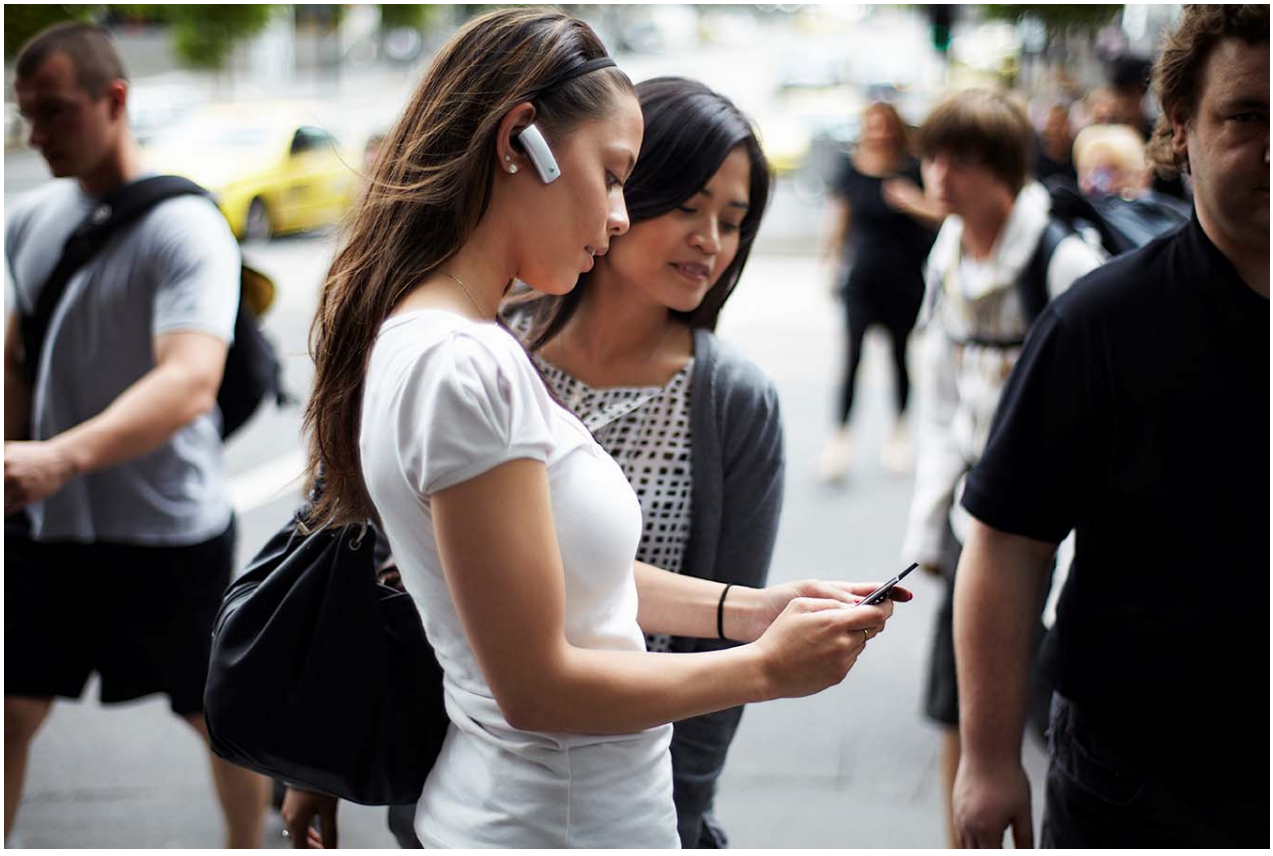
ericsson.com

Policy How To

Ericsson SmartEdge



Rev A

ERICSSON

Содержание

1 Полиси в SmartEdge

Под полиси в SmartEdge подразумевается действия, совершаемые над трафиком, проходящим через маршрутизатор. Полиси могут применяться как непосредственно к интерфейсам, так и к абонентским сессиям.

Все полиси в SmartEdge делятся на три основных типа:
- QoS Policy
- Forward Policy
- NAT Policy

QoS Policies отвечают за управление параметрами QoS, в частности, контроль и ограничение скорости, маркировка, Queueing и т.п. Forward Policies предоставляют возможности для организации функций HTTP Redirect, Policy Based Routing, Traffic Mirroring. NAT Policies говорят сами за себя – т.е. это трансляция сетевых адресов.

В общем случае любая Policy может быть применена к circuit (см. описание circuit в CLIPS How to), а это означает, что существует возможность управлять назначением той или иной полиси на абонентскую сессию средствами RADIUS, включая CoA. На практике это позволяет динамично по указанию с RADIUS применять выборочно NAT Policy, включать/отключать HTTP Redirect, либо Traffic Mirroring и т.д. и делать все это посессионно.

2 QoS Policy для организации функции Rate-Limit

2.1 Простые QoS Policy

Проще всего познакомится с полиси на базе примера. За выполнение функции rate-limit в SmartEdge отвечают два типа QoS Policy:
- QoS Policy Policing – это функция rate-limit трафика в направлении IN, с точки зрения SmartEdge;
- QoS Policy Metering – это функция rate-limit трафика в направлении OUT, с точки зрения SmartEdge.

Например, необходимо задать rate-limit на скорость 2 Мбит/с в обоих направлениях. Для данного случая QoS Policy будут выглядеть следующим образом.

```
--------------------------SmartEdge-------------------------
!
```

```
qos policy 2M-in policing
 rate 2000 burst 250000
!
qos policy 2M-out metering
 rate 2000 burst 250000
!
------------------------------------------------------------
```

Значение `rate` задается в Кбит/с, а значение `burst` в байтах. Сразу возникает вопрос, как вычислить `burst`? В общем случае `rate` это отношение `burst` к интервалу времени Tc (time committed). Значение Tc обычно берется равным 1 секунде. Тогда Burst = Rate*1/8. Получаем значение в байтах. Не забывайте в процессе вычислений, что в SmartEdge значение `rate` определяется в Кбит/с.

Такой способ вычисления `burst` прекрасно работает для обычного (штатного) трафика абонента, проходящего через полисер, т.е. в этом трафике есть несколько TCP-Flow, есть UDP. Иногда требуется специальная настройка полисеров под пользователей пытающихся проверить свою скорость при помощи Speedtest и прочих ресурсов. Rate-limit и token bucket механизмы весьма агрессивны в отношении TCP, поэтому, если попытаться «разогнаться» в рамках такого rate-limit при помощи одного единственного TCP-соединения, то «полка» может наступить немного ниже настроенного значения rate. При этом если «разгоняться» несколькими TCP-соединениями, то суммарная скорость устанавливается в соответствии с настроенной. Для того чтобы сделать rate-limit более дружелюбным к случаю с единичным TCP-соединением, можно использовать excess-burst, при вычислении которого значения Te (time exceeded) рекомендуется брать в интервале 1,5 - 2 секунды. Также существует возможность проинструктировать полисер, о том, что вычислять скорость необходимо только с учетом информации L3, т.е. не принимать во внимание overhead вносимый инкапсуляциями на L2.

Ниже представлен пример с использованием `excess-burst` и с функцией вычисления скорости на уровне L3.

```
--------------------------SmartEdge-------------------------
!
qos policy 2M-in policing
 rate 2000 burst 250000 excess-burst 375000
 rate-calculation exclude layer-2-overhead
!
qos policy 2M-out metering
 rate 2000 burst 250000 excess-burst 375000
 rate-calculation exclude layer-2-overhead
!
------------------------------------------------------------
```

Для того чтобы применить данные rate-limit на абонентскую сессию при помощи RADIUS, существует несколько способов, можно передавать названия QoS Policy в специальных VSA. Например, так:

```
-------------------------RADIUS-----------------------------
Qos-Policy-Policing = “2M-in”
Qos-Policy-Metering = “2M-out”
------------------------------------------------------------
```

Где `Qos-Policy-Policing` и `Qos-Policy-Metering` это Redback (vendor id 2352) VSA за номерами 87 и 88 соответственно, тип полей – string.

Либо можно создать абонентский профиль в конфигурации SmartEdge, который будет содержать в себе ссылки на определенные ранее QoS Policies. Тогда с RADIUS можно передать только название абонентского профиля.

```
-------------------------SmartEdge------------------------
!
subscriber profile base-profile-2M
 qos policy policing 2M-in
 qos policy metering 2M-out
!
------------------------------------------------------------

-------------------------RADIUS-----------------------------
Sub-Profile-Name = “base-profile-2M”
------------------------------------------------------------
```

Где `Sub-Profile-Name` это Redback (vendor id 2352) VSA за номером 91 тип поля – string.

Иногда встречается потребность хранить как можно меньше информации в конфигурационном файле устройства, при этом передавать как можно больше настроек со стороны RADIUS сервера, включая скоростные параметры сессий.

Для таких случаев можно использовать следующий подход конфигурирования.

```
-------------------------SmartEdge------------------------
!
subscriber default
 qos policy policing rate-default-in
 qos policy metering rate-default-out
!
!
```

```
qos policy rate-default-in policing
 rate 64 burst 8000
 rate-calculation exclude layer-2-overhead
!
qos policy rate-default-out metering
 rate 64 burst 8000
 rate-calculation exclude layer-2-overhead
!
-------------------------------------------------------------
```

Используя конструкцию subscriber default можно применять на сессию так называемые атрибуты по умолчанию, т.е. перечисленные в subscriber default атрибуты, будут всегда применятся к любой сессии принимающей состояние UP. При этом сессия будет иметь ограничитель по скорости в 64 кбит/с согласно примеру выше. Передавая в access-accept, либо в CoA значения скорости с RADIUS в атрибуте `Dynamic-QoS-Param` будут переписываться значения по умолчанию на необходимые. Ниже представлен пример для данного способа назначения скорости:

```
-------------------------RADIUS-----------------------------
Dynamic-QoS-Param += “police-circuit-rate rate-absolute 2000”
Dynamic-QoS-Param += “police-circuit-burst 250000”
Dynamic-QoS-Param += “meter-circuit-rate rate-absolute 2000”
Dynamic-QoS-Param += “meter-circuit-burst 250000”
-------------------------------------------------------------
```

`Dynamic-QoS-Param` это Redback (vendor id 2352) VSA за номером 196 тип поля – string. Этот атрибут является очень гибким с точки зрения наполнения параметрами, более подробно описан в документации.

2.2 Policy и Классы

В разбираемых примерах выше использовались простые QoS Policies с функциями rate-limit, при применении которых весь трафик, проходящий через абонентскую сессию подлежит заданным скоростным ограничениям. На практике часто требуется избирательно ограничивать скорость, например, трафик пиринговых сетей, а также доступ к локальным ресурсам оператора должен иметь скорость отличную от скорости доступа к ресурсам сети Интернет. Данную особенность в SmartEdge можно реализовать при помощи так называемых классов используемых в полиси.

Что такое класс? Класс – это набор строчек в ACL логически объединенных в одну группу. Классы задаются при помощи так называемых Policy ACL.

Например, предположим, что трафик от и к префиксам 10.193.0.0/16 и 155.53.0.0/16 является локальным и должен быть всегда доступен для абонентов на скорости не выше 50 Мбит/с. Все остальное – согласно тарифу. Для определения классов нам необходимо создать два Policy ACL, которые будут определять локальный и нелокальный трафик в направлениях IN и OUT.

Вот так будут выглядеть policy acl для нашего примера:

```
-------------------------SmartEdge-------------------------
!
policy access-list acl-directions-in
 seq 10 permit ip any 10.193.0.0 0.0.255.255 class cls-LOCAL
 seq 20 permit ip any 155.53.0.0 0.0.255.255 class cls-LOCAL
 seq 30 permit ip any any class cls-INET
!
policy access-list acl-directions-out
 seq 10 permit ip 10.193.0.0 0.0.255.255 any class cls-LOCAL
 seq 20 permit ip 155.53.0.0 0.0.255.255 any class cls-LOCAL
 seq 30 permit ip any any class cls-INET
!
-----------------------------------------------------------
```

Теперь для созданных классов мы зададим желаемые значения rate-limit:

```
-------------------------SmartEdge-------------------------
!
qos policy 2M-50M-in policing
 access-group acl-directions-in local
  class cls-LOCAL
   rate 50000 burst 6250000
  class cls-INET
   rate 2000 burst 250000
!
qos policy 2M-50M-out metering
 access-group acl-directions-out local
  class cls-LOCAL
   rate 50000 burst 6250000
  class cls-INET
   rate 2000 burst 250000
!
-----------------------------------------------------------
```

Где cls-LOCAL это класс трафика, т.е. доступ к локальным ресурсам, со скоростью не выше 50 Мбит/с, а cls-INET класс трафика, описывающий доступ к ресурсам сети Интернет со скоростью не выше 2 Мбит/с.

Применять данные policy с RADIUS можно также как и в примерах выше, т.е. либо через `Qos-Policy-Policing` и `Qos-Policy-Metering`, либо через абонентский профиль.

Также существует способ с использованием атрибута `Dynamic-QoS-Param`. При этом появляется возможность задавать параметры скорости индивидуально для каждого класса в access-accept, либо в CoA. Описанный ниже способ часто применяется в случаях, когда оператор предпочитает хранить как можно данных в RADIUS, а не в SmartEdge. Обычно у любого провайдера есть как минимум две категории пользователей, это пользователи лимитных тарифов и пользователи безлимитных тарифов с точки зрения потребления объема трафика. В примере ниже под категорию пользователей безлимитных тарифов был создан специальный абонентский профиль, который будет наследовать базовые значения полиси.

Итак, ниже представлен пример решающий следующую задачу: Для безлимитных тарифов скорость доступа к локальным ресурсам (10.193.0.0/16, 89.233.4.0/22) не ограничена (100 Мбит/с), скорость доступа в пиринг (10.0.0.0/8, 155.53.16.0/24, 188.12.224.0/22) ограничена скоростью 50 Мбит/с, скорость доступа в Интернет согласно параметрам получаемым индивидуально на каждую сессию с RADIUS.

```
--------------------------SmartEdge-------------------------
!
context local
!
policy access-list acl-directions-in
 seq 10 permit ip any 10.193.0.0 0.0.255.255 class cls-LOCAL
 seq 20 permit ip any 89.233.4.0 0.0.3.255 class cls-LOCAL
 seq 30 permit ip any 10.0.0.0 0.255.255.255 class cls-PEERING
 seq 40 permit ip any 155.53.16.0 0.0.0.255 class cls-PEERING
 seq 50 permit ip any 188.12.224.0 0.0.3.255 class cls-PEERING
 seq 60 permit ip any any class cls-INET
!
policy access-list acl-directions-out
 seq 10 permit ip 10.193.0.0 0.0.255.255 any class cls-LOCAL
 seq 20 permit ip 89.233.4.0 0.0.3.255 any class cls-LOCAL
 seq 30 permit ip 10.0.0.0 0.255.255.255 any class cls-PEERING
 seq 40 permit ip 155.53.16.0 0.0.0.255 any class cls-PEERING
 seq 50 permit ip 188.12.224.0 0.0.3.255 any class cls-PEERING
 seq 60 permit ip any any class cls-INET
!
!
subscriber profile base-unlimited
 qos policy policing qos-default-in
 qos policy metering qos-default-out
!
! ** End Context **
!
```

```
!
qos policy qos-default-in policing
 access-group acl-directions-in local
  class cls-LOCAL
   rate 100000 burst 12500000
  class cls-PEERING
   rate 50000 burst 6250000
  class cls-INET
   rate 64 burst 8000
!
qos policy qos-default-out metering
 access-group acl-directions-out local
  class cls-LOCAL
   rate 100000 burst 12500000
  class cls-PEERING
   rate 50000 burst 6250000
  class cls-INET
   rate 64 burst 8000
!
------------------------------------------------------------
```

Тогда на момент аутентификации пользователя необходимо задать соответствующие ему параметры класса `cls-INET`. Если мы хотим ограничить доступ в Интернет скажем до 2 Мбит/с, то access-accept должен выглядеть приблизительно следующим образом:

```
--------------------------RADIUS-----------------------------
Sub-Profile-Name = “base-unlimited”
Dynamic-QoS-Param += “police-class-rate cls-INET rate-absolute 2000”
Dynamic-QoS-Param += “police-class-burst cls-INET 250000”
Dynamic-QoS-Param += “meter-class-rate cls-INET rate-absolute 2000”
Dynamic-QoS-Param += “meter-class-burst cls-INET 250000”
------------------------------------------------------------
```

Точно таким же способом можно устанавливать значения скоростей для других классов, если это необходимо. Если скоростные параметры для класса не передаются с RADIUS, то применяются значения указанные в конфигурации.

Параметры скорости для класса можно поменять для сессии при помощи RADIUS CoA, для этого нужно включить в пакет CoA название контекста, где находится данная сессия и имя пользователя, а также новые значения `Dynamic-QoS-Param`. Если по каким-то причинам `User-Name` не устраивает, то его можно заменить на `Framed-IP-Address` или `Acct-Session-Id`. Пример:

```
--------------------------RADIUS-----------------------------
Framed-IP-Address = 10.193.224.222
```

```
Dynamic-QoS-Param += “police-class-rate cls-INET rate-absolute 2000”
Dynamic-QoS-Param += “police-class-burst cls-INET 250000”
Dynamic-QoS-Param += “meter-class-rate cls-INET rate-absolute 2000”
Dynamic-QoS-Param += “meter-class-burst cls-INET 250000”
-------------------------------------------------------------
```

3 Forward Policy

Forward Policies предоставляют возможности для организации функций HTTP Redirect, Policy Based Routing (PBR), Traffic Mirroring. Рассмотрим наиболее популярные применения forward policy – это HTTP Redirect и PBR. Любая Forward Policy использует точно такую же модель классов, как и QoS Policy.

3.1 Пример использования HTTP Redirect

Задача HTTP Redirect в общем случае, это перенаправление запросов браузера абонента на специальный сайт. Самый общий случай это способ сообщения забывчивому абоненту о том, что средства на его счете израсходованы и необходимо его пополнение. С потребительской точки зрения это выглядит приблизительно следующим образом. Абонент входит в сеть привычным для него способом (запуская PPPoE или L2TP соединение, или аутентификация пользователя происходит неявно, как в случае с CLIPS). Далее абонент запускает браузер, вбивает в адресную строку что-то, но его запрос перенаправляется на сайт личного кабинета, где ему сообщается о том, что воспользоваться услугой он не может до тех пор, пока не пополнит счет. Посмотрим, как это может быть реализовано в SmartEdge.

Первое, необходимо определить, те ресурсы, которые абоненту доступны в момент, когда его аккаунт заблокирован. Как минимум абоненту должны быть доступны DNS сервера, для того чтобы была возможность резолвить url в адреса. А также сама страница личного кабинета, например, http://stats.domonet.ru. Иногда к этому списку добавляются дополнительные ресурсы, например сторонние системы online оплаты и прочее.

Второе, нужно определить тот трафик, который будет подлежать redirect. Как правило, это все TCP соединения, обращающиеся на порт 80, за исключением обращений на хосты личного кабинета и прочих ресурсов которые позволены абоненту, когда он заблокирован.

Третье, определение всего остального трафика, предположим, что весь остальной трафик мы собираемся сбрасывать.

Для описания данных условий мы будем использовать Policy ACL, определив три типа (класса) трафика. Допустим, что за http://stats.domonet.ru скрывается адрес 195.14.50.26, а DNS сервера имеют адреса 195.14.50.1 и 195.14.50.2.

```
-------------------------SmartEdge-----------------------
!
policy access-list acl-for-http-redirect
 seq 10 permit udp any host 195.14.50.1 eq domain class cls-NORMAL
 seq 20 permit udp any host 195.14.50.2 eq domain class cls-NORMAL
 seq 30 permit tcp any host 195.14.50.26 eq www class cls-NORMAL
 seq 40 permit tcp any any eq www class cls-REDIRECT
 seq 50 permit ip any any class cls-DROP
!
-------------------------------------------------------------
```

Итак, классы определены. Теперь к этим классам нужно назначить действия. Это определяется при помощи forward policy.

```
-------------------------SmartEdge-----------------------
!
forward policy HTTP-REDIRECT
 access-group acl-for-http-redirect ctx-name
  class cls-NORMAL
  class cls-REDIRECT
   redirect destination local
  class cls-DROP
   drop
!
-------------------------------------------------------------
```

Команда redirect destination local указанная в качестве действия для класса трафика cls-REDIRECT означает, что весь трафик, попадающий под данный класс должен перенаправляться на локальный HTTP сервер самого SmartEdge. Таким образом, необходимо этот сервер активировать. Выполняется это так:

```
-------------------------SmartEdge-----------------------
!
http-redirect server
 port 80
!
-------------------------------------------------------------
```

Осталось задать этому серверу параметры перехода (редиректа). Эти параметры определяются в так называемом http redirect профиле.

```
-------------------------SmartEdge-----------------------
```

```
!
 http-redirect profile CABINET
  url "http://stats.domonet.ru"
!
-------------------------------------------------------------
```

Теперь необходимо определится, каким образом эта функция будет применятся к абоненту. Для того чтобы применить функции HTTP Redirect на subscriber circuit возможны следующие варианты:

1. Передать все необходимые атрибуты из RADIUS в access accept и CoA

```
--------------------------RADIUS-----------------------------
HTTP-Redirect-Profile-Name = “CABINET”
Forward-Policy = “in:HTTP-REDIRECT”
-------------------------------------------------------------
```

Где `HTTP-Redirect-Profile-Name` и `Forward-Policy` это Redback (vendor id 2352) VSA за номерами 107 и 92 соответственно, тип полей – string.

2. Поместить атрибуты в какой-либо абонентский профиль в конфигурации SmartEdge

```
--------------------------SmartEdge--------------------------
!
 subscriber profile profile-with-redirect
  http-redirect profile CABINET
  forward policy CABINET in
!
-------------------------------------------------------------

--------------------------RADIUS-----------------------------
Sub-Profile-Name = “profile-with-redirect”
-------------------------------------------------------------
```

Где `Sub-Profile-Name` это Redback (vendor id 2352) VSA за номером 91 тип поля – string.

3. Поместить атрибуты в профиль по умолчанию, тогда данные атрибуты будут применяться ко всем сессиям, которые принимают состояние UP:

```
--------------------------SmartEdge--------------------------
```

```
!
 subscriber default
  http-redirect profile CABINET
  forward policy CABINET in
!
-------------------------------------------------------------
```

В этом случае с RADIUS ничего не передается для редиректа.

Для того чтобы выключить функцию перенаправления при помощи CoA, например, необходимо послать следующий CoA запрос:

```
--------------------------RADIUS-----------------------------
Forward-Policy = “in:”
-------------------------------------------------------------
```

Атрибут `HTTP-Redirect-Profile-Name` снять по CoA нельзя, он останется до разрушения сессии, при этом он ни на что не влияет. Но при необходимости значение этого атрибута можно переписать на другое.

Интересной особенностью включения функций HTTP Redirect через абонентские профили или профиль по умолчанию, является возможность применения функций перенаправления только один раз. Т.е. с потребительской точки зрения это выглядит так. Абонент подключается, ему присваиваются профиль с атрибутами перенаправления. Послав трафик, подлежащий перенаправлению – система перенаправляет этот трафик, и функции перенаправления автоматически снимаются с сессии. Все это происходит без посылок специальных CoA запросов.

```
-------------------------SmartEdge-------------------------
!
 subscriber profile profile-with-redirect
  http-redirect profile CABINET temporary
  forward policy CABINET in
!
-------------------------------------------------------------
```

Конфигурация отличается только наличием специального параметра `temporary`. Данная функция может быть полезна в различных ситуациях, таких как: организация уведомлений/объявлений для абонентов или определенных категорий абонентов, организации адресной рекламы и проч.

Описанный способ организации HTTP перенаправления не позволяет сохранять и передавать изначально запрашиваемый абонентом URL в сторону портала. Для организации функций прозрачного перенаправления, лучше использовать функциональность Destination NAT, когда модифицируется destination IP адрес отправляемых абонентом пакетов (о том, как настроить NAT см. ниже).

3.2 Пример использования Forward Policy с функциями PBR

Настройка Forward Policy для организации функциональности Policy Based Routing в SmartEdge осуществляется достаточно просто. Forward Policy для PBR также как и любая forward-policy может использовать, либо не использовать классы. Froward Policy применятся к любой circuit – port, vlan, stacked vlan, subscriber. В каждый момент времени к circuit может быть применена только одна Forward Policy. Если в Forward Policy используются классы, то для каждого класса можно определять индивидуальные действия над трафиком этого класса. Если классы не заданы, то действие, указанное в Forward Policy применяется ко всему трафику, передаваемому в circuit, к которому данная полиси применена.

Стандартный пример, есть маршрутизатор таблица которого наполнена маршрутами (статическими и/или динамическими), нам необходимо настроить policy, которая позволяла бы маршрутизировать определенный трафик по правилам отличным от того, что находится в таблице маршрутизации. Например, таблица маршрутизации показывает, что подсеть 192.168.100.0/24 достижима через next-hop 10.177.1.2. Нам по каким-то причинам необходимо трафик, имеющий в качестве destination любой адрес из подсети 192.168.100.0/24, но адресованный на tcp порт 8080 направлять на next-hop 10.199.1.55, весь остальной трафик к 192.168.100.0/24 должен маршрутизироваться согласно таблице маршрутизации (т.е. через next-hop 10.177.1.2).

Для того чтобы это настроить, сначала нужно определить классы, согласно нашему примеру нам нужно определить два класса:

```
--------------------------SmartEdge-------------------------
!
context local
!
policy access-list acl-classess-PBR1-in
 seq 10 permit tcp any 192.168.100.0 0.0.0.255 eq 8080 class cls-CLASS1
 seq 20 permit ip any any class cls-DEFAULT
!
-------------------------------------------------------------
```

Теперь точно также, как и в случае с HTTP Redirect, создаем Forward Policy, только действия для каждого класса указываем согласно нашей задаче.

```
-------------------------SmartEdge-----------------------
!
forward policy fwd-PBR1
 access-group acl-classess-PBR1-in local
  class cls-DEFAULT
  class cls-CLASS1
   redirect destination next-hope 10.199.1.55
!
-------------------------------------------------------------
```

Таким образом, полиси имеет два класса, класс `cls-DEFAULT` мы определили как касс без какого-либо action, т.е. тип трафика принадлежащий данному классу будет маршрутизироваться так как, это определяет таблица маршрутизации. Класс `cls-CLASS1` содержит action, который определяет, что трафик данного класса должен маршрутизироваться на next-hope 10.199.1.55.

Команда `redirect destination next-hope` может содержать до 8-ми значений next-hope, порядок адресов определяет их приоритет, если первый в списке next-hope становится недоступен, то полиси начинает использовать следующий и т.д. Если более приоритетный next-hope становится недоступен, то время от времени SmartEdge проверяет стал ли он доступен и при его доступности начинает маршрутизировать на него вновь. Команда может быть дополнена опциональным словом `default`, например `redirect destination next-hope 10.199.1.55 default`. В таком случае, при недоступности 10.199.1.55 будет использоваться нормальный режим маршрутизации согласно routing table. Если опциональное слово `default` стоит первым в списке, то последующий next-hope будет использовать только в том случае, если нормальная маршрутизация не возможна.

После того как Forward Policy поределена, остается ее применить к circuit. Forward Policy может применятся к абонентской сессии такими же способами как в случае HTTP Redirect, например,

```
-------------------------RADIUS-----------------------------
Forward-Policy = “in:fwd-PBR1”
------------------------------------------------------------
```

Где `Forward-Policy` это Redback (vendor id 2352) VSA за номером 92, тип поля – string.

Говоря о применении Forward Policy на абонентские сессии следует упомянуть, что ее применение также возможно при помощи конструкций `subscriber default` и `subscriber profile name`, о которых уже упоминалось в этом документе.

Forward Policy можно применять к обычным circuit, таким как порт, dot1q pvc и проч.

```
-------------------------SmartEdge-----------------------
!
port ethernet 3/2
 bind interface to-nevada local
 forward policy fwd-PBR1 in
!
-------------------------------------------------------------
```

Froward Policy можно применять в любом направлении: in и out.

В примере выше была продемонстрирована простейшая конфигурация для L3 полиси. Forward Policy может также работать на уровне L2. Дополнительно, в качестве action помимо PBR (`redirect destination next-hope`), можно использовать функции зеркалирования (mirror `destination`). Более подробно об этом можно прочитать в документации.

4 NAT Policy

SmartEdge поддерживает трансляцию сетевых адресов NAT (Network Address Translation). В этом параграфе будут рассмотрены наиболее актуальные способы организации трансляции сетевых адресов:

- NAPT при использовании SmartEdge как BRAS
- Static 1-to-1 трансляции
- Destination NAT как способ организации функций L3 Redirect

4.1 NAPT при использовании SmartEdge как BRAS

NAPT – Network Address Port Translation функциональность, также известная как NAT Overload или NAT N-to-1. NAPT позволяет организовать доступ к сети Интернет для частных приватных адресов через небольшую группу так называемых белых адресов.

Реализация NAPT в SmartEdge имеет определенную специфику и обладает следующими преимуществами:

1. NAT является функцией Forwarding Plane и зависит только от возможностей линейных модулей. Модули поколения PPA2/PPA3 поддерживают до 1M одновременных трансляций на модуль. Таким образом, максимальное количество активных трансляций определяется количеством линейных модулей в системе. При работе NAT ресурсы Control Plane не расходуются.

2. Реализация NAT в SmartEdge полностью аппаратная и не влияет на производительность (pps).

3. Функциональность NAPT для абонентов активируется посредством вовлечения модуля AAA, это означает, что NAPT можно включать/отключать для абонентских сессий персонализировано в одном и том же сегменте доступа.

4. Абонентский NAPT может работать совместно с абонентским NetFlow, статистика, отправляемая в коллектор, будет содержать серые адреса абонентских flow и реальные адреса сети Интернет.

При использовании NAPT в SmartEdge также следует помнить об особенностях текущей его реализации и ограничениях:

1. Текущая реализация NAPT в SEOS поддерживает только три протокола: TCP, UDP и ICMP. Таким образом, на сегодняшний день через NAPT на SmartEdge не будут работать, например, VPN-туннели.

2. В NAPT пулах не поддерживается функциональность address pairing. NAPT пул может состоять из группы реальных адресов или являться небольшой подсетью. Как будет показано дальше, NAPT пул может быть назначен на какую-то группу серых подсетей. Из-за того, что отсутствует address pairing, со временем возникает ситуация когда сессия абонента может быть транслирована через два разных IP адреса NAPT пула. Это может вызвать проблемы на стороне абонента при работе с некоторыми ресурсами в сети Интернет, особенно такими, которые требуют реавторизации (web-почта, порталы и прочее). Функциональность Address Pairing будет добавлена в SEOS в 2011 году, также ниже описан workaround позволяющий обойти отсутствие address paring в текущих версиях SEOS.

3. IP адреса, используемые в NAPT пулах не должны пересекаться с адресными пространствами L3 интерфейсов контекста маршрутизации где они определены.

NAPT в SmartEdge состоит из следующих сущностей:
- NAPT Pool
- NAT Policy

NAPT Pool представляет собой один или группу реальных адресов. Каждый реальный адрес в NAPT пуле может быть разбит на 16 групп портов для трансляций – так называемых port-block-ов, по 4096 портов в каждом блоке. При работе NAPT в маршрутизируемых пакетах переписывается source IP адрес и порт, эта связка в терминологии NAT назевается endpoint, который запоминается и хранится в FIB. Приходящий обратный трафик на этот end-point позволяет сделать обратную трансляцию в нужный серый адрес. Это основной принцип работы NAPT.

Типовая настройка NAPT пула в SmartEdge выглядит следующим образом:

```
--------------------------SmartEdge-------------------------
!
 ip nat pool NAPT-pool-1 napt multibind
  address 155.53.3.1/32 port-block 1 to 15
!
------------------------------------------------------------
```

Данный пример создает NAPT пул, состоящий из одного реального IP адреса, для которого доступны 15 port-block-ов. При этом не используется порт блок 0. Такая конфигурация сделана для того, чтобы исключить использование низких портов для формирования трансляций. Порт блок 0 – это диапазон портов от 0 до 4095. Некоторые ресурсы в сети Интернет (интернет-банки, личные кабинеты и проч.) не позволяют устанавливать tcp соединения с низких well-known source портов по причинам безопасности (в unix системах, например, открыть tcp сессию с well-known source порта может только root). Таким образом, емкость пула в указанном выше примере составляет 15*4096=61440 возможных трансляций.

Как уже упоминалось ранее, NAPT Pool может иметь группу адресов:

```
--------------------------SmartEdge-------------------------
!
 ip nat pool NAPT-pool-1 napt multibind
  address 155.53.3.1/32 port-block 1 to 15
  address 155.53.3.2/32 port-block 1 to 15
  address 155.53.3.3/32 port-block 1 to 15
  address 155.53.3.4/32 port-block 1 to 15
!
------------------------------------------------------------
```

После того как созданы пулы, осталось настроить NAT Policy. NAT Policy по принципу настройки мало чем отличается от Forward и QoS Policy. Также как и любая policy, NAT Policy может состоять из классов или не иметь классов. Ниже представлен пример просто NAT Policy без классов:

```
--------------------------SmartEdge-------------------------
!
```

```
 nat policy nat-policy-1
  pool NAPT-pool-1 local
!
-------------------------------------------------------------
```

Теперь для того чтобы применить NAT полиси к абонентской сессии? Как вариант, можно включить следующий атрибут в access-accept или в CoA запрос.

```
-------------------------RADIUS-----------------------------
NAT-Policy-Name = “nat-policy-1”
-------------------------------------------------------------
```

Где NAT-Policy-Name это Redback (vendor id 2352) VSA за номером 105 тип поля – string.

Для назначения Nat Policy на абонентскую сессию работают любые варианты передачи информации в AAA модуль: access-accept, CoA, конструкции `subscriber default` и `subscriber profile name` (см. разделы выше).

Теперь, зная базовый способ формирования NAPT, рассмотрим более реалистичный пример. Предположим, подсеть, из которой назначаются адреса абонентам – 10.193.0.0/20. Необходимо организовать трансляцию этой подсети через 4 реальных IP адреса. Пример конфигурации может выглядеть следующим образом:

```
-------------------------SmartEdge-------------------------
!
context local
!
 ip nat pool NAPT-pool-1 napt multibind
  address 155.53.3.1/32 port-block 1 to 15
!
 ip nat pool NAPT-pool-2 napt multibind
  address 155.53.3.2/32 port-block 1 to 15
!
 ip nat pool NAPT-pool-3 napt multibind
  address 155.53.3.3/32 port-block 1 to 15
!
 ip nat pool NAPT-pool-4 napt multibind
  address 155.53.3.4/32 port-block 1 to 15
!
!
 policy access-list NAT-acl
  seq 10 permit ip 10.193.0.0 0.0.3.255 class CLASS1
  seq 20 permit ip 10.193.4.0 0.0.3.255 class CLASS2
  seq 30 permit ip 10.193.8.0 0.0.3.255 class CLASS3
```

```
  seq 40 permit ip 10.193.12.0 0.0.3.255 class CLASS4
!
 nat policy Dyn-NAPT-1
! Default class
  ignore
! Named classes
  access-group NAT-acl
   class CLASS1
    pool NAPT-pool-1 local
    endpoint-independent filtering udp
   class CLASS2
    pool NAPT-pool-2 local
    endpoint-independent filtering udp
   class CLASS3
    pool NAPT-pool-3 local
    endpoint-independent filtering udp
   class CLASS4
    pool NAPT-pool-4 local
    endpoint-independent filtering udp
!
-------------------------------------------------------------
```

Теперь передавая в модуль AAA название NAT Policy – “`Dyn-NAPT-1`” при аутентификации абонентов из подсети 10.193.0.0/20, эта полиси будет назначаться на абонентские сессии, и в зависимости от того, к какому классу относится source ip абонента, полиси будет осуществлять трансляцию через соответствующий пул. Каждый пул в данной конфигурации – это 61440 потенциальных трансляций и таким образом, для всей полиси данного примера кол-во потенциальных трансляций 245760. Данный пример раскрывает NAT-механику в SemartEdge немного шире, но все равно не полностью. Несколько слов следует уделить команде `endpoint-independent filtering udp`, которая присутствует в каждом классе. Классический NAT является также своего рода файрволом, когда какая-либо трансляция создает endpoint для source информации пользовательского пакета, то классический NAT также сохраняет информацию о том, какой был destination IP/порт и обратный трафик, приходящий на созданный endpoint должен идти обязательно с этого destination IP/порт (т.е. иметь эти значений в source). Этот режим согласно RFC 4787 является endpoint depended. Если не выключить этот режим работы для udp трафика, то на практике перестанет работать voip, могут возникнуть проблемы с multimedia трафиком, ухудшится обнаружение многих игровых серверов, могут возникнуть проблемы с некоторыми online играми и проч.

На практике может возникнуть необходимость в подстройке времени жизни трансляций для всех поддерживаемых протоколов и состояний tcp. А также может пригодиться возможность ограничивать каждого абонента в кол-ве создаваемых трансляций. Контроль всех этих параметров в NAT Policy выглядит приблизительно следующим образом.

```
--------------------------SmartEdge-------------------------
!
 nat policy Dyn-NAPT-1
  connections tcp maximum 2000
  connections udp maximum 2000
  connections icmp maximum 20
! Default class
  ignore
! Named classes
  access-group NAT-acl
   class CLASS1
    pool NAPT-pool-1 local
    timeout tcp 18000
    timeout udp 60
    timeout fin-reset 60
    timeout icmp 30
    timeout syn 60
    admission-control tcp
    admission-control udp
    admission-control icmp
    endpoint-independent filtering udp
   class CLASS2
    pool NAPT-pool-2 local
    timeout tcp 18000
    timeout udp 60
    timeout fin-reset 60
    timeout icmp 30
    timeout syn 60
    admission-control tcp
    admission-control udp
    admission-control icmp
    endpoint-independent filtering udp
   class CLASS3
    pool NAPT-pool-3 local
    timeout tcp 18000
    timeout udp 60
    timeout fin-reset 60
    timeout icmp 30
    timeout syn 60
    admission-control tcp
    admission-control udp
    admission-control icmp
    endpoint-independent filtering udp
   class CLASS4
    pool NAPT-pool-4 local
    timeout tcp 18000
    timeout udp 60
    timeout fin-reset 60
```

```
    timeout icmp 30
    timeout syn 60
    admission-control tcp
    admission-control udp
    admission-control icmp
    endpoint-independent filtering udp
!
------------------------------------------------------------
```

Абонент, которому будет назначена данная NAT Policy, не сможет создавать более 2000 активных трансляций по tcp, такое же кол-во по udp, а также будет ограничен 20-тью трансляциями для протокола icmp. Параметры Admission Control действую индивидуально для сессий, а не суммарно для всех абонентов, которые привязаны к данной полиси/классу в случае BRAS приложения.

Масштабирование NAPT в отсутствии функции address pairing осуществляется за счет увеличения количества классов в самой полиси, но при этом следует помнить об ограничении, общем для всех типов Policy в SmartEdge – каждая Policy не может иметь более 8-ми классов. При статичной привязке серых подсетей к классам, а соответственно к NAPT пулам, единственным способом масштабировать конфигурацию является создание дополнительных NAT полисей, их кол-во в конфигурации не имеет пределов. Единственное, о чем нужно помнить – это то, что в модульных SE каждый модуль рассчитан на 1M активных трансляций. Один и тот же NAPT Pool, а, следовательно, адрес может быть использован несколькими линейными модулями, т.к. линейный модуль работает с port-block-ами, а не с пулами как таковыми.

Описаний выше способ не всегда может давать равномерное распределение абонентов по классам (пулам). До появления функции address pairing в NAPT пулах (что должно произойти в течение 2011 года), можно использовать следующий workaround. Например, стоит задача создать конфигурацию на 1M активных трансляций, при этом не хотелось бы делать какие-то жесткие привязки серых подсетей к определенным классам (пулам), но при этом каждый абонент должен транслироваться строго через один и тот же адрес за время жизни его сессии, т.е. при следующем подключении адрес может быть другим, но опять таки неименным в течении всего сеанса. Можно создать конфигурацию, в которой каждой NAT Policy соответствует один NAPT Pool с одним единственным IP адресом без определения каких-либо классов. Кол-во таких полиси можно выбрать любым подходящим. Если с радиуса передавать рандомно названия этих полисей, то распределение абонентов по ним будет равномерным и такая схема очень легко масштабируется. Ниже представлен пример на 1M трансляций через 32 реальных IP адреса. Из конфига вырезаны все подстройки для admission control endpoint filtering и таймеры, для того чтобы не загромождать пример.

```
--------------------------SmartEdge-------------------------
!
 ip nat pool NAPT-pool-1 napt multibind
  address 155.53.3.1/32 port-block 9 to 15
!
 ip nat pool NAPT-pool-2 napt multibind
  address 155.53.3.2/32 port-block 9 to 15
!
 ip nat pool NAPT-pool-3 napt multibind
  address 155.53.3.3/32 port-block 9 to 15
!
!* <snip>*
!
 ip nat pool NAPT-pool-32 napt multibind
  address 155.53.3.32/32 port-block 9 to 15
!
!
 nat policy 1
  pool NAPT-pool-1 local
!
 nat policy 2
  pool NAPT-pool-2 local
!
 nat policy 3
  pool NAPT-pool-3 local
!
!* <snip>*
!
 nat policy 32
  pool NAPT-pool-32 local
!
-------------------------------------------------------------
```

Таким образом, передавая для `NAT-Policy-Name` со стороны RADIUS случайное число от 1 до 32 на каждый запрос на аутентификацию можно равномерно распределять абонентов по 32ум пулам. Т.к. каждый NAPT Pool в этом примере содержит 7 порт-блоков, то каждый пул может дать 7*4096=28672 трансляций. Всего в конфигурации 32 полиси/пула, т.е. суммарно это дает 32*28672=917504 трансляций. Добавляя/убавляя пулы и соответствующие им полиси из конфига можно легко масштабировать NAPT на коробке в целом.

Еще один важный момент для работы NAPT в SmartEdge. В начале этого параграфа говорилось о том, что IP адреса, используемые в NAPT пулах не должны попадать ни в одну из подсетей интерфейсов контекста маршрутизации, где они используются. На практике это не создает никаких дополнительных проблем, т.к. для того чтобы анонсировать адреса NAPT пулов в любой из протоколов маршрутизации достаточно использовать команду ‘`redistribute nat`’. Когда какому-либо абоненту назначается NAT полиси и он попадает в соответствующий NAPT Pool, то если эта команда присутствует в конфигурации протокола маршрутизации (RIP, OSPF, ISIS, BGP), адрес/префикс NAPT пула начинает анонсироваться в этот протокол.

4.2 Static 1-to-1 трансляции

SmartEdge поддерживает 1-to-1 NAT, т.е. когда каждому приватному IP адреса ставится во взаимооднозначное соответствие уникальный публичный IP адрес. Такие правила статичны и прописываются в конфигурации самой полиси. Для организации статичной NAT полиси не нужны пулы.

Пример 1-to-1 NAT полиси имеет следующий вид:

```
--------------------------SmartEdge-------------------------
!
 nat policy static-NAT-1
! Default class
  drop
! Static rules
  ip static in source 10.193.1.1 155.53.3.1
  ip static in source 10.193.1.2 155.53.3.2
  ip static in source 10.193.1.3 155.53.3.3
  ip static in source 10.193.1.4 155.53.3.4
  ip static in source 10.193.1.5 155.53.3.5
! **etc**
!
------------------------------------------------------------
```

Такая полиси может быть применена к абонентской сессии аналогичными для Policy способами и тогда и тогда адрес абонента будет транслироваться согласно соответствующей записи в этой полиси. Можно иметь много полиси в конфиге (количество не ограничено). Рекомендуется иметь не более 256 записей в каждой полиси (можно больше, но тогда нужно сначала протестировать).

Записи в самой Policy не обязательно должны быть упорядочены как в примере с точки зрения продолжительности подсетей и проч. И могу идти вразнобой.

При таком способе организации NAT подсчет трансляций ведется по-другому: каждая статичная запись равноценна одной трансляции, неважно, сколько 5-tuple реально проходит через каждую трансляцию (в отличие от NAPT).

4.3 Destination NAT как способ организации функций L3 Redirect

В первом разделе этого HowTo был рассмотрен пример конфигурации HTTP Redirect. Иногда такой способ перенаправления абонента не всегда устраивает конечно потребителя, основной проблемой является факт модификации исходного URL пользователя. Когда произошел redirect абонента на портал или в личный кабинет и абонент закончил там работу, то нет возможности вернуть его на исходную страницу, если только абонент сам не нажмет кнопу back своего браузера.

В SE существует возможность создать NAT полиси, которая меняет не source, а destination адрес абонентских пакетов и после этого делает FIB-lookup для измененного destination. Это позволяет прозрачно перенаправить например входящий трафик абонента. Такие правила можно создавать индивидуально для классов (тем самым, указав только трафик, посланный на порт 80, например), либо определять целиком для полиси, если в ней нет классов.

Простой пример настройки destination NAT

```
--------------------------SmartEdge-------------------------
!
 nat policy DestNAT
! Default class
  ignore
  destination 155.53.15.15
!
------------------------------------------------------------
```

Назначив данную полиси на circuit будет происходить подмена destination ip адреса всех входящих пакетов на указанный.