DDoS-атаки: почему они возможны, и как их предотвращать

Алексей Семеняка
Qrator Labs
as@qrator.net

ENOG Tutorial

Кому адресована презентация?

- Системным администраторам и сетевым инженерам интернет-ресурсов
  - *Там, где они есть* ☹
- Системным администраторам хостинговых компаний
- Сетевым инженерам датацентров
- Сетевым инженерам операторов связи

Зачем она нужна?

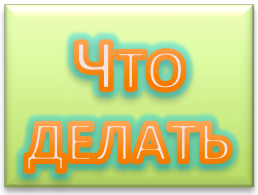
- Угроза атак вида DoS/DDoS по-прежнему недооценена
- Существует множество принципиально разных технологических реализаций таких атак
- Полностью предотвратить угрозу очень сложно
- Просто «купить решение» и положиться на него вообще невозможно

Прогресс может быть достигнут только в результате коллективных усилий.

Некоторые жертвы DDoS'еров:

Наши задачи

- Выяснить, что же такое DoS/DDoS-атаки
- Рассмотреть их классификацию
  - Чтобы уметь их правильно распознать
- Дать представление о том, как их проводят
  - **И как не стать невольным помощником злоумышленников**
- Обсудить способы защиты себя, своих клиентов **и экосистемы в целом**.
  - Сформулировать набор рекомендаций для каждого игрока индустрии
  - Но серебряной пули не будет
  - Слайды с рекомендациями отмечены вот таким значком:



Чего мы НЕ будем делать

- Обсуждать вред от DoS/DDoS-атак
- Обсуждать коммерческие решения по защите от DDoS
- Сравнивать параметры различных моделей оборудования
- Пытаться дать рецепты на все случаи жизни
  - В том числе в виде «правильных» записей в разнообразных конфигурационных файлах

Что такое DoS/DDoS-атака?

DoS = Denial of Service, отказ в обслуживании

DDoS = Distributed Denial of Service,
распределенный отказ в обслуживании

В обоих случаях речь идет про **полную или частичную потерю доступности ресурса «извне»** без нарушения его внутренней структуры («взлома»).

Обычно:

DoS – атака на отдельную уязвимость
(ping of death, INVITE of death, **route leaks**)

DDoS – атака на исчерпание каких-либо ресурсов
(SYN-flood, amp-атаки)

Традиционные схемы проведения атак

DoS-атака:

DDoS-атака:

Какие бывают DDoS-атаки?

- А зачем вообще их классицицировать?
  - Чтобы уметь эффективно противодействовать
- Следовательно, атакам одного класса должны соответствовать однотипные подходы в противодействии
- Модель OSI? Слишком много пунктов

Классификация

Поэтому будем исходить из того, на что именно нацелен удар:

1. Канальная емкость
2. Инфраструктура
3. Операционная система (сетевой стек)
4. Приложение
5. Деньги жертвы

Атаки на канальную ёмкость

Атаки на канальную ёмкость

- Они же «атаки на полосу», они же «атаки на канал», они же «volumetric attacks»
- Всегда DDoS, то есть атака производится из многих точек
- Ранее основным инструментом для ее создания были **ботнеты**, теперь – **амплификаторы трафика**.

Атаки на канальную ёмкость

Ботнеты

Что такое ботнет

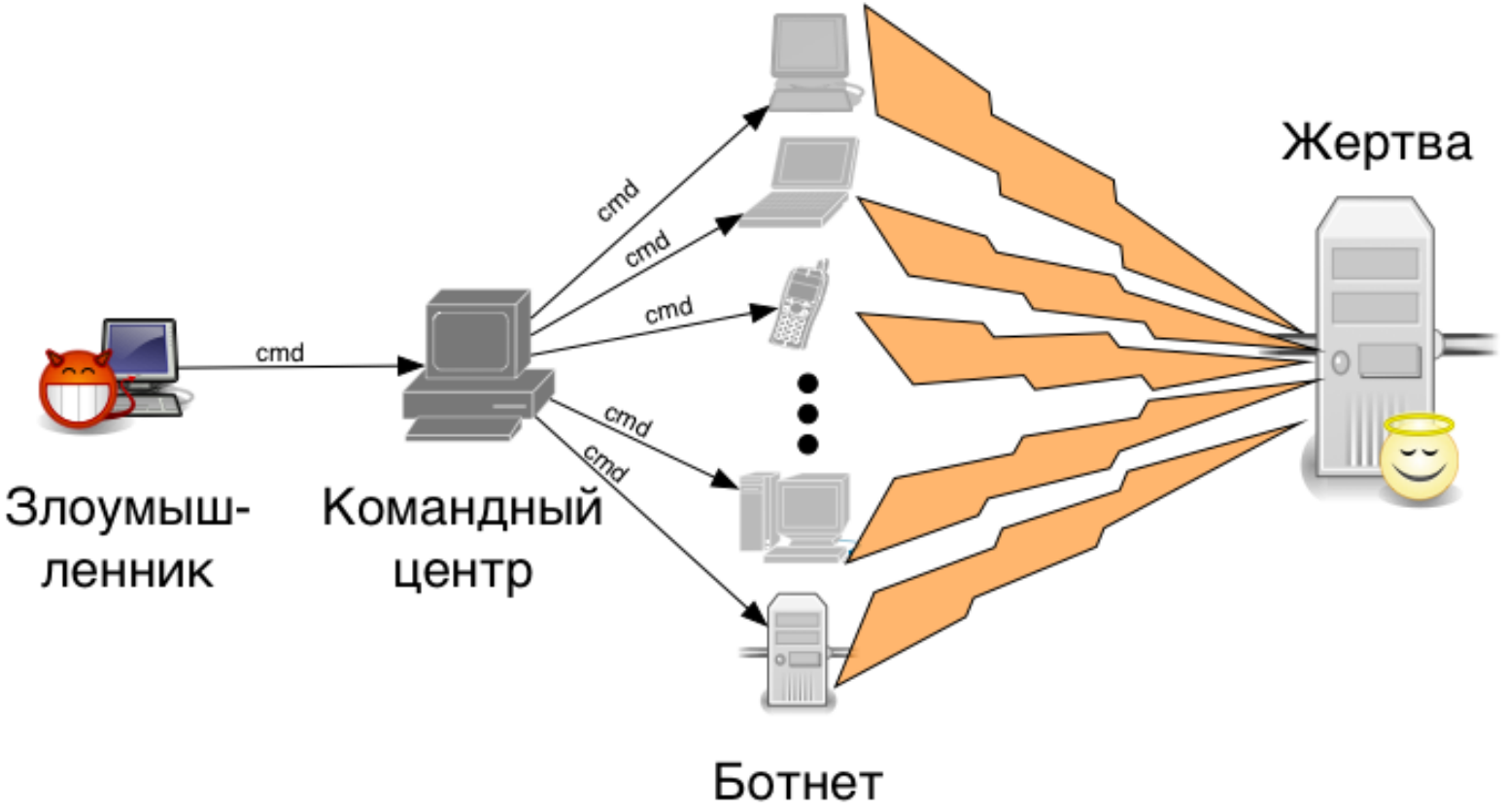
- Robot ➔ bot ➔ бот
- **Бот** – программный код на
  - серверах,
  - пользовательских компьютерах,
  - мобильных устройствах
  - и любых других устройствах, подключенных к сети,

  способный выполнять команды, поступающие из центра управления.
- Множество ботов под единым управлением называется **ботнетом**
- Ботнеты используются для атак разных видов, не только для атак на канальную емкость.

Как создаются ботнеты?

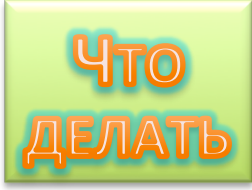
- Заражение вирусами (типичный сценарий для пользовательских компьютеров)
- Взлом систем через стандартные уязвимости (типичный сценарий для серверов, пример – уязвимость в WordPress)
- Легитимная установка ПО, содержащего «закладки» (например, из Google Play).

Схема ботнета



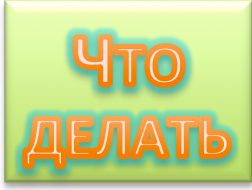
Как противодействовать?

- Существенный прогресс требует коллективных усилий участников индустрии, в первую очередь:
  - Операторов
  - Датацентров
  - Хостеров
  - Владельцев ресурсов



Ботнеты vs оператор связи/датацентр/хостер

- Участвуйте в профилактике заражений (дистрибьюция антивирусов, например)
  - Эффективность невелика, но… лучше, чем ничего
- Отслеживайте участников ботнетов в своих сетях, используйте эту информацию в работе
  - Анализ аномалий трафика на стороне оператора или через независимый мониторинг
    - FastNetMon Павла Одинцова (см.ссылки в конце)
  - Предпринимайте максимум мер противодействия, разрешенных вашими договорными обязательствами



Не становитесь частью ботнета!

- Установите антивирус для сайтов (например, Manul)
  - Или хотя бы проводите контроль целостности файлов
- Организуйте регулярный анализ журнальных файлов
  - Кстати, защищайте журнальные файлы!
- Рассмотрите возможность использования независимого мониторинга
  - Как минимум - проверяйте попадание в черные списки поисковиков!

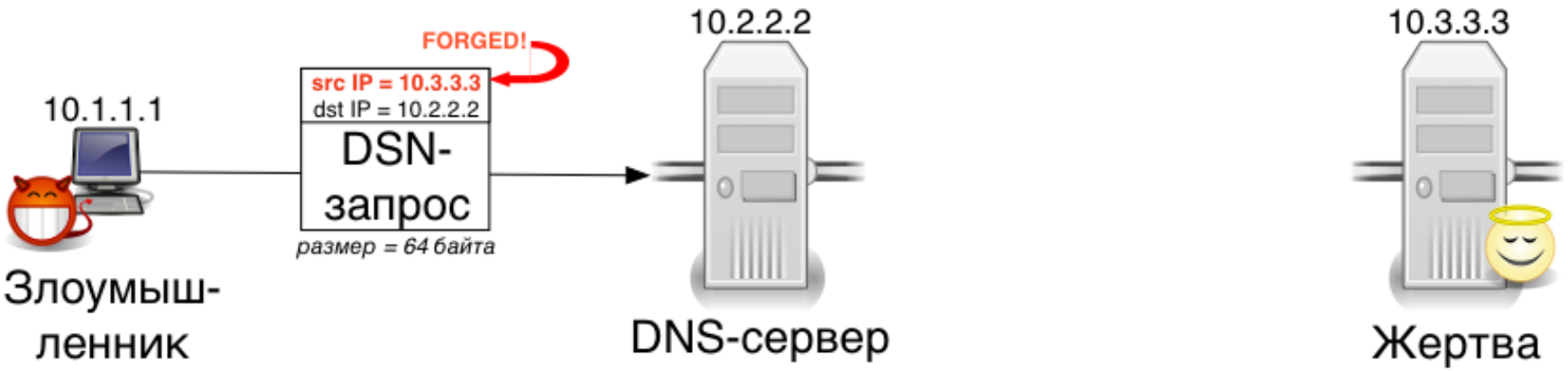
Атаки на канальную ёмкость

Амплификаторы

Что такое амплификаторы (1)

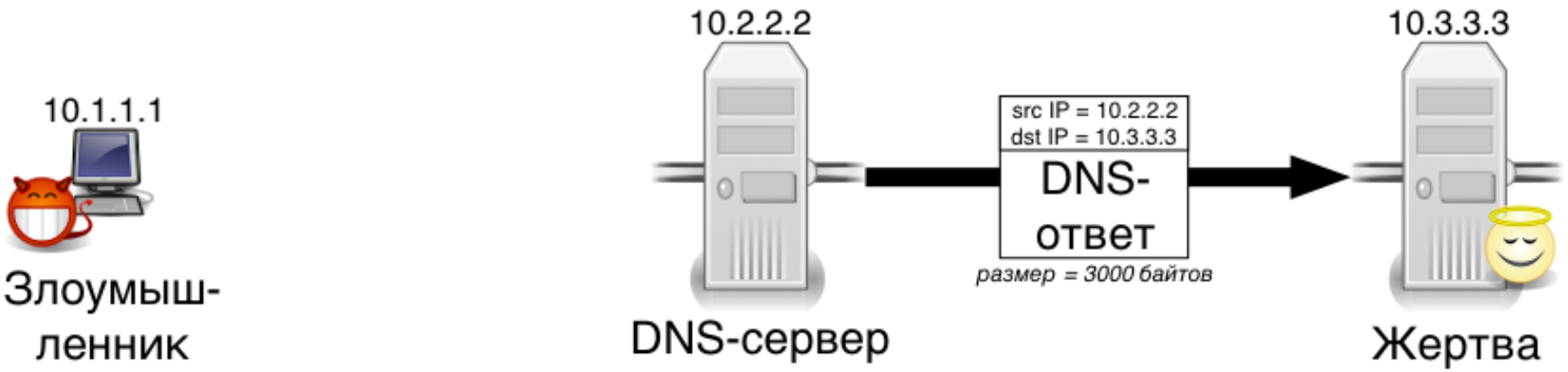
Рассмотрим сценарий:

- Злоумышленник с адреса 10.1.1.1 отправляет DNS-запрос на DNS-сервер с адресом 10.2.2.2, поменяв в IP-пакетах IP-адрес отправителя со своего на 10.3.3.3



Что такое амплификаторы (2)

- DNS-сервер отвечает на запрос, отсылая его отправителю. Так как адрес отправителя подделан, то ответ уходит в адрес компьютера-жертвы с адресом 10.3.3.3



Что такое амплификаторы (3)

- Предположим, что DNS-сервер по адресу 10.2.2.2 поддерживает расширение EDNS0 (увеличивающее разрешенный размер ответного пакета до 4096 байтов), и существует запрос (например, ANY), порождающий ответ размером 2000-4000 байтов в ответ на короткий запрос (64 байта).
- Тогда на каждый байт, отосланный злоумышленником 10.1.1.1, в адрес жертвы 10.3.3.3 будет отправляться 30-60 байт мусора. При этом атакующий «экранирован» от жертвы промежуточным DNS-сервером.
- Таким образом, **всего один такой сервер способен превратить 100 мегабит/с полосы со стороны атакующего в 3-6 гигабит/с в сторону жертвы**

Такая ситуация называется **амплификацией трафика**, а DNS-сервер в описанном сценарии называется **амплификатором**.

Что такое амплификаторы (4)

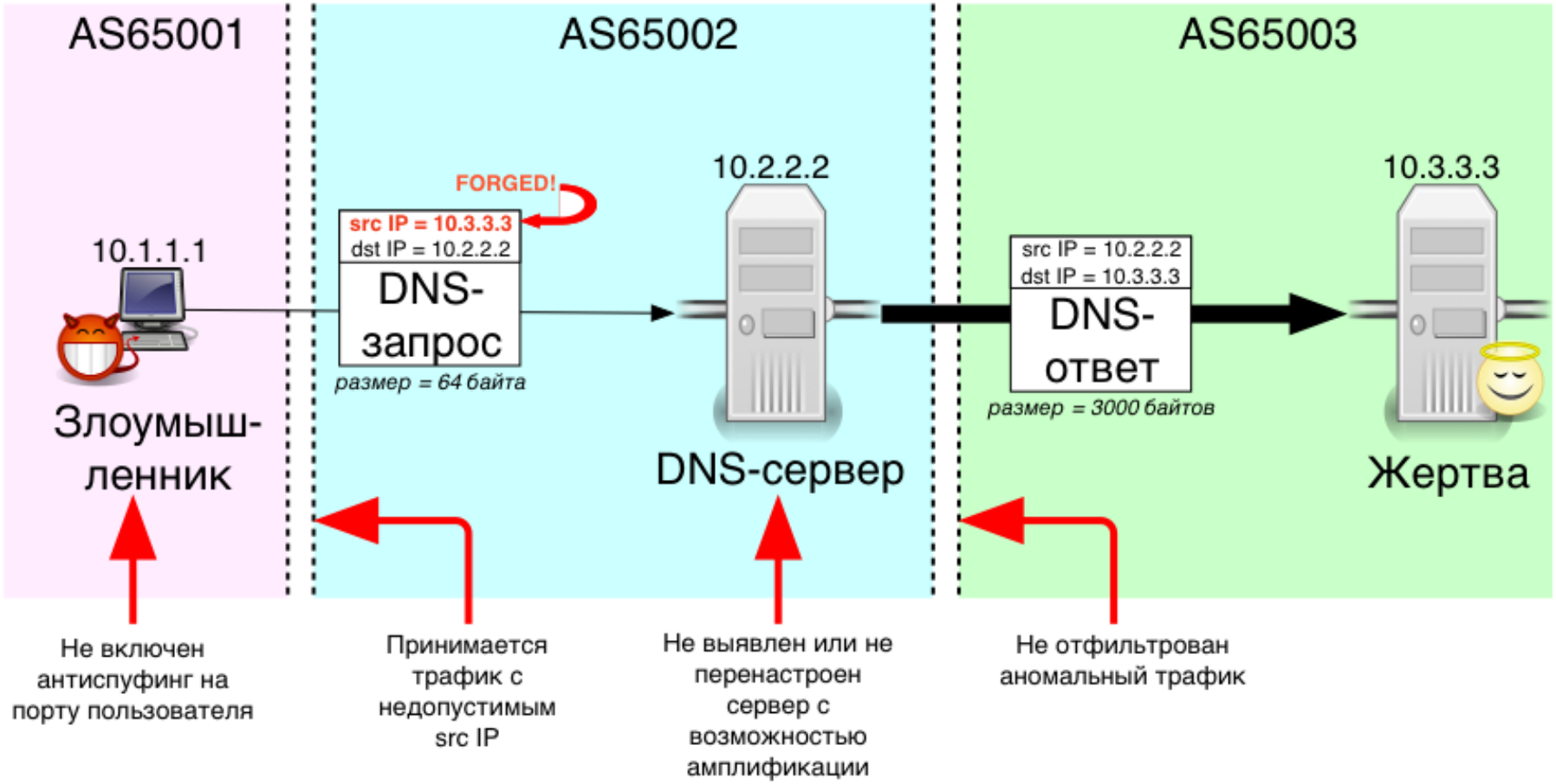
- Атака с амплификацией трафика становится возможна при выполнении нескольких условий:
  - Ни на прикладном уровне, ни на уровне приложения не устанавливается соединение,
  - Не производится авторизация запроса,
  - Размер ответа на запрос может существенно и воспроизводимо превышать размер запроса.
- Популярные протоколы, используемые для амплификации: DNS, NTP, SSDP, Chargen/UDP, ICMP.
- Общая полоса атаки свыше 100 Гбит/с с использованием набора амплификаторов – уже обыденное и рутинное явление.

Коэффициенты амплификации популярных протоколов

Протокол	Коэффициент
DNS	28-54
NTP	500-1300
SNMPv2	6
NetBIOS	4
SSDP	30
Chargen	350
QOTD	140
BitTorrent	4
Kad	16
Quake Network Protocol	64
RIPv1	130
Portmap (RPCbind)	7-28

Заглянем немного глубже…

4 причины, почему атака стала возможной

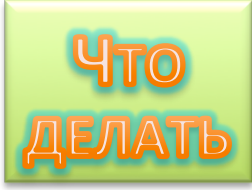


А почему это должно волновать операторов?

Некоторые последствия атаки (список не исчерпывающий):

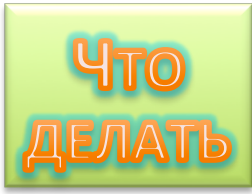
- Атакуемый ресурс выведен из строя
- Вместе с ним выведены из строя и другие ресурсы в AS65003.
  - Недополученная прибыль оператора **от всех** клиентов
  - Ущерб репутации оператора
- Оплата мусорного трафика
  - Наиболее вероятный вариант, что AS65002 и AS65003 – небольшие операторы и не имеют прямого стыка. Наиболее типичный вариант расчетов за трафик в этом случае – по превалирующему виду (входящий/ исходящий). Поэтому за паразитный трафик заплатят **оба** оператора.

Вывод: атаки DDoS опасны всей индустрии, а не только конкретным жертвам.



Как не стать частью атаки с использованием амплификации?

- Контролируйте трафик пользователей на их порту, блокируйте спуфинг.
- Блокируйте трафик с некорректным источником на границе сети
- Выявляйте сервера в своей сети и в клиентских сетях, которые могут использоваться для амплификации
  - Своими силами или с помощью внешних мониторинговых сервисов
  - Исправляйте настройки уязвимых серверов под вашим управлением
  - Обсуждайте с клиентами настройки их серверов, или ограничивайте потенциально опасный трафик от них
- Выявляйте аномальный трафик на границе сети (трафик амплификаторов достаточно легко распознать)
  - FastNetMon



Общие принципы настройки сервисов

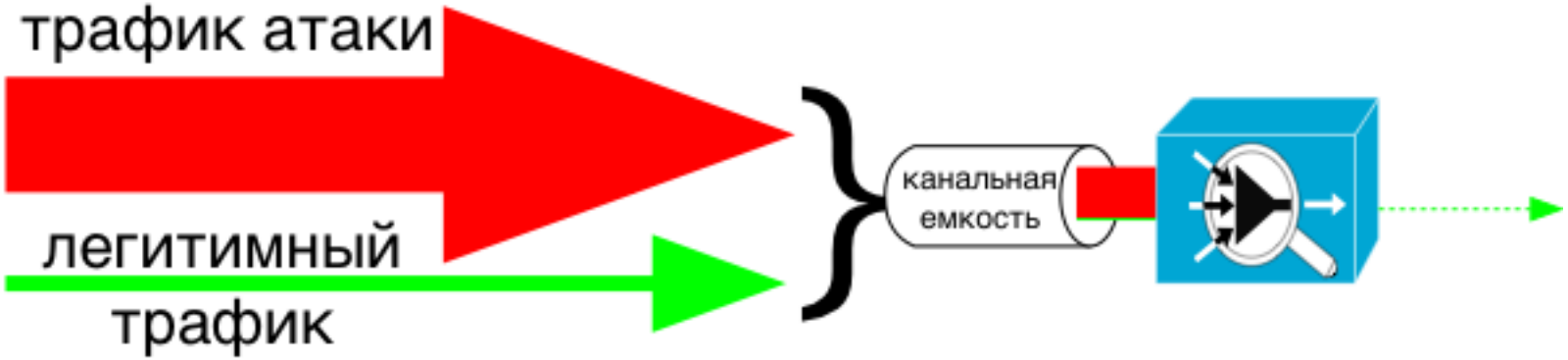
- Rate limit
  - Общий
  - Per src-IP
  - **Per src-prefix**
  - Не забывайте, однако, про NAT
- Ограничение круга пользователей ваших сервисов (полное или по типам запросов), ограничение опасных сервисов
  - Амплификация NTP выполняется с помощью мониторингового запроса monlist. Если она вам не нужна (скорее всего) – запретите её. Если нужна, то разрешите её для использования только вами.
- Fall-back на протоколы на основе сессий
  - Google DNS: переход с UDP на TCP, если ответ больше чем в 2 раза превышает запрос
  - NTP: команда mrulist вместо monlist

Атаки на канальную ёмкость

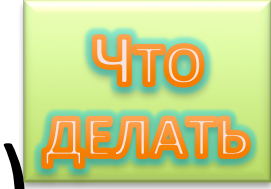
Общие рекомендации

Общие сложности

- Простое включение фильтрации на стороне клиента не помогает
  - канал забит, и даже если отфильтровать весь мусор - полезного трафика все равно почти нет



- Наращивание связности – долго и дорого.



Потенциальным жертвам (1)

Если есть своя автономная система:

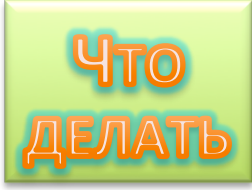
- Развивайте максимально связность – самостоятельно или с помощью облачных сервисов
  - BGP Anycast
- Фильтруйте аномалии трафика на всех стыках
  - Покупка оборудования или в рамках облачного сервиса, предоставившего связность
- Используйте BGP Flowspec для фильтрации аномалий, если аплинк дает такую возможность
  - Случается это пока что, увы, крайне редко.
- Защищайте свои префиксы целиком
  - Ваши префиксы не являются секретом (RIPE DB, RADB).
  - Чтобы забить канал, нужно послать мусор не на точный IP-адрес ресурса, а на любой IP-адрес, маршрутизируемый в канал – или на диапазон адресов.

Потенциальным жертвам (2)

Если своей автономной системы нет:

- Не используйте провайдеро-независимые адреса (PI)
- Выбирайте хостинг/датацентр с максимально развитой связностью
  - Может помочь против «пионерских» атак
- Избегайте shared hosting
- Используйте ступенчатую фильтрацию трафика
  - Много мусора можно отфильтровать просто на уровне firewall
  - Узнайте, не дает ли ваш хостер и/или датацентр возможности фильтровать часть трафика ещё у него?
- Задумайтесь о подключении к облачным сервисам фильтрации
  - Возможно, не через публичную сеть, а по MPLS VPN.

Операторам, дацентрам, хостерам



- Анализ небольших потоков трафика – это дешево
  - Выявляйте аномалии максимально близко к клиентским стыкам
- Превентивно анализируйте свою сеть на наличие ботнетов и амплификаторов
  - Своими силами или с помощью внешних мониторинговых сервисов

Важные инициативы индустрии

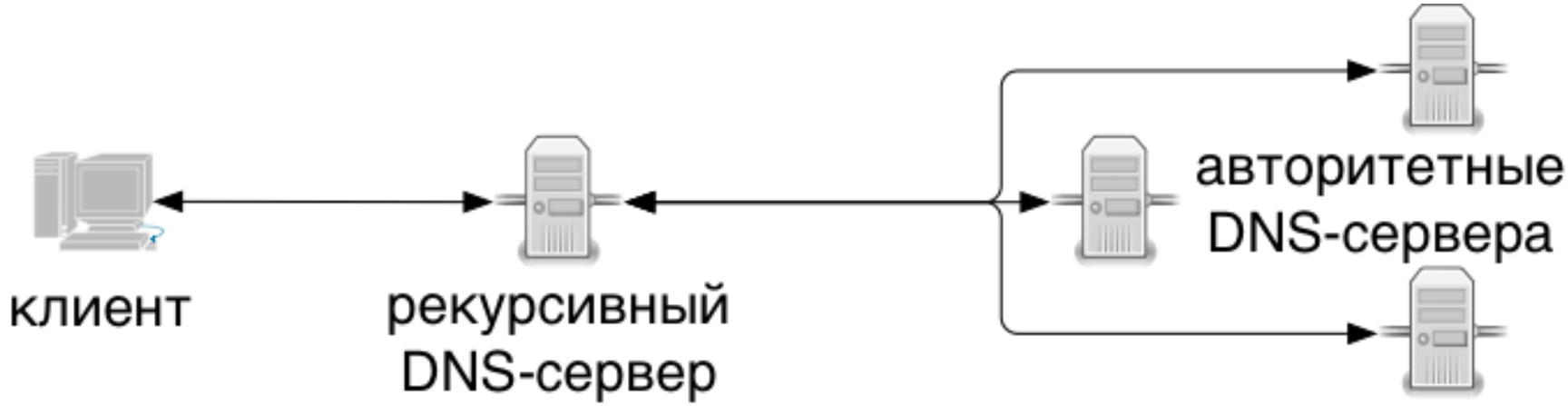
- MANRS = Mutually Agreed Norms for Routing Security
  - Цель: противодействие искажению маршрутной информации и спуфингу IP-адресов
  - Инициатива ISOC, описывающая предметный набор действий для достижения цели
  - Операторам можно (и нужно) присоединиться уже сейчас
- DOTS = DDoS Open Threat Signalling
  - Цель: создание инфраструктуры для сквозной сигнализации об аномалиях трафика начиная с CPE и кончая облачным сервисом.
  - Рабочая группа в IETF
  - Приглашаются сетевые вендоры и разработчики

(См. ссылки в конце презентации)

Атаки на инфраструктуру

DNS

- Невозможность разрешения имени ресурса в IP-адрес = недоступность ресурса
- Общеизвестная схема работы DNS:



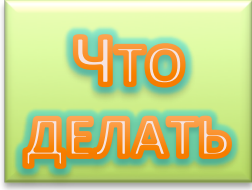
Атаки на инфраструктуру

DNS

Атаки на DNS

- Вывод из строя авторитетных (primary и secondary) DNS-серверов.
  - Могут использоваться как уязвимости ПО, так и DoS/DDoS-атака на сами сервера
  - После истечения время жизни записи у рекурсивных DNS-серверов ресурс становится недоступен
    - Хорошая новость: чтобы это случилась, необходима весьма долгая атака, что дает выигрыш по времени
    - Плохая новость: в кеше многих рекурсивных серверов нужной записи может не оказаться
- Исчерпание ресурсов на рекурсивных серверах
  - Запросы случайных доменов, возвращающие ошибку (NXDOMAIN): вымывание кеша и потеря производительности
  - Запросы случайных доменов, остающиеся без ответа вообще: исчерпание пула запросов и блокирование создания новых
  - Запросы доменов, сервера которых переходят на DNS/TCP, и открывают соединение и не передают данные: исчерпание сокетов и пула запросов

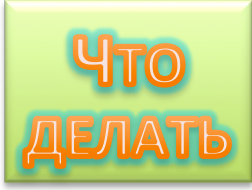
Итог. Атаки на DNS могут отражаться на полной или частичной доступности ресурса при том, что управление DNS-серверами зачастую осуществляется сторонней компанией.



Рекурсивные DNS-сервера

- Ограничивайте доступ из мира к своим рекурсивным DNS
  - Это не вполне тривиальная задача!
- Анализируйте их журнальные файлы, мониторьте состояние для выявления атак
  - Это ваши деньги: если у пользователя не откроется www.google.com, то виноваты будете вы.
- Настраивайте параметры серверов (размер пула запросов, размеры кешей и пр.) с запасом
- Если не умеете делать всего вышеперечисленного – переводите пользователей на внешние рекурсивные DNS-сервера, устойчивые к атакам
  - Google, Yandex, OpenDNS (Cisco)

Авторитетные DNS-сервера и данные зон



- Используйте сеть DNS-серверов
  - Размер зависит от вашей оценки риска атаки
- Защищайте DNS-сервера от DoS/DDoS-атак та же, как и основные сервера вашего ресурса
- Прописывайте время жизни записей с умом
  - Большой TTL – медленные обновления
  - Маленький TTL – быстрое устаревание кешей

Тоже проще не создавать всё самим, а воспользоваться готовыми облачными защищенными DNS-сервисами.

Атаки на инфраструктуру

Воровство маршрутов

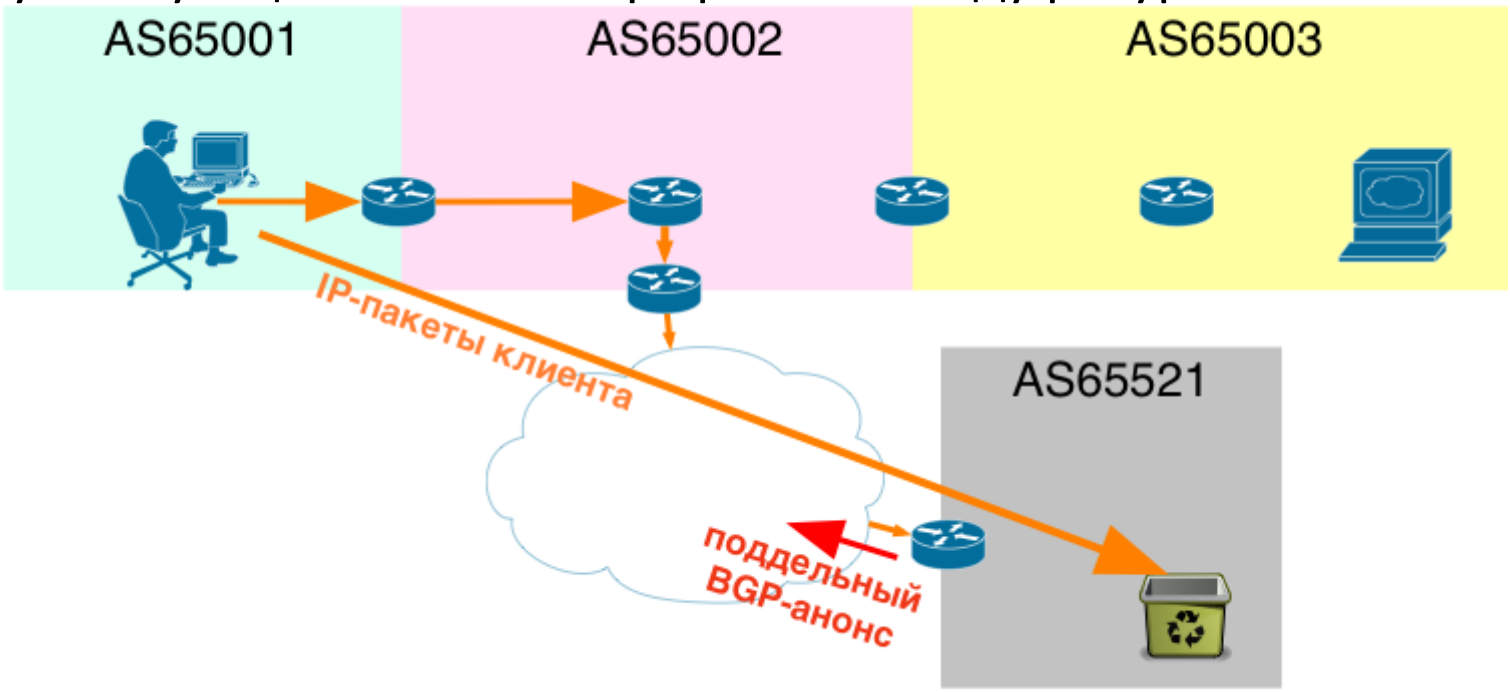
Маршрутная информация

- Для построения маршрутов в интернете используется протокол BGP
- В оригинальном протоколе BGP отсутствует защита от подделки маршрутной информации
- Набирающая популярность технология RPKI позволяет предотвратить только часть подделок
  - Проверяется валидность сочетания префикса и Origin
  - Корректность AS PATH не контролируется!
  - Но всё же лучше, чем ничего

Атака на систему маршрутизации

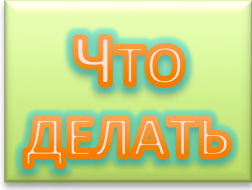
Уязвимости в прошивках маршрутизаторов могут давать злоумышленникам возможность создавать некорректные BGP-анонсы, уводя трафик к ресурсу «в никуда».

- Маршрутизатор при этом может быть любой, не обязательно участвующий в обмене трафиком между ресурсом и клиентом:



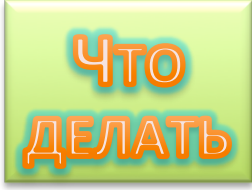
Проблемы атак на систему маршрутизации

- Может приводить к «непонятному» частичному нарушению доступности ресурса
- Не только владелец ресурса, но и его оператор не могут обнаружить такую атаку на основании *своей* маршрутной информации
  - Используя публичные looking glass и специализированные средства мониторинга атаку выявить можно – но специалисты жертвы должны знать, куда смотреть
- [Пока что?] нет средств для автоматической отработки таких атак
  - Требуется «ручная» высококвалифицированная работа.



Владельцам автономных систем

- Не экономьте на сетевых инженерах
  - (если вам важна стабильная работа сети)
- Мониторьте свою связность
  - BGPlay, BGPMON, Renesys, Radar/Qrator
- Подумайте о анонсах /24
  - Самый маленький допустимый в интернете префикс
- Внедрите RPKI
  - не решает всех проблем, но лучше, чем ничего



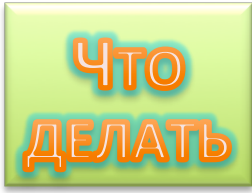
Всем остальным

- Если вы не готовы строить свою сеть, получать автономную систему и набирать команду сетевых инженеров – ищите грамотного оператора
  - Сами вы ничего в этом случае сделать не сможете

Атаки на операционную систему

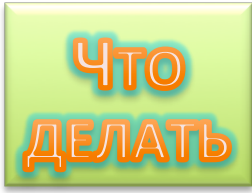
Атаки на сетевой стек

- Обратная сторона медали: сессионность TCP означает необходимость расходовать ресурсы на соединения, что создает возможность для атак
  - Наиболее известные пример - SYN-flood: запросы на открытие TCP-соединений
  - Создание «пустых» соединений, которые расходуют ресурсы, но по которым потом не передаются данные, кроме служебных данных протокола TCP.
- Ошибки в сетевом стеке
  - Нашумевшие когда-то примеры:
    - LAND-attack: src IP = dst IP, вызывавшая бесконечную обработку поступивших сегментов TCP
    - Christmas tree attack: включение всех TCP-флагов вызывало нарушение работы операционной системы
    - Teardrop attack: kernel crash при обработке перекрывающихся сегментов TCP
  - Новые уязвимости могут быть обнаружены в любой момент



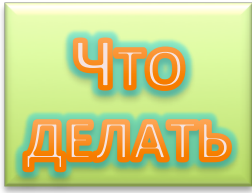
Операторам

- SYN-flood, SYN-ACK-flood и пр. требуют спуфинга адреса отправителя
  - Все те же меры по борьбе со спуфингом



Владельцам ресурсов (1)

- Настраивайте TCP/IP-стек вашей операционной системы, поднимайте лимиты до разумных (для конкретно вашего проекта) значений
  - Вместе с тем, ограничивайте число соединений с одного IP-адреса
- SYN-flood по-прежнему очень популярен
  - Включите SYN-Cookie
    - Механизм SYN-Cookie помещает информацию об отправителе в номера последовательности протокола TCP ответного пакета SYN-ACK. Ресурсы под соединение выделяются только после получения последнего пакета ACK в TCP handshake.
- Фильтруйте мусор на ранней стадии
  - Изучите возможности сетевого фильтра в используемой операционной системе
  - Избегайте фильтрации, требующей хранить какую-либо информацию (пожар бензином не тушат)
  - Используйте существующие возможности по настройке вашей операционной системы
  - Если у вас сильная команда разработки – можно применить механизмы типа netmap для анализа поступающего трафика (см.ссылки в конце)



Владельцам ресурсов (2)

- Мониторьте состояние открытых TCP-соединений, разрывая подозрительные (скорость передачи ниже минимальной, длительность соединения выше максимальной)
  - Требуется сопоставление данных ядра системы и данных приложения
- Следите за предупреждениями об уязвимостях и появлении исправлений для ваших операционных систем.
- Если риски велики и/или вы не можете сами реализовать описанное выше – используйте специализированное оборудование или облачные сервисы защиты.

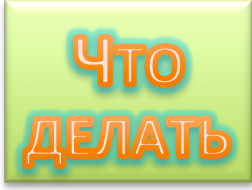
Атаки на приложение

Атаки на приложение (1)

- Они же «L7-атаки».
- Наиболее сложные в подавлении, так как основываются на особенностях конкретных приложений.
- Не всегда легко обнаруживаются.
  - Например, при эмуляции full stack browser на ботах
    - Но это недешевое удовольствие.

Атаки на приложение (2)

- Атаки могут осуществляться на ядро информационной системы
  - Например, повторяющиеся «тяжелые» запросы к базе замедляют или парализуют работу базы
  - Или массовые запросы случайных URL, проходящие через frontendы на backend (так как в кешах таких URL нет).
- Атаки могут осуществляться на frontendы
  - Множественные HTTP-запросы, эмулирующие огромное число посетителей
- Атаки могут осуществляться на клиентское ПО
  - Для отрисовки страницы браузер клиента строит его внутреннее представление – Document Object Model (DOM). Если какой-то элемент страницы не загружается, то DOM не строится, и страница не отображается или отображается с задержкой.
  - Атака: выявление внешних объектов, которые загружаются синхронно (участвуют в построении DOM) и атака на ресурсы, где эти объекты расположены



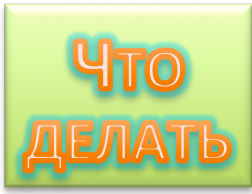
Владельцам ресурсов

- Определяйте разумное число одновременных запросов с одного IP и выставляйте лимиты
  - Помните про NAT
  - Не очень помогает против ботнетов
- Профилируйте свой ресурс, выявляйте «тяжелые» запросы, оптимизируйте архитектуру
  - Разделите приложение на backend и легкие frontendы, если оно еще не разделено
    - Nginx – большой опыт использования, много best practice
  - Подготовьте механизм существенного увеличения числа frontendов в случае необходимости
  - В простых случаях модель атаки может быть определена по журнальным файлам и заблокирована вручную в конфигурации сервера и/или firewall
- Уменьшайте зависимость ресурса от внешних объектов
  - Все обращения к внешним объектам должны быть асинхронными, не влияющими на построение DOM
- В тяжелых случаях используйте CAPTCHA
  - Минусы – возможность обхода и ухудшение UX
- Подумайте об использовании облачного сервиса или оборудования для защиты от L7-атак.
- Даже при наличии защиты имейте двух-трехкратный запас по производительности
  - Доля false negative в случае атаки на приложение может быть высокой в силу сложности распознавания

Финансовые атаки

На что расходуются деньги?

- Трафик
  - Существует способ отдачи контента, денежная стоимость которого для владельца достаточно высока
    - Атака: выкачка ботнетом «тяжелого» контента этим способом
- Мощности облака
  - Ресурс автоматически контролирует число виртуальных машин в зависимости от нагрузки
    - Атака: ботнет создает иллюзию большого количества посетителей, ходящий по сайту, вызывая создание большого числа виртуальных машин
  - Ресурс загружает пользовательские данные в облачное хранилище, не контролируя их размер
    - Ботнет начинает загружать большие «пользовательские объекты», расходуя место и деньги владельца ресурса



Потенциальным жертвам

- Заранее оценивайте, расход каких ресурсов может быть связан с существенными расходами
- Оценивайте возможность бесконтрольного расходования этих ресурсов и меры по противодействию такому расходованию
- Ведите автоматический, независимый и постоянный контроль учета платных расходуемых ресурсов.

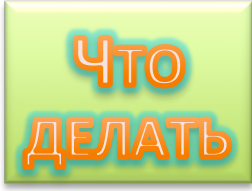
Почему важно не быть видимым

Зачем прятать ресурсы

- Наиболее типичный сценарий – использование облачной защиты через публичную сеть
  - Облачная защита пропускает трафик через себя и фильтрует трафик атаки. При этом для внешнего мира адресом ресурса является адрес самой облачной защиты.
  - Но если злоумышленник выяснил настоящий адрес ресурса, то он может атаковать его напрямую.
    - Даже если на пути к ресурсу будет firewall, атака на канальную емкость окажется успешной.
- Также, очевидно что базы данных и backendы не должны быть видны «напрямую»
  - Особенно важно в случае, если, например, база данных и backend расположены в разных датацентрах и взаимодействую через публичную сеть
  - Если злоумышленники выяснят, в каком именно датацентре стоит база данных (или уж подавно если узнают ее IP-адрес), то становится возможна атака на неё (напрямую или на датацентр) и вывод ресурса из строя.

Источники нежелательной информации

- Исходящие соединения
  - Самый популярный вариант – почтовая рассылка с немодифицированными заголовками письма
  - Обращения за внешними объектами («введите URL изображения с вашим аватаром»)
  - Сообщения о внутренних ошибках, выдаваемые пользователю («the database on 10.5.8.4 is not responding»)
- Журнальные файлы, по ошибке доступные по Web
- Имена и адреса внутренних сервисов в доступной всем DNS-зоне
- Старые DNS-зоны
  - До перевода ресурса под облачную защиту, например



Владельцам ресурса

- Не допускать исходящих соединений от «спрятанных» сервисов наружу напрямую
  - В частности, в случае почты – отсылать через маскирующий сервер или вовсе перейти на внешний почтовый сервис
- Подумать о переходе с публичных сетей для внутреннего взаимодействия (облачная защита↔ресурс, backend↔database etc) на MPLS VPN или выделенные каналы
- Контролировать сообщения об ошибках, выдаваемых посетителю
- Ограничивать доступ к служебной информации
  - Журнальные файлы, конфигурационные файлы etc

Collateral damage

Опасность общих объектов

Любой общий (разделяемых) объект может являться причиной нарушения доступности к интернет-ресурсу: атака на один ресурс часто выводит из строя сотни и тысячи соседних.

Чтобы не разделить их судьбу, надо внимательно анализировать используемые разделяемые объекты.

Неполный список примеров

- Shared hosting: процессор, память. Атака на один сервер в shared hosting выводит из строя всё.
- Внешние и внутренние каналы связи датацентра, хостеру, внутри структуры хостера
  - Подключение «1 Gbps на стойку». Тривиальная атака на канал с полосой 2 Gbps делает недоступными сразу все сервера в стойке.
  - Подключение к облачному сервису защиты по MPLS VPN... который предоставлен тем же оператором, что и подключение к публичной сети, и идет по тому же волокну через интерфейс 10 Gbps. Атака в 12 Gbps делает защищенные сервис недоступным.
- Сетевое оборудование датацентра
  - DoS на основе свежей уязвимости делает недоступным весь датацентр

Что еще почитать и посмотреть

http://www.ddosattacks.net/

https://www.nanog.org/sites/default/files/10-TzvetanovDDOS2.pdf

https://www.nanog.org/sites/default/files/meetings/NANOG64/1045/20150603_Tzvetanov_Tutorial_Denial_Of_v2.pdf

https://www.routingmanifesto.org/manrs/

https://www.ietf.org/proceedings/93/slides/slides-93-dots-3.pdf

https://www.us-cert.gov/ncas/alerts/TA14-017A

http://www.slideshare.net/MeYouSlide/beloenko-ecommerce

https://threatpost.ru/metkij_vystrel_po_oblachnoj_zashshite_ddos/12548/

https://blog.cloudflare.com/single-rx-queue-kernel-bypass-with-netmap/

http://www.enog.org/presentations/enog-9/17-FastNetMon_ENOG_pdf.pdf

https://www.nanog.org/sites/default/files/wed.general.trafficdiversion.serodio.10.pdf

https://www.youtube.com/watch?v=ifmRgQX82O4

Спасибо!