

Защита от DDoS - геофильтрация трафика на маршрутизаторах Juniper с применением Source Class Usage (SCU)

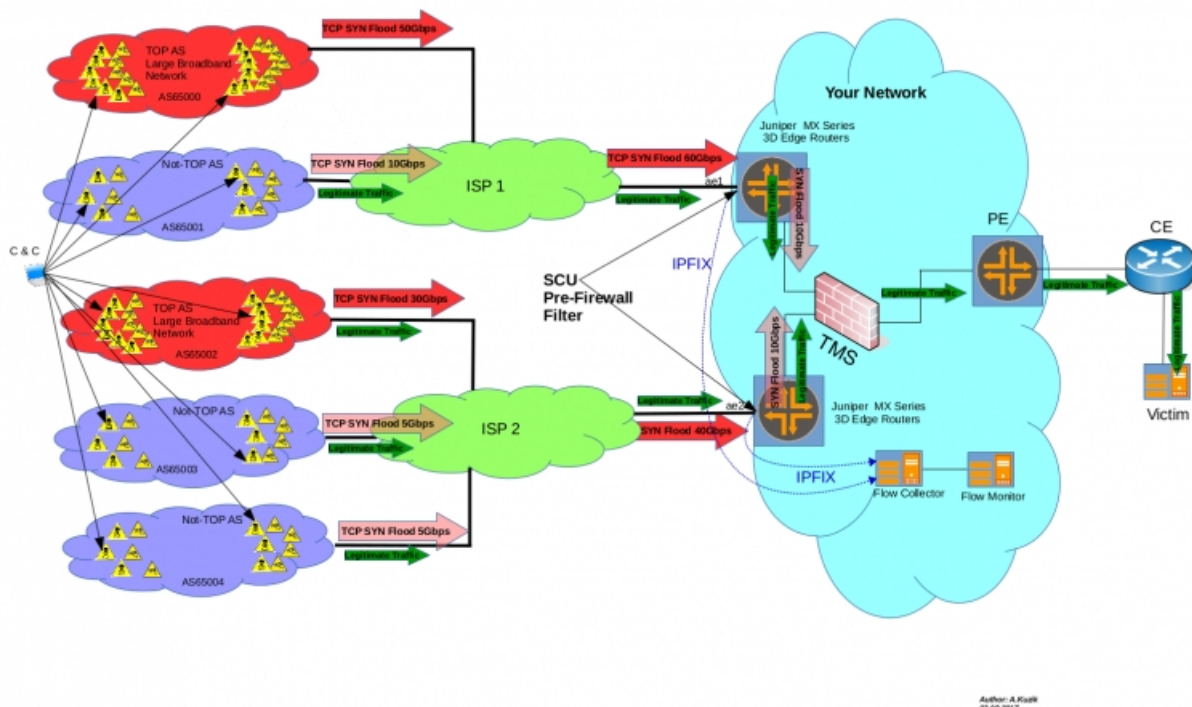
При написании первой статьи мною не преследовалась цель сделать какое-либо открытие, а просто хотелось поделиться практическим опытом внедрения. На мой взгляд, описанная конфигурация является более простой, по сравнению с отраженной в презентации от вендора с применением Destination Class Usage (DCU). Но именно последнее натолкнуло на мысль оценить возможность применения DCU и SCU в системах фильтрации трафика, т. к. имею хорошее представление и практический опыт работы с DCU в Ростелекоме, где он применялся для зонной тарификации трафика VPN-сервисов.

В последнее время многие пишут и обсуждают способы защиты от DDoS-атак, пытаются создать и внедрить новые стандарты, томо ожидая их превращения из Draft в полноценный Internet Standard, а потом еще больше ожидая вендорских имплементаций. Мне кажется, что в оперативном плане более оправданным является применение существующих решений производителей, хотя и развитие рабочих предложений тоже необходимо.

В любой операторской сети с реализованными сервисами защиты от DDoS, как правило, существует несколько ступеней очистки трафика, что вполне логично. Принять несколько сотен Gbps или Mpps "грязного" трафика и очистить его исключительно только с помощью TMS (Threat Management System) - неоправданно дорого. Обычно, первой ступенью очистки, особенно важной при атаках с большой volumetric, является Pre-firewall filter, задача которого уменьшить полосу "мусорного" трафика до значения, которое может легко "переварить" операторская сеть и TMS. Существует два основных варианта реализации данного фильтра - на пограничных маршрутизаторах собственной сети и в сетях аплинков. Если рассматривать вариант с реализацией внутри собственной сети, то это, обычно, stateful firewall filters, работающие на input/output на интерфейсах, или внедренный внутри сети FlowSpec. В сетях аплинков, как правило - FlowSpec, работающий по внутренней сигнализации, и по сигнализации, принимаемой с клиентских сетей. Если в первом варианте существует возможность более тонкой фильтрации, ограничив определенный destination IP, протокол, порт, флаг, размер пакета полисером, то во втором варианте предусмотрен только жесткий discard, мало чем отличающийся от blackhole, если атака имеет характеристики работающего сервиса.

SCU и DCU, по мануалам Juniper, предназначены для accounting-трафика по сопоставлению с определенным policy с match по определенным критериям. Т.е., если кратко, можно считать трафик по описанным в спецификации класс конкретным префиксам, BGP-community, target-community, etc. Firewall filter позволяет делать match по source-class и destination-class. Данные опции позволяют создать firewall filter с возможностью тонкой фильтрации по source AS, или если выразить простым языком, осуществлять геофильтрацию трафика с помощью Pre-firewall filter.

Пример практической реализации фильтрации TCP Syn-flood представлен на схеме. Конфигурация и ее описание даны ниже.



Данные о flow "чистого" и "грязного" трафика с помощью протокола IPFIX (в данной статье его работа рассматриваться не будет) снимаются с line cards маршрутизатора с определенной дискретностью (sampling) и передаются на Flow Collector, где обрабатываются и отображаются в системе мониторинга Flow Monitor. Это один из способов реализации мониторинга входящего flow, в разных операторских сетях он реализован по-разному. Одним из важных параметров, отображаемый в IPFIX является SrcAS. Кто сталкивался с огромными по объему атаками в сотни Gbps и наблюдал за автономными системами происхождения "грязного" трафика, обратил внимание, что при определенных видах атак распределение источников того же флуда далеко неоднородно. Т.е. существует какое-то число топовых ASN с преобладающим значением. Подобная картина отчетливо прослеживалась во время прошлогодних DDoS-атак с применением знаменитого ботнета Mirai.

Предлагаемый способ позволяет существенно ограничить уровень "грязного" трафика именно с ASNs, входящих в TOP.

Давайте поэтапно рассмотрим функционал на примере конфигурационного файла.

1. Создаем policy scu_ddos в котором делаем матч по протоколу bgp и по as-path source-as. Перечисленным атрибутам присваиваем source-class ddos.

```
set policy-options policy-statement scu_ddos term 1 from protocol bgp
set policy-options policy-statement scu_ddos term 1 from as-path source-as
set policy-options policy-statement scu_ddos term 1 then source-class ddos
```

2. В as-path source-as в виде регулярного выражения добавляем origin-as для топовых ASNs.

```
set policy-options as-path source-as ".* 65000|65002"
```

В нашем примере на рисунке они обозначены красным цветом, и с них, в приведенном примере, поступает 80Gbps TCP syn-flood.

3. Применяем созданный policy на экспорт к forwarding-table. Если на экспорт уже есть примененные policy, то scu_ddos желательно поставить первым, используя команду insert в соответствующем конфигурационном блоке.

```
set routing-options forwarding-table export scu_ddos
```

4. Создаем policer lim100m.

```
set firewall policer lim100m filter-specific
set firewall policer lim100m shared-bandwidth-policer
set firewall policer lim100m if-exceeding bandwidth-limit 100m
set firewall policer lim100m if-exceeding burst-size-limit 1m
set firewall policer lim100m then discard
```

5. Создаем firewall filter scu_ddos_limit, в котором осуществляем match по source-class ddos, ограничиваемому протоколу и флагу, считаем и ограничиваем полисером в 100 Mbps весь трафик соответствующий данным критериям.

```
set firewall filter scu_ddos_limit term1 from source-class ddos
set firewall filter scu_ddos_limit term1 from protocol tcp
set firewall filter scu_ddos_limit term1 from tcp-flags syn
set firewall filter scu_ddos_limit term1 then policer lim100m
set firewall filter scu_ddos_limit term1 then count scu_ddos_count
set firewall filter scu_ddos_limit term2 then accept
```

6. На интерфейсах включения аплинков включаем accounting и применяем на input созданный фильтр scu_ddos_limit.

```
set interfaces ae1 unit 0 family inet accounting source-class-usage input
set interfaces ae1 unit 0 family inet filter input scu_ddos_limit
```

7. Проверяем срабатывание счетчиков и полисера на интерфейсе.

```
run show firewall filter scu_ddos_limit-ae1.0-i | match "lim100m-ae1.0-i|scu_ddos_count-ae1.0-i"
scu_ddos_count-ae1.0-i 90763066484 1745338287
lim100m-ae1.0-i 343023655 9801281
```

Таким образом, мы можем нивелировать значительную часть syn-флуда (в примере на рисунке из суммарной полосы в 100 Gbps до 20 Gbps), тем самым предотвратить перегрузку собственных сетевых ресурсов и устройств очистки трафика.

При правильно подобранном значении полисера дропы пакетов проявляются только во время атаки. Необходимо понимать, что в момент атак полисером также будет дропаться и часть легитимного TCP syn-трафика с ограничиваемых автономных систем, что существенно не повлияет на уже установившиеся соединения.