Павлов
Виталий Викторович,
заместитель начальника Вычислительного Центра информационного центра УМВД России по Забайкальскому краю,
подполковник внутренней службы

Построение регионального сегмента единого информационного пространства на основе GRE-тоннелей. Защита информации, обеспечение устойчивой телефонной/селекторной связи, повышение пропускной способности «узких» каналов

Региональный сегмент интегрированной мультисервисной сети передачи данных (ИМТС) представляет собой объединение территориальных органов внутренних дел в единую корпоративную сеть с центром в региональном управлении внутренних дел, имеющим, в свою очередь, подключение к ЕИТКС МВД.

В настоящий момент построение региональных ИМТС уже завершено и базируется на аренде магистральных каналов связи локальных поставщиков услуг связи (далее провайдеров).

Однако активное внедрение в 2011–2012 годах СМЭВ и включение в систему предоставления государственных слуг в электронном виде отделов и отделений полиции на районном уровне предъявило новые требования к функционированию региональных ИМТС.

Данная статья наиболее актуальна для регионов, в которых значительная часть ИМТС построена на арендуемых каналах связи, что в свою очередь ставит необходимость постоянного общения с техническим персоналом провайдера.

ИМТС регионов построены по следующей технологии: для пользовательских сетей выделен диапазон ip-адресов 10.x.x.x, для технологических сетей, связывающих пользовательские, выделен диапазон 172.x.x.x. При этом, провайдер, предоставляющий каналы связи, создает на своем коммутационном оборудовании статические либо динамические таблицы маршрутизации, описывающие маршруты до конечных пользовательских сетей, в том числе до сетей федерального уровня.

Недостатки существующей схемы:

— открытость маршрутизации для компаний провайдеров, что дает возможность «злоумышленнику», имеющему доступ к коммутациям провайдера либо работающему у провайдера, получения прямого доступа к ведомственному внутрисетевому трафику. Это не является критичным, если канал криптуется конечным оборудованием органов внутренних дел. Однако далеко не все регионы были обеспечены достаточным количеством криптошлюзов;

— необходимость постоянного взаимодействия с провайдером при внесении изменений в расположение пользовательских сетей либо при необходимости обеспечения выхода какого-либо районного подразделения на корпоративные сети федерального уровня, что наиболее актуально в момент организации доступа подразделений к СМЭВ;

— невозможность выделения приоритета голосового трафика без участия специалистов провайдера. Качество ip-телефонии и селекторной связи между подразделениями является одной из приоритетных задач и доставляет немало хлопот сотрудникам региональных ЦИТСиЗИ.

Для решения указанных проблем была выбрана схема построения сети с организацией тоннелей через технологическое пространство провайдера. В таком случае провайдер должен обеспечить только прохождение пакетов через свое технологическое пространство от конечных технологических точек подключения подразделений органов внутренних дел (далее ОВД). Реальная организация пользовательской сети для него закрыта. Маршрутизацией внутренних сетей и их полноценным администрированием занимаются соответствующие специалисты ОВД.

Данная концепция была реализована на базе GRE-тоннелей (General Routing Encapsulation). Достоинство данного протокола в том, что он поддерживается практически всеми unix/linux-системами и большей частью специализированного коммутационного оборудования, в частности оборудованием компании Cisco. Собственно Cisco и разработала данный протокол. Принцип работы GRE-протокола хорошо описан в различных Интернет-источниках, поэтому в данной статье рассматриваться не будет.

Между двумя маршрутизаторами A и B находится несколько маршру-

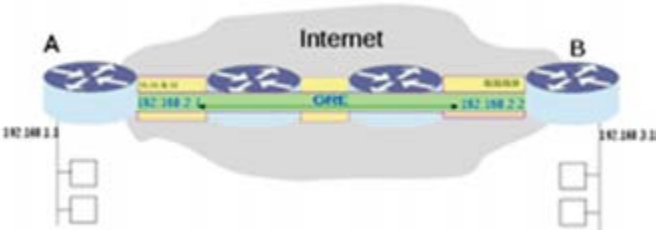

Рис. 1. Пример работы GRE туннеля

Рис. 2. Схема организации сети удаленного подразделения с ЕИТКС через GRE-тоннель

тизаторов поставщика услуг связи. Туннель позволяет обеспечить связь между сегментами сети 192.168.1.0/24 и 192.168.3.0/24 так, как если бы маршрутизаторы A и B были соединены прямым линком.

В качестве корневого оборудования регионального УМВД (ГУВД) субъекта РФ были выбраны маршрутизаторы компании Cisco, хотя данную роль может выполнить любой самосборный PC-роутер на платформе Linux.

В качестве оборудования для удаленных подразделений пригоден достаточно большой диапазон роутеров, продающихся в розничной сети в диапазоне цен до 3 тысяч рублей! В частности были использованы роутеры LinkSYS WRT54GL, LinkSYS WRT160NL, TP-link TL-WR1043ND. Последнюю модель можно встретить в продаже практически в любом компьютерном магазине.

Стандартная прошивка роутеров не пригодна для наших целей. В качестве альтернативной прошивки выбрана свободно распространяемая OpenWRT, доступная на сайте openwrt.org. В зависимости от модели роутеров использовались версии Whiterussian RC5 и BackFire 10.03.1-rc6.

После перепрошивки роутера мы получаем возможность работы не только со стандартными функциями через web-интерфейс, но и возможность установки необходимых пакетов и скриптов и их гибкой настройки посредством telnet- или ssh-сессии. Первичную настройку wan-порта, смотрящего в сторону провайдера, и lan-порта, в свою очередь смотрящего в сторону внутренней сети подразделения, можно произвести через интуитивно-понятный web-интерфейс.

Рассмотрим пример конфигурирования на конкретном подразделении.

Исходные данные:

- технологический адрес роутера (выделенный провайдером), в нашем случае 192.168.2.42/30;
- default gateway, то есть технологический адрес шлюза на стороне провайдера для роутера подразделения, в нашем случае 192.168.2.41/30;
- адрес роутера, являющийся шлюзом для локальной сети подразделения, в нашем случае 10.10.10.118/24;
- технологический адрес шлюза, смотрящего в сторону провайдера для роутера УМВД/ГУВД, в нашем случае 172.172.172.100/29.

Создаем файл сетевой конфигурации и поднятия тоннельного интерфейса S90tun_p21 (имя файла носит произвольный характер) для конкретного подразделения в соответствии с исходными данными:

```
echo 'Network configures
echo 'internal Lan '
nvram set lan_ipaddr=10.10.10.118
#Этот адрес будет являться шлюзом для
внутренней сети удаленного подразделения
nvram set lan_netmask=255.255.255.0
echo 'network to Provider'
ip addr add 192.168.2.42/30 dev vlan1
#Это технологический адрес выделенный
провайдером для роутера подразделения
ip route add 172.172.172.100/29 via
192.168.2.41
# 172.172.172.100 — технологический адрес,
выделенный провайдером для нашего роутера
в региональном УМВД (ГУВД)
# 192.168.2.41 — технологический адрес
на стороне провайдера (шлюз для удаленного
подразделения)

# непосредственно туннель:
echo 'tunnel'
insmod ip_gre
ip tunnel add pos21 mode gre remote
172.172.172.100 local 192.168.2.42 ttl 255
ip link set pos21 up
ip addr add 192.168.201.6/30 dev pos21
#Это адрес тоннельного интерфейса,
назначаемый нами произвольно, либо, если
есть резервы, в соответствии с распределением
технологических сетей.

ip route add default dev pos21
#Устанавливаем умолчательный маршрут через
тоннельный интерфейс.

echo 'disable radio'
#Отключаем радио в целях безопасности, если
необходимо — можно оставить работающим.
nvram set wl_radio=0
echo 'set name of router'
nvram set wan_hostname=Kadala
nvram set wl0_ssid=Kadala
nvram commit
```

Скрипт настройки сетевых интерфейсов для данного подразделения готов.

Создаем скрипт первоначальной конфигурации роутера с установкой необходимых пакетов. Можно и не создавать его, а выполнить вручную. Но, когда надо настроить не один роутер, удобнее использовать скрипт. В нашем случае его имя init_conf5 (название носит произвольный характер).

Содержимое файла:

```
echo 'For get and apply this configures on
LinkSYS
echo 'run next command:
echo ''
echo 'wget http://you_web_server/path/init_
conf5'
echo 'chmod 775 init_conf5'
echo './init_conf5'
cd /tmp
wget http:// my_web_server/ packages5/ ip_2.6.
11-050330-1 _mipsel.ipk
#Здесь my_web_server — это адрес любого
вашего web-сервера, с которого удобнее
производить загрузку пакетов.В данном случае
это 192.168.1.2
ipkg install ip_2.6.11-050330-1_mipsel.ipk
wget http://192.168.1.2/packages5/kmod-
tun_2.4.30-brcm-3_mipsel.ipk
ipkg install kmod-tun_2.4.30-brcm-3_mipsel.ipk
wget http://192.168.1.2/packages5/kmod-
gre_2.4.30-brcm-3_mipsel.ipk
ipkg install kmod-gre_2.4.30-brcm-3_mipsel.ipk
wget
http://192.168.1.2/packages5/libpcap_0.9.4-1_
mipsel.ipk
ipkg install libpcap_0.9.4-1_mipsel.ipk
wget http://192.168.1.2/packages5/kmod-
sched_2.4.30-brcm-3_mipsel.ipk
ipkg install kmod-sched_2.4.30-brcm-3_mipsel.
ipk
wget
http://192.168.1.2/packages5/ tc_2.6.11-050
330-1_mipsel.ipk
ipkg install tc_2.6.11-050330-1_mipsel.ipk
echo '========================='
echo 'Packages install finished'
echo '========================='
cd /etc/init.d
wget http://192.168.1.2/kadala/S90tun_p21
chmod 775 S90tun_p21
echo '========================='
echo 'Tunnel startup script installed'
echo '========================='
echo 'disable firewall'
cd /etc
cp firewall.user firewall_init.user
>firewall.user
cd /etc/init.d
echo 'change 754 to 644'
chmod 644./S45firewall
nvram commit
echo 'firewall disable finished'
echo 'Now you can reboot you router'
echo 'Warniing! After reboot routers adress will
CHANGED!! '
```

Итак, заходим по ssh на LinkSYS и в консоли выполняем:

```
root@
OpenWrt:~# wget
http://192.168.1.2/kadala/init_conf5
root@OpenWrt:~# chmod 755 init_conf5
root@OpenWrt:~#./init_conf5:
root@OpenWrt:~# reboot
```

Роутер удаленного подразделения готов к эксплуатации. Но нам необходимо решить еще одну сопутствующую задачу — выделить голосовой трафик в приоритетный класс, чтобы обеспечить стабильность голосовой связи независимо от загруженности каналов и ограничить трафик из удаленных подразделений в региональный УМВД, так как в случае вспышек «гриппа» они забивают пакетами центральный роутер.

Для осуществления фильтрации трафика скачиваем и устанавливаем пакеты kmod_sched.ipk, tc.ipk

Подключаем модули, необходимые для работы утилиты tc:

```
root@OpenWrt:/lib/modules/2.4.30# insmod
cls_u32 cls_route sch_sfq sch_htb
```

Добавляем дисциплину обработки очереди

```
root@OpenWrt:~# tc qdisc add dev br0 root
handle 1: htb default 20
```

- dev br0 — Указываем устройство br0, к которому мы подключаем дисциплину обработки очереди.

В данном случае:

- root — Указываем, что это корневая дисциплина, то есть для исходящего трафика. В случае входящего необходимо использовать ingress
- handle 1:0 — Задаем дескриптор в форме старший номер — младший номер. Младший номер для любой дисциплины обработки очереди должен равняться нулю.
- htb — Указываем тип дисциплины обработки очереди, который мы хотим подключить.

Дисциплина HTB (Hierarchical Token Bucket) использует идею токенов. Благодаря классовости, поддержки технологии заема полосы пропускания, HTB позволяет организовывать сложное и тонкое управление трафиком.

Важной составляющей дисциплины HTB является механизм заема полосы пропускания. Подклассы начинают занимать часть полосы пропускания у своих родительских классов только когда трафик превышает значение, заданное параметром rate.

```
root@# tc class add dev br0 parent 1: classid 1:1
htb rate 100mbit ceil 100mbit burst 32k
root@# tc class add dev br0 parent 1:1 classid
1:10 htb rate 32kbit ceil 128kbit burst 16k
root@# tc class add dev br0 parent 1:1 classid
1:20 htb rate 96kbit ceil 128kbit burst 16k
```

Здесь:

- dev br0 — устройство, к которому мы подключаем новый класс.
- parent 1:1 — дескриптор родителя, к которому мы подключаем данный класс.
- classid 1:1 — уникальный дескриптор для данного класса. Младший номер должен быть отличным от нуля.
- htb — классовые дисциплины обработки очередей требуют, чтобы все подклассы были одного типа с родителями. Поэтому у дисциплины обработки очереди HTB классы будут тоже HTB.
- rate 128kbit, ceil 128bit — параметры класса.

Распределяем трафик по классам, с помощью tc filter:

```
root@# tc filter add dev br0 protocol ip par-
ent 1:0 prio 1 u32 match ip dst 172.116.25.7
flowid 1:10
```

Здесь:

- отправляем весь трафик, у которого получатель 172.116.25.7 в класс 1:10.
- protocol ip — Задаем протокол, с которым будет работать фильтр.
- prio 5 — параметр prio позволяет присвоить классифицированному этим фильтром трафику приоритет.
- u32 — классификатор, в данном случае это u32.

В итоге скрипт фильтрации пакетов будет вглядеть следующим образом:

```
tc qdisc add dev br0 root handle 1: htb default 20
tc class add dev br0 parent 1: classid 1:1 htb rate
128kbit ceil 128kbit burst 32k
tc class add dev br0 parent 1:1 classid 1:10 htb
rate 64kbit ceil 128kbit burst 16k
tc class add dev br0 parent 1:1 classid 1:20 htb
rate 64kbit ceil 128kbit burst 16k
tc filter add dev br0 protocol ip parent 1:0 prio 1
u32 match ip dst 172.116.25.167/32 flowid 1:10
```

Здесь 172.116.25.167 — адрес голосового сервера в региональном УМВД (ГУВД).

Таким образом мы поделили канал шириной 128kb пополам. При наличии голосового трафика максимальная ширина полосы для него будет 64kb. При отсутствии трафика класса 10 вся полоса в 128kb будет предоставлена трафику класса 20.

В результате дальнейших эмпирических наблюдений было установлено, что пиковая загрузка голосовым трафиком не превышает 31kb. То есть можно отдать голосу 32kb. Но данная величина в каждом конкретном случае, скорее всего, зависит от типа используемых голосовых шлюзов.

Настройки на стороне регионального узла связи для организации тоннеля с данным удаленным подразделением выглядят следующим образом:

```
interface Tunnel1010
description GRE tunnel Kadala
ip address 10.10.10.118 255.255.255.0
ip tcp adjust-mss 1400
tunnel source GigabitEthernet0/0
tunnel destination 192.168.2.42
……..
ip route 10.10.10.0 255.255.255.0
Tunnel1010
```

Таким образом, мы получили полный контроль над маршрутизацией пользовательских сетей, отсутствие необходимости согласования сетей с провайдером, возможность управления голосовым трафиком, возможность применения гибких правил разграничения доступа к ресурсам ЕИТКС и фильтрации пакетов утилитой iptables, возможность анализа сетевого трафика средствами tcpdump и многое другое. Для обеспечения защиты трафика можно настроить IPsec.