin: **“# echo “*’ > ~/.rhosts”**, * - для включения доверия с любого узла.



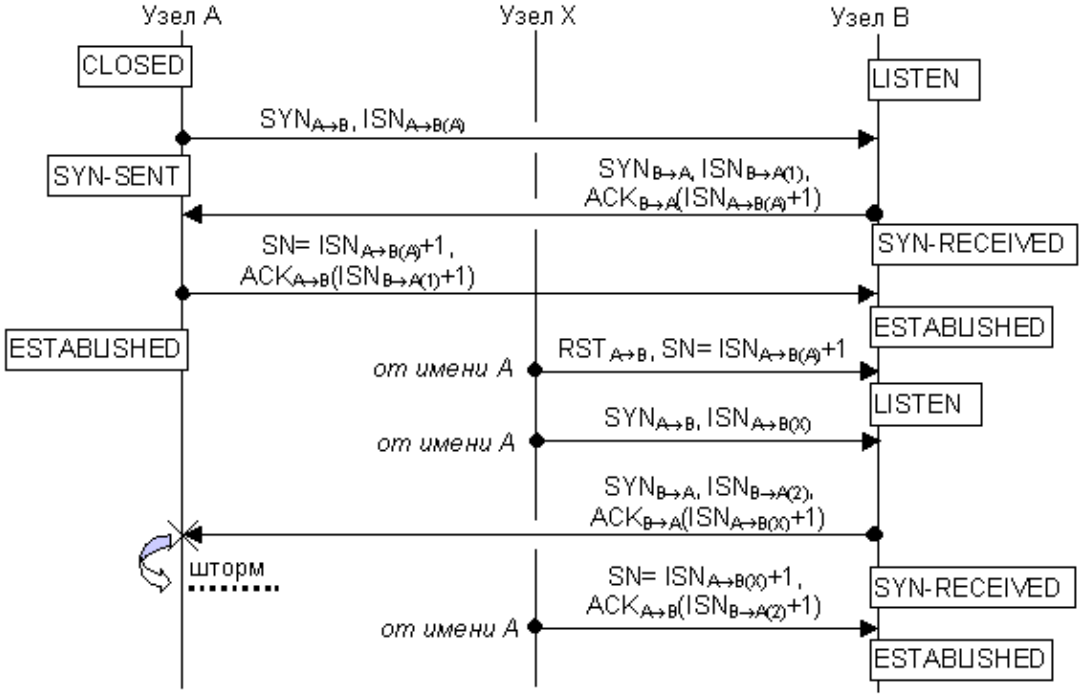
**Защита** – практически все перечисленные ниже компоненты защиты уже реализуются в сетях.

- Следует минимизировать доверие машин друг другу.
- Перейти на протокол ssh.
- Усложнить угадывание sequence number (ключевой элемент атаки). Например, можно увеличить скорость изменения sequence number на сервере или выбирать коэффициент увеличения sequence number случайно (желательно, используя для генерации случайных чисел криптографически стойкий алгоритм).
- Использовать рукопожатие с COOKIE.
- Шифрование TCP/IP-потока решает в общем случае проблему IP-spoofing.
- Настроить firewall для фильтрации пакетов, посланных нашей сетью наружу, но имеющих адреса, не принадлежащие нашему адресному пространству. Это защитит мир от подобных атак из вашей собственной сети.

5. **Sniffing** - атака заключаются в перехвате и анализе сетевого потока.

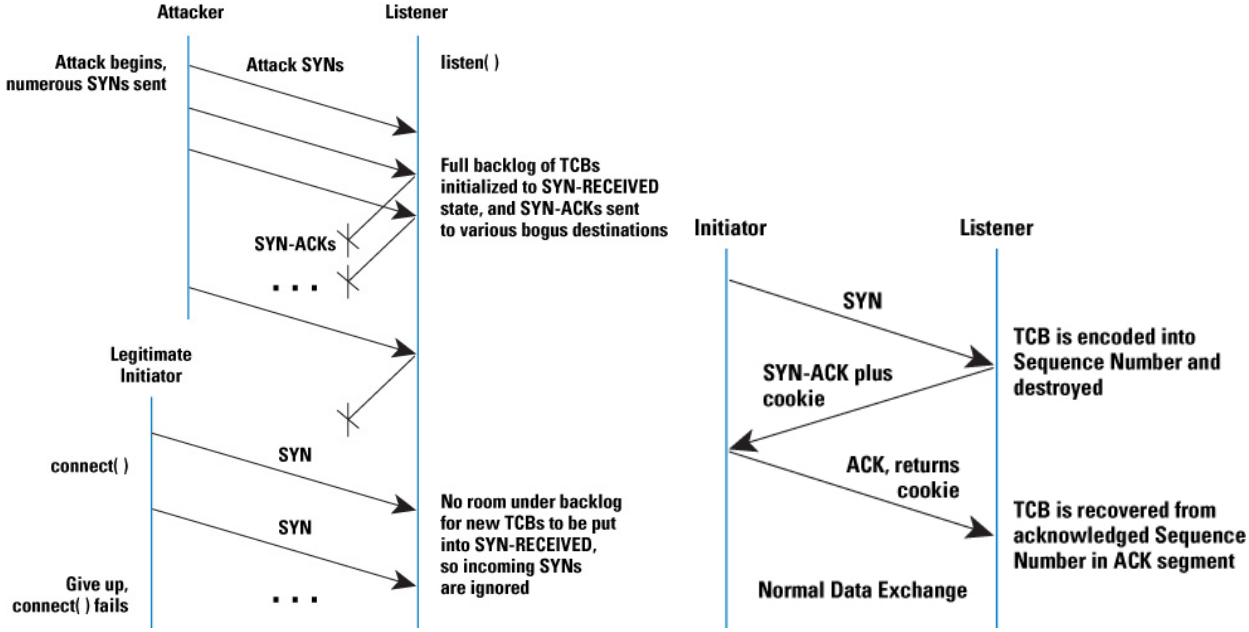
**Защита** - использование свитчей и шифрование.

6. **TCP Session Hijacking** – перехват TCP-сессии через десинхронизацию, в данном случае перехватывается весь сетевой поток **уже после установки соединения**. Далее сессия строится произвольным образом. Атакующему нужно быть на пути трафика и работать как MiM. Метод является комбинацией "подслушивания" и IP spoofing'а.




**Защита** – противодействие IP-spoofing и шифрование на уровне IP.

7. **SYN Flooding** – затопление полуоткрытыми сессиями, переполняющими очереди сервера, после чего сервер перестает отвечать на запросы легитимных клиентов. Зачастую достаточно 50-100 ложных сессий и сервер будет «тормозить».




**Защита** – частичная за счёт уменьшения таймаутов между SYNs-ACK-с и ACK-s; полная защита за счёт введения cookie (квитанции), см. рис. ниже или за счёт четырёхкратного рукопожатия с cookie, см. протокол SCTP.

## 8. TCP Full Connection Flooding.

Атаку SYN flood удерживает большое число соединений на атакуемом узле в состоянии SYN-RECEIVED. Но, состояния множество ESTABLISHED и FIN-WAIT-1 также вызывают DoS.

С сервером-жертвой создаётся множество легитимных полных троекратных TCP соединений с одного узла (до 65 000 штук по количеству портов для сокета), что истощит очередь сессий.

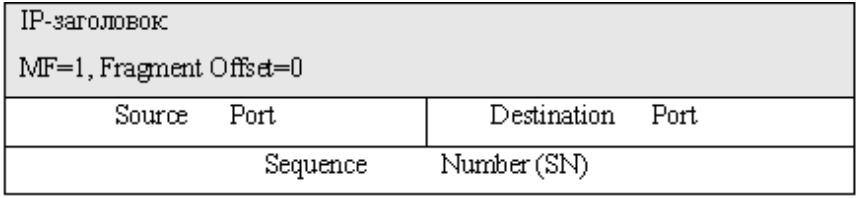
Дополнительно можно сформировать вредоносную нагрузку, разработанную под конкретный сервис, например можно:

- запросить загрузку с сервера какого-нибудь большого файла, сервер загрузит первую часть этого файла в стек TCP для отправки, используя при этом буферы в памяти ядра системы. Вернуть эти буферы система не сможет пока нападающий не подтвердит, что данные им получены. Закончится доступная память на подвергающейся атаке системе.
- провести манипуляции с размером окна TCP – установить в 0.
- запросить создание динамической страницы, для истощения процессора **(HTTP Flood**).
- подключить фрагментацию IP - отсылать множество пакетов больших размеров, в каждом из которых содержится один пропущенный фрагмент.
- подключить сегментацию TCP - создать «дыру» в TCP потоке, отсылая данные из конца текущего окна, в середине которого ничего не содержится. Система зарезервирует эти данные до тех пор, пока вы не решите переслать недостающие пакеты.

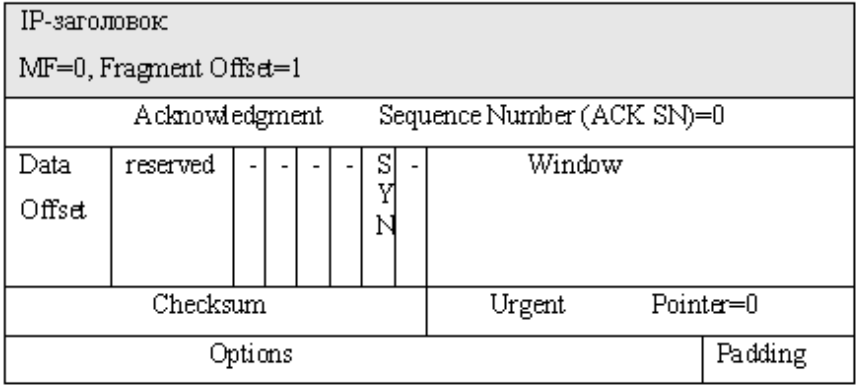
**Защиты нет!** – все варианты атаки эксплуатируют одну, но, фундаментальную слабость TCP - **неконтролируемое число соединений к TCP-IP сокету жертвы**. Можно ослабить воздействие атак используя специализированные системы защиты: FireProof фирмы Radware или TrafficMaster Enforcer фирмы Mazu Networks. Стоимость может достигать 500 000 $.

**9. Tiny Fragment Attack** – атака крошечными фрагментами.

Если на вход фильтрующего маршрутизатора поступает фрагментированная дейтаграмма, маршрутизатор производит досмотр только первого фрагмента дейтаграммы (определяется по Fragment Offset=0 в IP). Если фрагмент не удовлетворяет условиям, он уничтожается. Остальные фрагменты пропускаются, без затрат вычислительных ресурсов фильтра, т.к. без первого фрагмента дейтаграмма все равно не может быть собрана на узле назначения.




При конфигурировании фильтра перед сетевым администратором часто стоит задача: разрешить соединения с TCP-сервисами Интернет, инициируемые компьютерами внутренней сети, но запретить установление соединений внутренних компьютеров с внешними по инициативе последних. Для решения поставленной задачи фильтр конфигурируется на запрет пропуска TCP-сегментов, поступающих из внешней сети и имеющих установленный бит SYN



в отсутствии бита ACK; сегменты без этого бита беспрепятственно пропускаются в охраняемую сеть, поскольку они могут относиться к соединению, уже установленному ранее по инициативе внутреннего компьютера.

Рассмотрим, как злоумышленник может использовать фрагментацию, чтобы обойти это ограничение, то есть, передать SYN-сегмент из внешней сети во внутреннюю.

**Защита** - фильтрующему маршрутизатору не следует инспектировать содержимое первых фрагментов датаграмм — это было бы равносильно сборке датаграмм на промежуточном узле, что быстро поглотит все вычислительные ресурсы маршрутизатора. Достаточно реализовать один из двух следующих подходов:

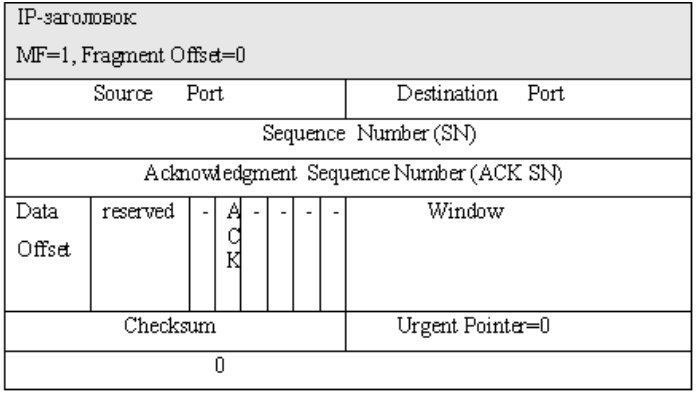
1) не пропускать датаграммы с Fragment Offset=0 и Protocol=6 (TCP), размер поля данных которых меньше определенной величины, достаточной, чтобы вместить все «интересные поля» (например, 20);

2) не пропускать датаграммы с Fragment Offset=1 и Protocol=6 (TCP): наличие такой датаграммы означает, что TCP-сегмент был фрагментирован с целью скрыть определенные поля заголовка и что где-то существует первый фрагмент с 8 октетами данных. Несмотря на то, что в данном случае первый фрагмент будет пропущен, узел назначения не сможет собрать датаграмму, так как фильтр уничтожил второй фрагмент.

Т.к. в реальной жизни никогда не придется фрагментировать датаграмму до минимальной величины, риск потерять легальные датаграммы, применив предложенные выше методы фильтрации, равен нулю.

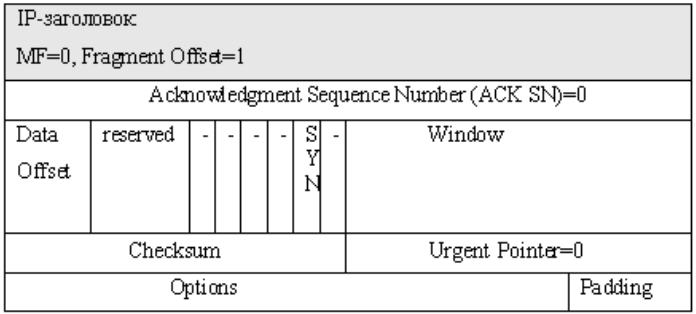
**10. Overlapping Fragment Attack** – атака накладывающимися фрагментами.

Рассмотрим пример датаграммы, несущей TCP-сегмент и состоящей из двух фрагментов. В поле данных первого фрагмента находится полный TCP-заголовок, без опций, дополненный нулями до размера, кратного восьми октетам.




В поле данных второго фрагмента — часть другого TCP-заголовка, начиная с девятого по порядку октета, в котором установлен флаг SYN.

**Защита.** Если для защиты от Tiny Fragment Attack применяется подход 1) из описанных выше (инспекция первого фрагмента датаграммы), то с помощью накладывающихся фрагментов злоумышленник может обойти эту защиту.




Маршрутизатор, применяющий второй подход, будет успешно противостоять Tiny Fragment Attack с накладывающимися фрагментами.

## 11. Защита в целом.

Системный администратор, исходя из политики сетевой безопасности в своей организации, и имея четкое представление о возможных инцидентах и их последствиях, должен определить, какие меры являются необходимыми и приемлемыми для его сети.

### 11.1. Фильтрация на маршрутизаторе

Фильтры на маршрутизаторе, соединяющем сеть предприятия с Интернетом, применяются для запрета пропуска датаграмм, которые могут быть использованы для атак как на сеть организации из Интернета, так и на внешние сети злоумышленником, находящимся внутри организации.

1. Запретить пропуск датаграмм с широковещательным адресом назначения между сетью организации и Интернетом.
2. Запретить пропуск датаграмм, направленных из внутренней сети (сети организации) в Интернет, но имеющих внешний адрес отправителя.
3. Запретить пропуск датаграмм, прибывающих из Интернета, но имеющих внутренний адрес отправителя.
4. Запретить пропуск датаграмм с опцией «Source Route» и, если они не используются для групповой рассылки, инкапсулированных датаграмм (IP-датаграмма внутри IP-датаграммы).
5. Запретить пропуск датаграмм с ICMP-сообщениями между сетью организации и Интернетом, кроме необходимых (Destination Unreachable: Datagram Too Big — для алгоритма Path MTU Discovery; также Echo, Echo Reply, Destination Unreachable: Network

Unreachable, Destination Unreachable: Host Unreachable, TTL exceeded).

6. На сервере доступа клиентов по коммутируемой линии — разрешить пропуск датаграмм, направленных только с или на IP-адрес, назначенный клиенту.
7. Запретить пропуск датаграмм с UDP-сообщениями, направленными с или на порты echo и chargen, либо на все порты, кроме используемых (часто используется только порт 53 для службы DNS).
8. Использование TCP Intercept для защиты от атак SYN flood.
9. Фильтрация TCP-сегментов выполняется в соответствии с политикой безопасности: разрешаются все сервисы, кроме запрещенных, или запрещаются все сервисы, кроме разрешенных (описывая каждый прикладной сервис в главе 3, мы будем обсуждать вопросы фильтрации сегментов применительно к сервису). Если во внутренней сети нет хостов, к которым предполагается доступ из Интернета, но разрешен доступ внутренних хостов в Интернет, то следует запретить пропуск TCP SYN-сегментов, не имеющих флага ACK, из Интернета во внутреннюю сеть1, а также запретить пропуск датаграмм с Fragment Offset=1 и Protocol=6 (TCP).

Отметим, что более безопасным и управляемым решением, чем фильтрация того или иного TCP-трафика следующего от или к компьютеру пользователя, является работа пользователей через прокси-серверы.

Преимущества этого решения следующие.

Прокси-сервер находится под контролем администратора предприятия, что позволяет реализовывать различные политики для дифференцированного управления доступом пользователей к сервисам и ресурсам Интернета, фильтрации передаваемых данных (защита

от вирусов, цензура и т.п.), кэширования (там, где это применимо).

С точки зрения Интернета от имени всех пользовательских хостов предприятия действует один прокси-сервер, то есть имеется только один потенциальный объект для атаки из Интернета, а безопасность одного прокси-сервера, управляемого профессионалом, легче обеспечить, чем безопасность множества пользовательских компьютеров.

### 11.2. Анализ сетевого трафика.

Анализ сетевого трафика проводится для обнаружения атак, предпринятых злоумышленниками, находящимися как в сети организации, так и в Интернете.

1. Сохранять и анализировать статистику работы маршрутизаторов, особенно — частоту срабатывания фильтров.
2. Применять специализированное программное обеспечение для анализа трафика для выявления выполняемых атак (NIDS — Network Intrusion Detection System). Выявлять узлы, занимающие ненормально большую долю полосы пропускания, и другие аномалии в поведении сети.
3. Применять программы типа arpwatch для выявления узлов, использующих нелегальные IP- или MAC-адреса.

Применять программы типа Antisniff для выявления узлов, находящихся в режиме прослушивания сети.

### 11.3. Защита маршрутизатора.

Мероприятия по защите маршрутизатора проводятся с целью предотвращения атак, направленных на нарушение схему маршрутизации датаграмм или на захват маршрутизатора злоумышленником.

1. Использовать аутентификацию сообщений протоколов маршрутизации с помощью алгоритма MD5.
2. Осуществлять фильтрацию маршрутов, объявляемых сетями-клиентами, провайдером или другими автономными системами. Фильтрация выполняется в соответствии с маршрутной политикой организации; маршруты, не соответствующие политике, игнорируются.
3. Использовать на маршрутизаторе, а также на коммутаторах статическую ARP-таблицу узлов сети организации.
4. Отключить на маршрутизаторе все ненужные сервисы (особенно так называемые «диагностические» или «малые» сервисы TCP: echo, chargen, daytime, discard, и UDP: echo, chargen, discard).
5. Ограничить доступ к маршрутизатору консолью или выделенной рабочей станцией администратора, использовать парольную защиту; не использовать telnet для доступа к маршрутизатору в сети, которая может быть прослушана.
6. Использовать последние версии и обновления программного обеспечения, следить за бюллетенями по безопасности, выпускаемыми производителем.

## 11.4. Защита хоста.

Мероприятия по защите хоста проводятся для предотвращения атак, цель которых — перехват данных, отказ в обслуживании, или проникновение злоумышленника в операционную систему.

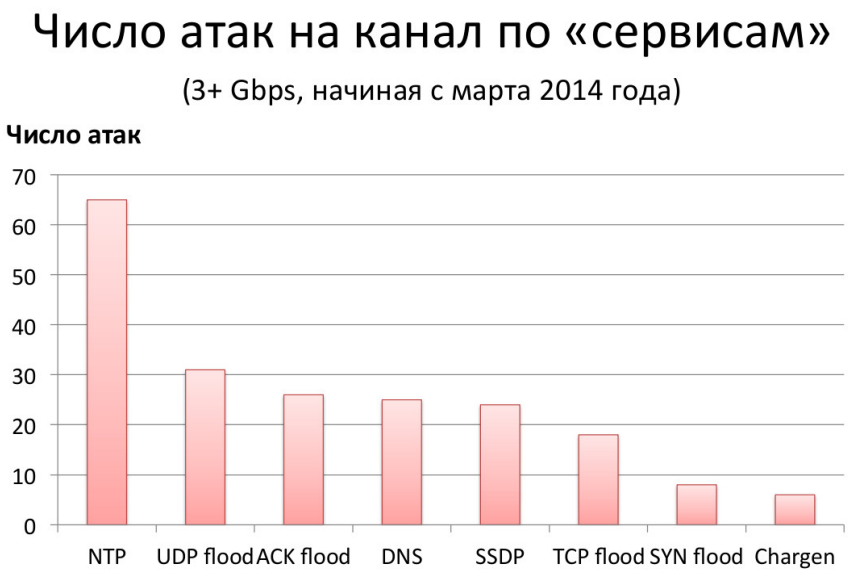
1. Запретить обработку ICMP Echo-запросов, направленных на широковещательный адрес.
2. Запретить обработку ICMP-сообщений Redirect, Address Mask Reply, Router Advertisement, Source Quench.
3. Если хосты локальной сети конфигурируются динамически сервером DHCP, использовать на DHCP-сервере таблицу соответствия MAC- и IP-адресов и выдавать хостам заранее определенные IP-адреса.
4. Отключить все ненужные сервисы TCP и UDP (читай: отключить все сервисы, кроме явно необходимых). Под отключением сервиса мы понимаем перевод соответствующего порта из состояния LISTEN в CLOSED.
5. Если входящие соединения обслуживаются супердемоном inetd, то использовать оболочки TCP wrappers или заменить inetd на супердемон типа xinetd или tcpserver, позволяющий устанавливать максимальное число одновременных соединений, список разрешенных адресов клиентов, выполнять проверку легальности адреса через DNS и регистрировать соединения в лог-файле.
6. Использовать программу типа tcplog, позволяющую отследить попытки скрытного сканирования (например, полуоткрытыми соединениями).
7. Использовать статическую ARP-таблицу узлов локальной сети.
8. Применять средства безопасности используемых на хосте прикладных сервисов.

9. Использовать последние версии и обновления программного обеспечения, следить за бюллетенями по безопасности, выпускаемыми производителем.

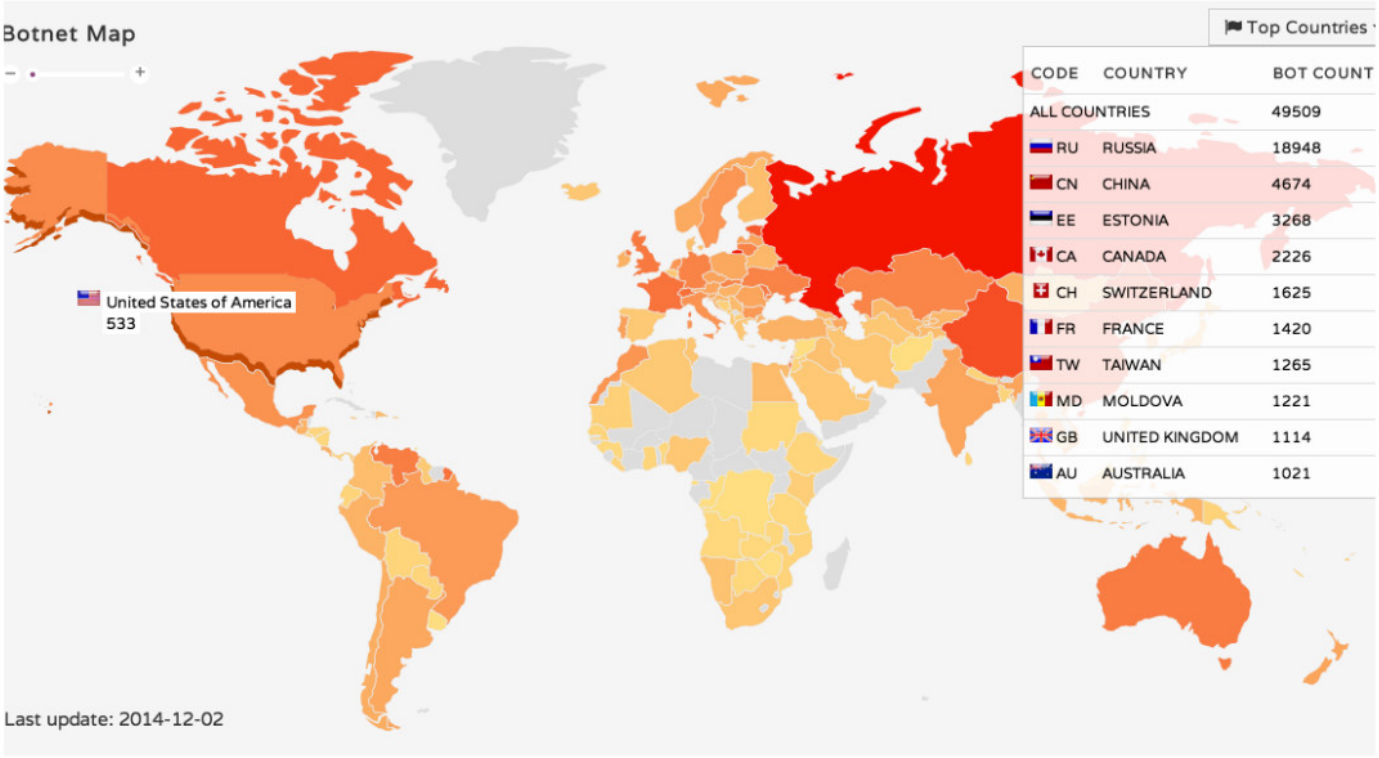
### 11.5. Превентивное сканирование.

Администратор сети должен знать и использовать методы и инструменты злоумышленника и проводить превентивное сканирование сети организации для обнаружения слабых мест в безопасности до того, как это сделает злоумышленник. Для этой цели имеется также специальное программное обеспечение — сканеры безопасности, network security scanners, типа Nessus.

# 12. Статистика угроз.




Кол-во атак, из-за криво настроенной службы NTP, впереди планеты всей…. Хотя настройка службы NTP занимает не более 5 минут.




Эстония и Китай рядом по количеству ботов…. а насколько велика разница по количеству народа в Китае и в Эстонии?

# 13. Бесплатное программное обеспечение.

Purdue University CERIAS Security Archive. Разные программы.
Libnet. Библиотека для формирования кадров и дaftаграмм произвольного формата.
Libpcap. Библиотека для извлечения пакетов из сети (используется программой tcpdump).
OpenSSL. Библиотека для поддержки SSL.
Tcpdump. Программа для прослушивания сети (sniffer).
Tcptrace. Программа для анализа и конвертирования дамп-файлов, записанных различными программами прослушивания.
AntiSniff. Программа для обнаружения в сети узлов, находящихся в режиме прослушивания.
Nmap. Сканер сети.
Nessus. Сканер для обнаружения проблем с безопасностью в сети. Использует nmap.
Nemesis. Программа для формирования сообщений различных протоколов.
Arpwatch. Программа для обнаружения несоответствия между IP- и MAC-адресами в сети.
Snort. NIDS (система обнаружения атак в сети).
SOCKS. Универсальный прокси-сервер для TCP-сервисов (reference implementation).
TCP wrappers. Программа мониторинга и фильтрации запросов на установление TCP-соединений через супердемон inetd.
Tcplogd. Программа для обнаружения попыток сканирования хоста.
Tripwire. Утилита для отслеживания изменений в файловой системе.
Xinetd. Супердемон для замены inetd. Осуществляет фильтрацию соединений и уменьшает риск DoS атак.
Tcpserver. Еще один вариант замены супердемона inetd.
SSH, OpenSSH, FreeSSH. Реализации протокола SSH - защищенного варианта Telnet + FTP.
Kerberos. Система распределенной аутентификации в сети.
Swatch. Утилита слежения за лог-файлами протоколов.

## 14. Сайты.

http://www.cert.org/
http://www.securityfocus.com
http://www.sans.org/
http://www.cerias.purdue.edu/
http://www.iss.net/
http://www.packetfactory.net/
http://www.insecure.org/
http://blacksun.box.sk/tutorials.html
http://antionline.com/

## 15. Сайты на русском языке.

http://www.void.ru/
http://www.bugtraq.ru/
http://www.hackzone.ru/
http://www.security.nnov.ru/
http://www.xakep.ru/