Построение отказоустойчивого решения с применением технологии RRI (reverse route injection); защита мобильного доступа

Описание стенда

Сценарий иллюстрирует построение защищенного соединения между клиентом «С-Терра Клиент» (устройство Client1) и подсетью SN1, которая защищается парой шлюзов безопасности «С-Терра Шлюз» (GW1 и GW2). Устройство Client1 сможет общаться по защищенному каналу (VPN) с устройствами из подсети SN1 (в частности с Host1). Адрес мобильного клиента неизвестен заранее - клиент находится за динамическим NAT-ом. В ходе построения защищенного соединения мобильный клиент получает адрес из заранее определенного на шлюзе пула.

В рамках данного сценария для аутентификации партнеры будут использовать сертификаты. В качестве криптопровайдера будет использована криптографическая библиотека, разработанная компанией «С-Терра СиЭсПи». Шлюзы безопасности «С-Терра Шлюз» версии 4.2. Клиент «С-Терра Клиент» версии 4.2.

Параметры защищенного соединения:

Параметры протокола IKE:

- Аутентификация при помощи цифровых сертификатов, алгоритм подписи – ГОСТ Р 34.10-2012;
- Алгоритм шифрования – ГОСТ 28147-89 (ключ 256 бит);
- Алгоритм вычисления хеш-функции – ГОСТ Р 34.11-2012 ТК26 (ключ 256 бит);
- Алгоритм выработки общего ключа (аналог алгоритма Диффи-Хеллмана) – VKO_GOSTR3410_2012_256 (ключ 256 бит).

Параметры протокола ESP:

- Комбинированный алгоритм шифрования и имитозащиты (контроль целостности) – ESP_GOST-4M-IMIT (ключ 256 бит).

Схема стенда (Рисунок 1):

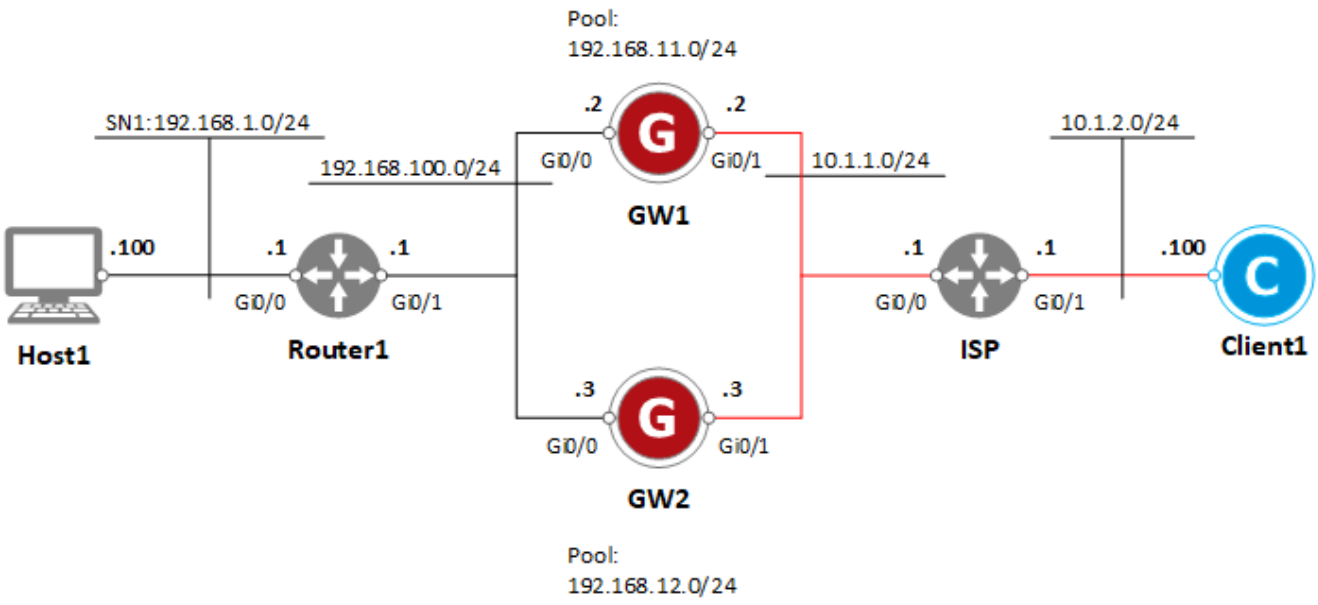

Рисунок 1

Логика работы

Мобильный клиент Client1 может присоединяться как к шлюзу GW1, так и к шлюзу GW2. В настройках мобильного клиента мы указываем подключаться к случайному шлюзу из списка пиров. Таким образом осуществляется балансировка мобильных клиентов. При построении защищенного туннеля устройству Client1 присваивается адрес из пула. Маршрут до мобильного клиента передается при помощи протокола RIP маршрутизатору Router1. В случае если один из шлюзов выходит из строя, туннель от Client1 устанавливается до другого шлюза, который в свою очередь добавляет маршрут и обновляет о нем информацию на Router1 так же посредством RIP.

Аналогично к отказоустойчивому центру могут подсоединяться и шлюзы. Сценарий «Построение отказоустойчивого решения с применением технологии RRI (reverse route injection). Защита филиальной сети».

Настройка стенда

Настройка шлюза безопасности GW1

Начальная настройка шлюза в S-Terra administrative console при первом включении состоит из следующих действий:

- Пройдите процедуру аутентификации (пользователь по умолчанию – `administrator`, пароль по умолчанию – `s-terra`).
- Пройдите процедуру инициализации (команда `initialize`).
- Активируйте политику драйвера по умолчанию (команда `run csconf_mgr activate`).
  - Команда `run csconf_mgr activate` применяет текущую политику драйвера. При первичной настройке шлюза применится политика драйвера по умолчанию, при которой прохождение трафика не блокируется.
- Для доступа через SSH установите пароль на пользователя `root` (команда `run passwd`).

Более подробно консоль разграничения доступа S-Terra administrative console описана в документации.

Настройка интерфейсов

1. Перейдите из консоли разграничения доступа (S-Terra Administrative console) в консоль настройки шлюза (cisco-like интерфейс). По умолчанию имя пользователя – `cscons`, пароль – `csp`:

```
administrator@sterragate] configure
sterragate login: cscons
```

```
Password:
...
sterragate#
```

2. Перейдите в режим настройки:

```
sterragate#conf t
```

```
Enter configuration commands, one per line.  End with CNTL/Z.
```

3. В настройках интерфейсов задайте IP-адреса:

```
GW1(config)#interface GigabitEthernet 0/0
GW1(config-if)#ip address 192.168.100.2 255.255.255.0
GW1(config-if)#no shutdown
GW1(config-if)#exit
GW1(config)#interface GigabitEthernet 0/1
GW1(config-if)#ip address 10.1.1.2 255.255.255.0
GW1(config-if)#no shutdown
GW1(config-if)#exit
```

4. Задайте статические маршруты:

```
GW1(config)#ip route 0.0.0.0 0.0.0.0 10.1.1.1
GW1(config)#ip route 192.168.1.0 255.255.255.0 192.168.100.1
```

5. Выйдите из cisco-like интерфейса:

```
sterragate(config)#end
sterragate#exit
```

Дальнейшую настройку можно проводить через SSH подключение.

Важно! Среда передачи в этом случае должна быть доверенной. Описание создания доверенной среды описано в соответствующей инструкции.

Регистрация сертификата УЦ

В данном сценарии для аутентификации используются сертификаты. Для корректной работы необходимо зарегистрировать доверенный сертификат УЦ и локальный сертификат, выданный данным УЦ.

1. Подключитесь по SSH к шлюзу.

6. Установите правильное системное время.

   Например:

   ```
   root@sterragate:~# date -s "01/31/2017 15:00"
   Tue Jan 31 15:00:00 MSK 2017
   ```

   Данная запись соответствует 31 января 2017 года 15:00.

   В данном случае формат даты указывается в виде месяц/день/год (ММ/ДД/ГГГГ).

   Для автоматической настройки правильного времени рекомендуется настроить NTP-клиент по соответствующей инструкции.

7. Создайте папку `/certs`:

   ```
   root@sterragate:~# mkdir /certs
   ```

8. Перенесите доверенный сертификат УЦ на шлюз.

   Способы передачи данных на шлюз описаны в документации.

9. С помощью утилиты `cert_mgr` зарегистрируйте сертификат в базе продукта:

   ```
   root@sterragate:~# cert_mgr import -f /certs/ca.cer -t
   1 OK C=RU,O=S-Terra,CN=RootCA
   ```

   Ключ `-t` в данной команде указывает на то, что импортируемый сертификат – доверенный сертификат УЦ.

Регистрация локального сертификата

Для регистрации локального сертификата в базе продукта выполните следующие действия:

1. Сформируйте запрос на сертификат при помощи утилиты `cert_mgr`:

   ```
   root@sterragate:~# cert_mgr create -subj "C=RU,O=S-Terra CSP,OU=Research,CN=GW1" -GOST_R341012_256 -fb64 /home/gw_req
   ```

   - Ключ `-subj <DN>` задает поля сертификата.
   - Ключ `-GOST_R341012_256` предполагает использование ГОСТ Р 34.10-2012. На УЦ для его поддержки должно быть установлено СКЗИ «КриптоПро CSP» версии 4.0 или новее. При необходимости, есть возможность использовать старый алгоритм (ГОСТ Р 34.10-94), который задается ключом `-GOST_R3410EL`.
   - Ключ `-fb64 <путь до файла>` позволяет сохранить запрос в файл по указанному пути.

10. Передайте полученный запрос сертификата на УЦ. Процедура выдачи сертификата на УЦ по запросу описана в документации.

11. Зарегистрируйте локальный сертификат в базе продукта, применив утилиту `cert_mgr`:

    ```
    root@sterragate:~# cert_mgr import -f /certs/GW1.cer
    1 OK C=RU,O=S-Terra CSP,OU=Research,CN=GW1
    ```

12. Убедитесь, что сертификаты импортированы успешно:

    ```
    root@sterragate:~# cert_mgr show
    Found 2 certificates. No CRLs found.
    1 Status: trusted C=RU,O=S-Terra,CN=RootCA
    2 Status: local   C=RU,O=S-Terra CSP,OU=Research,CN=GW1
    ```

Создание политики безопасности

После регистрации сертификатов необходимо создать политику безопасности для шлюза GW1

1. Для входа в консоль запустите `cs_console`:

   ```
   root@sterragate:~# cs_console
   sterragate>enable
   Password:
   ```

Пароль по умолчанию – `csp`.

Важно! Пароль по умолчанию необходимо сменить.

2. Перейдите в режим настройки:

```
sterragate#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
```

3. Смените пароль по умолчанию:

```
sterragate(config)#username cscons password <пароль>
```

4. Смените название шлюза:

```
sterragate(config)#hostname GW1
```

5. Задайте тип идентификации:

```
GW1(config)#crypto isakmp identity dn
```

В данном сценарии для идентификации будет использоваться поле DN сертификата.

6. Задайте параметры DPD (dead peer detection)

```
GW1(config)#crypto isakmp keepalive 10 2
GW1(config)#crypto isakmp keepalive retry-count 5
```

Если в течение 10 секунд отсутствует входящий трафик в IPsec туннеле, то с интервалом в 2 секунды посылается 5 keepalive-пакетов в IKE туннеле, чтобы удостовериться в работоспособности туннеля. Если партнер не отвечает на keepalive-пакеты, то существующий IKE туннель переходит в состояние disabled, а связанные с ним IPsec туннели удаляются. В случае наличия исходящего трафика происходит попытка создать новый IKE туннель.

7. Задайте параметры для IKE:

```
GW1(config)#crypto isakmp policy 1
GW1(config-isakmp)#authentication gost-sig
GW1(config-isakmp)#encr gost
GW1(config-isakmp)#hash gost341112-256-tc26
GW1(config-isakmp)#group vko2
GW1(config-isakmp)#exit
```

8. Задайте параметры для IPsec:

```
GW1(config)#crypto ipsec transform-set TSET esp-gost28147-4m-imit
GW1(cfg-crypto-trans)#exit
```

9. Задайте пул из которого будет выдан адрес клиенту:

```
GW1(config)#ip local pool POOL 192.168.11.1 192.168.11.254
```

10. Опишите трафик, который планируется защищать. Для этого создайте расширенный список доступа:

```
GW1(config)#ip access-list extended LIST
GW1(config-ext-nacl)#permit ip 192.168.1.0 0.0.0.255 192.168.11.0 0.0.0.255
GW1(config-ext-nacl)#exit
```

11. Создайте динамическую крипто-карту:

```
GW1(config)#crypto dynamic-map DMAP 1
GW1(config-crypto-map)#match address LIST
GW1(config-crypto-map)#set transform-set TSET
GW1(config-crypto-map)#set pool POOL
GW1(config-crypto-map)#reverse-route
GW1(config-crypto-map)#exit
```

Обратите внимание на опцию `reverse-route`.

12. Привяжите динамическую карту к статической:

```
GW1(config)#crypto map CMAP 1 ipsec-isakmp dynamic DMAP
```

13. Привяжите крипто-карту к интерфейсу, на котором будет туннель:

```
GW1(config)#interface GigabitEthernet 0/1
GW1(config-if)#crypto map CMAP
GW1(config-if)#exit
```

14. Настройте получение списка отозванных сертификатов (CRL) по HTTP:

```
GW1(config)#crypto pki trustpoint s-terra_technological_trustpoint
GW1(ca-trustpoint)#crl download group GROUP http://10.0.221.179/certsrv/certcrl.crl
GW1(ca-trustpoint)#exit
```

Предполагается, что CRL выкладывается на общедоступное место (доступ к которому обеспечен без использования IPsec) для всех шлюзов. При указании имени домена, вместо IP-адреса, необходимо настроить адрес DNS-сервера в системном файле `/etc/resolv.conf`.

Также необходимо учитывать, что у CRL есть срок действия и нужно обеспечивать своевременное их обновление в данном общедоступном месте.

По умолчанию CRL будет запрашиваться раз в сутки (раз в 1440 минут), для изменения интервала запросов можно воспользоваться командой `crl download time <интервал в минутах>`.

При необходимости отключения CRL (не рекомендуется отключать CRL) воспользуйтесь командой `revocation-check none`.

15. Настройка устройства GW1 в cisco-like консоли завершена. При выходе из конфигурационного режима происходит загрузка конфигурации:

```
GW1(config)#end
GW1#exit
```

В приложении представлен текст cisco-like конфигурации и текст LSP конфигурации для шлюза GW1.

Настройка пакета динамической маршрутизации quagga

Динамическая маршрутизация настраивается при помощи пакета Quagga из консоли ОС (linux bash).

1. Примените патч (подходит для всех классов СКЗИ), подготавливающий Quagga к работе. Патч можно скачать по ссылке https://cloud.s-terra.com/index.php/s/g5rq36mMq6miZs6.

   1.1. Перенесите архив `quagga_preparation.tar` в `/root` и распакуйте его:

```
root@GW1:~# tar -xvf quagga_preparation.tar
```

   1.2. Перейдите в директорию `quagga_preparation`:

```
root@GW1:~# cd quagga_preparation/
```

   1.3. Выполните сначала скрипт `preparation.bash`, а потом `install.bash`:

```
root@GW1:~/quagga_preparation# ./preparation.bash
```

```
INFO: Backup "/etc/quagga/daemons" successfully completed!
INFO: log files successfully created!
INFO: "/etc/quagga/zebra.conf" successfully created!
INFO: "/etc/quagga/ospfd.conf" successfully created!
INFO: "/etc/quagga/bgpd.conf" successfully created!
INFO: "/etc/quagga/ripd.conf" successfully created!
INFO: "/etc/quagga/zebra.conf" successfully configured!
INFO: "/etc/quagga/ospfd.conf" successfully configured!
INFO: "/etc/quagga/bgpd.conf" successfully configured!
INFO: "/etc/quagga/ripd.conf" successfully configured!
Starting periodic command scheduler: cron.
update-rc.d: using dependency based boot sequencing
Stopping Quagga monitor daemon: (watchquagga).
Stopping Quagga daemons (prio:0): (bgpd) (ripd) (ospfd) (zebra) (ripngd) (ospf6d)
(isisd).
Removing all routes made by zebra.
Loading capability module if not yet done.
Starting Quagga daemons (prio:10): zebra bgpd ripd ospfd.
Starting Quagga monitor daemon: watchquagga.
```

```
root@GW1:~/quagga_preparation# ./install.bash
```

```
Creating backup for old "/etc/init.d/quagga" file... "/etc/init.d/quagga.old".
Creating backup for old "/etc/quagga/debian.conf" file...
"/etc/quagga/debian.conf.old".
patching file /etc/init.d/quagga
patching file /etc/quagga/debian.conf
INFO: Patch is successfully applied!
```

2. Отредактируйте файл `/etc/quagga/ripd.conf` следующим образом:

```
root@GW1:~# vi /etc/quagga/ripd.conf
```

```
hostname GW1
password csp
log file /var/log/quagga/rip.log debugging
!
router rip
version 2
redistribute kernel
network eth0
distribute-list acl-in in
distribute-list acl-out out
!
access-list acl-in deny any
access-list acl-out permit 192.168.11.0/24
access-list acl-out deny any
line vty
```

- Параметр `network` определяет в какой сегмент сети будут отправляться RIP-пакеты.
- Параметры `distribute-list acl-in in` и `distribute-list acl-out out` определяют фильтрацию на входящие и исходящие RIP-оповещения.

Важно! Шлюз не должен получать маршруты от других устройств, единственной задачей RIP является передача маршрутов об удаленной подсети маршрутизатору Router1.

Также для настройки Quagga можно воспользоваться командой vtysh:

```
rootGW1:~# vtysh
```

```
Hello, this is Quagga (version 1.0.20160315).
Copyright 1996-2005 Kunihiro Ishiguro, et al.
```

```
GW1# terminal length 0
GW1# show ip rip
```

3. Перезапустите Quagga:

```
root@GW1:~# service quagga restart
```

```
Stopping Quagga monitor daemon: watchquagga.
Stopping Quagga daemons (prio:0): bgpd ripd ospfd zebra (ripngd) (ospf6d) (isisd).
Removing all routes made by zebra.
Loading capability module if not yet done.
Starting Quagga daemons (prio:10): zebra bgpd ripd ospfd.
Starting Quagga monitor daemon: watchquagga.
```

4. Чтобы при перезагрузке демон стартовал автоматически, выполните команду:

```
root@GW1:~# update-rc.d quagga enable
```

```
update-rc.d: using dependency based boot sequencing
```

В приложении представлен текст ripd.conf для шлюза GW1.

Настройка шлюза безопасности GW2

Настройка шлюза безопасности GW2 происходит аналогично настройке шлюза GW1, с заменой IP-адресов в соответствующих разделах конфигурации.

Обратите внимание на измененный адрес пула:

```
GW2(config)#ip local pool POOL 192.168.12.1 192.168.12.254
```

Это необходимо для того чтобы не было коллизий при выдаче IP-адресов.

Также изменится access-list в файле /etc/quagga/ripd.conf:

```
access-list acl-out permit 192.168.12.0/24
```

В приложении представлен текст cisco-like конфигурации, текст LSP конфигурации и текст ripd.conf для шлюза GW2.

Настройка клиента Client1

Настройка клиента состоит из следующих этапов:

- установка приложения AdminTool на компьютере администратора;
- получение запросов на сертификаты (на основе новых контейнеров с ключами) на компьютере администратора;
- перенос запросов на УЦ, получение сертификатов из запросов, перенос сертификатов на компьютер администратора;
- формирование установочного пакета для целевого клиентского компьютера с помощью AdminTool;
- перенос пакета и его установка на целевом клиентском компьютере.

Предполагается, что формирование установочного пакета будет происходить на компьютере администратора. Контейнеры с ключами будут генерироваться также на компьютере администратора. На базе сгенерированных ключей будут выпускаться запросы на сертификаты. Запросы на сертификаты будут переданы на УЦ и на их основе будут получены сертификаты.

В данном сценарии не описывается процесс установки AdminTool, а также процесс выпуска и переноса сертификатов.

Более подробное описание программы AdminTool представлено в документации.

Процедура выдачи сертификата по запросу описана в документации.

1. Создайте запрос на сертификат с помощью утилиты `excont_mgr`, входящей в состав AdminTool

   1.1. Запустите командую строку от имени Администратора.

   1.2. Перейдите в директорию AdminTool:

   ```
   cd "C:\Program Files (x86)\S-Terra Client AdminTool st"
   ```

   1.3. Создайте запрос на сертификат. При этом будет создан новый контейнер с ключами.

   ```
   excont_mgr.exe create_req -subj "C=RU,O=S-Terra CSP,OU=Research,CN=Client1" -GOST_R341012_256 -kc Client1 -kcp 1234 -fo C:\Client1.req
   ```

   - `create_req` – создать запрос на сертификат;
   - `-subj <DN>` – указать поля сертификата;
   - `-GOST_R341012_256` – формат запроса;
   - `-kc <имя контейнера>` – задать имя контейнера;
   - `-kcp <PIN>` – задать пароль на контейнер;
   - `-fo <путь до файла>` – указать путь, по которому будет сохранен запрос на сертификат.

   Ключ `-GOST_R341012_256` предполагает использование ГОСТ 2012. На УЦ для его поддержки должно быть установлено СКЗИ «КриптоПро CSP» версии 4.0 или новее. При необходимости, можно воспользоваться более старым ключом `-GOST_R3410EL`.

   Более полное описание утилиты `excont_mgr` представлено в документации.

2. Созданный запрос перенесите на УЦ и получите на его основе пользовательский сертификат. Также получите сертификат данного УЦ. Перенесите сертификат УЦ и пользовательский сертификат на компьютер администратора.

3. Создайте установочный пакет для AdminHost.

   4.1. Запустите установленное приложение AdminTool.

   4.2. Во вкладке **Auth** выполните следующие действия (Рисунок 2):

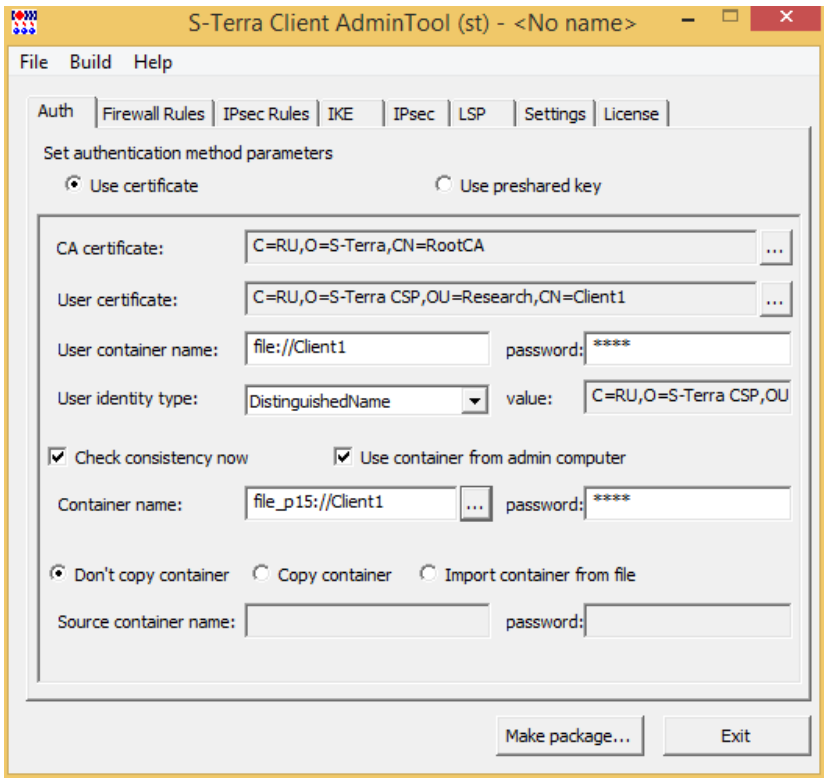

Рисунок 2

4.2.1 В данном сценарии используется метод аутентификации на сертификатах – переключатель установлен на **Use certificate** по умолчанию.

4.2.2 Укажите путь к сертификату УЦ и пользовательскому сертификату.

4.2.3 Отметьте флаг **Check consistency now** и нажмите кнопку **"..."**, где выберите созданный ранее контейнер. Если при создании запроса на сертификат указывался пароль на контейнер, введите его в поле **password**.

4.2.4 Отметьте флаг **Use container from admin computer**. Указанный в п.3.2.3 контейнер будет помещен в установочный пакет.

4.2.5 Задайте имя контейнера в поле **User container name.** В данном случае указано – `file://Client1`. Данное поле указывает по какому пути искать контейнер при работе. Так как отмечен флаг **User container name,** то контейнер будет скопирован по указанному пути.

4.2.6 В поле **User identity type** необходимо использовать **DistiguishedName** (выбрано по умолчанию).

4.3. Во вкладке **Firewall Rules** (Рисунок 3) можно настроить правила фильтрации трафика. В данном сценарии оставьте настройки по умолчанию - разрешать весь трафик.

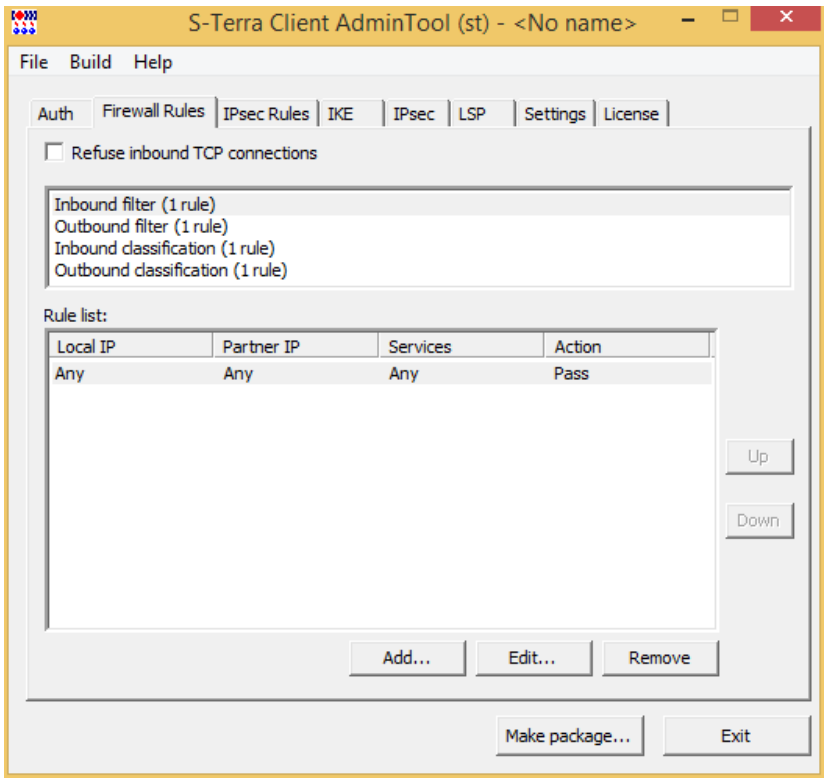

Рисунок 3

4.4. Вкладка **IPsec Rules**:

4.4.1 Добавьте правило для трафика, подлежащего шифрованию. Нажмите кнопку **Add** и в открывшемся окне **Add Rule** проведите следующие настройки (Рисунок 4):

4.4.1.1 В разделе **Action** выберите из списка **Protect using IPsec**, укажите адрес шлюза GW1 – 10.1.1.2 и шлюза GW2 – 10.1.1.3, а также отметьте флаг **Request IKECFG address**.

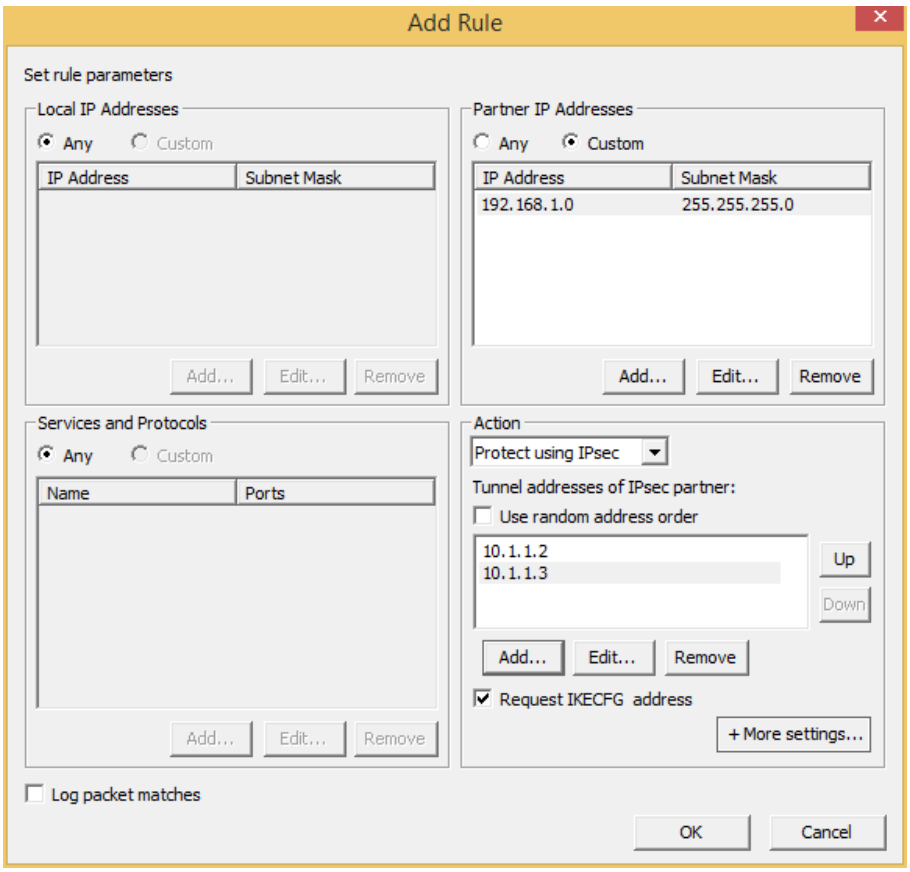

Рисунок 4

4.4.2 Добавленное правило поднимите вверх (Рисунок 5).

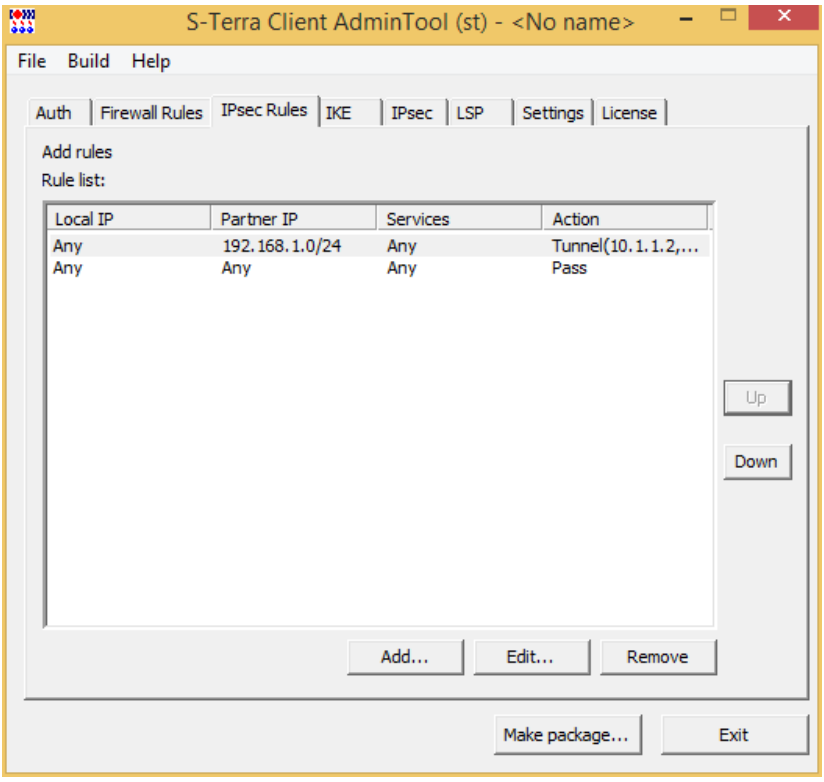

Рисунок 5

4.5. Во вкладке **IKE** по умолчанию установлены нужные настройки (Рисунок 6). При необходимости можно поднять в приоритете используемое правило.

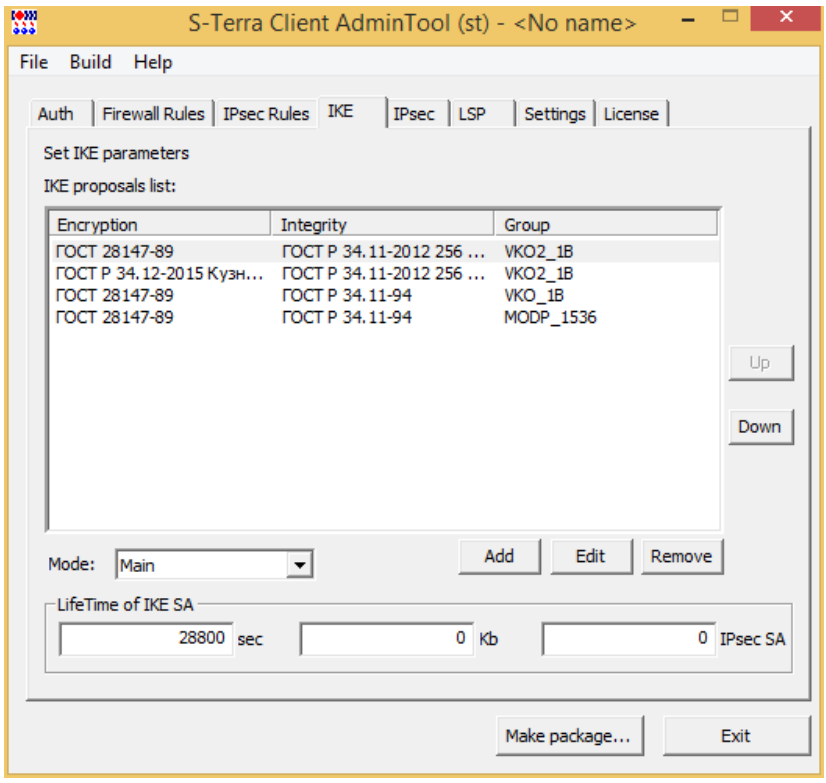

Рисунок 6

4.6. Во вкладке **IPsec** по умолчанию установлены нужные настройки (Рисунок 7). При необходимости можно поднять в приоритете используемое правило.

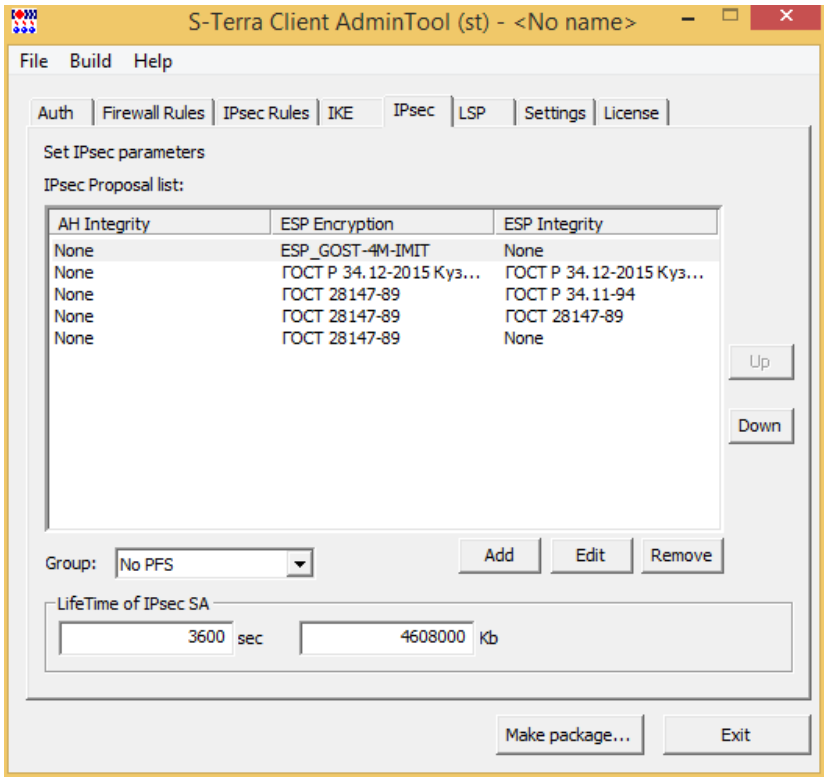

Рисунок 7

4.7. Во вкладке **LSP** можно просмотреть получившуюся политику безопасности.

4.8. Во вкладке **License** введите лицензию на продукт «С-Терра Клиент» версии

4.9. Сохраните файл созданного проекта, на тот случай, если захотите в будущем сделать похожий клиентский пакет. Для этого в меню **File** выберите **Save project**.

4.10. Создайте установочный exe-файл для «С-Терра Клиент», нажав кнопку **Make package...**.

5. Установка на целевой клиентский компьютер.

5.1. Установите на клиентском компьютере полученный exe-файл.

5.2. В области уведомлений появится иконка «С-Терра Клиент» (Рисунок 8). Для начала работы необходимо пройти процедуру аутентификации (Рисунок 9). Имя пользователя по умолчанию – `user`. Пароль по умолчанию отсутствует, в дальнейшем его можно установить.

Рисунок 8

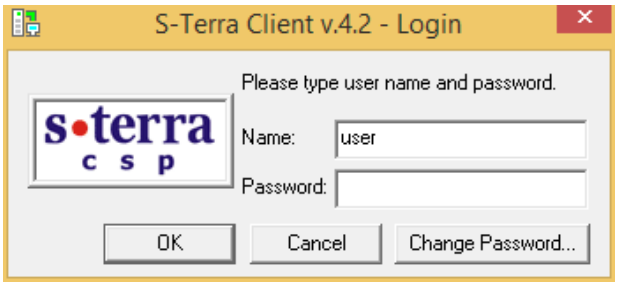

Рисунок 9

В приложении представлен текст LSP конфигурации для клиента Client1.

Настройка устройства Router1

На устройстве Router1 необходимо настроить соответствующие IP-адреса.

Настройте динамическую маршрутизацию по протоколу RIP (показан синтаксис маршрутизатора Cisco):

```
router rip
 version 2
 network 192.168.100.0
```

Настройка устройства ISP

На устройстве ISP необходимо настроить соответствующие IP-адреса.

Настройка устройства Host1

На устройстве Host1 задайте IP-адрес, а в качестве шлюза по умолчанию укажите адрес внутреннего интерфейса устройстваRouter1 – 192.168.1.1.

Проверка работоспособности стенда

После того, как настройка всех устройств завершена, инициируйте создание защищенного соединения.

На устройстве Client1 выполните команду ping:

```
C:\Users\Administrator> ping 192.168.1.100
```

```
Обмен пакетами с 192.168.1.100 по с 32 байтами данных:
Ответ от 192.168.1.100: число байт=32 время=1мс TTL=61
Ответ от 192.168.1.100: число байт=32 время<1мс TTL=61
Ответ от 192.168.1.100: число байт=32 время<1мс TTL=61
Ответ от 192.168.1.100: число байт=32 время<1мс TTL=61

Статистика Ping для 192.168.1.100:
    Пакетов: отправлено = 4, получено = 4, потеряно = 0
    (0% потерь)
Приблизительное время приема-передачи в мс:
    Минимальное = 0мсек, Максимальное = 1 мсек, Среднее = 0 мсек
```

В результате выполнения этой команды между устройствами Client1 и GW1 будет установлен VPN туннель.

Убедиться в этом можно, выполнив на устройстве GW1 команду:

```
root@GW1:~#  sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 1 (10.1.1.2,4500)-(10.1.1.1,4500) active 1808 3068

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 1 (192.168.1.0-192.168.1.255,*)-(192.168.11.1,*) * ESP nat-t3-tunn 256 256
```

На устройстве Router1 можно посмотреть добавленный маршрут:

```
Router1# show ip route
```

```
Codes: K - kernel route, C - connected, S - static, R - RIP,
       O - OSPF, I - IS-IS, B - BGP, P - PIM, A - Babel,
       > - selected route, * - FIB route

C>* 10.0.0.0/16 is directly connected, eth4
C>* 127.0.0.0/8 is directly connected, lo
C>* 192.168.1.0/24 is directly connected, eth0
C>* 192.168.100.0/24 is directly connected, eth1
```

Пустите с устройства Client1 бесконечный ping и сделайте обрыв связи:

```
Обмен пакетами с 192.168.1.100 по с 32 байтами данных:
Ответ от 192.168.1.100: число байт=32 время=1мс TTL=61
Ответ от 192.168.1.100: число байт=32 время<1мс TTL=61
Ответ от 192.168.1.100: число байт=32 время<1мс TTL=61
Ответ от 192.168.1.100: число байт=32 время<1мс TTL=61
Ответ от 192.168.1.100: число байт=32 время<1мс TTL=61
Ответ от 192.168.1.100: число байт=32 время<1мс TTL=61
Превышен интервал ожидания для запроса.
…
Превышен интервал ожидания для запроса.
Ответ от 192.168.1.100: число байт=32 время<1мс TTL=61
Ответ от 192.168.1.100: число байт=32 время<1мс TTL=61
```

Туннель перестроился на второй шлюз:

```
root@GW2:~# sa_mgr show
```

```
ISAKMP sessions: 0 initiated, 0 responded

ISAKMP connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) State Sent Rcvd
1 1 (10.1.1.3,4500)-(10.1.1.1,4500) active 1808 3068

IPsec connections:
Num Conn-id (Local Addr,Port)-(Remote Addr,Port) Protocol Action Type Sent Rcvd
1 1 (192.168.1.0-192.168.1.255,*)-(192.168.12.1,*) * ESP nat-t3-tunn 832 832
```

Выявление ошибок

Чтобы разобраться на каком этапе возникла ошибка можно воспользоваться руководством, которое представлено на портале документации: http://doc.s-terra.ru/rh_output/4.2/Scenarios/output/mergedProjects/1main/ver_4_2_troubleshooting_guide.pdf.

Приложение

Текст cisco-like конфигурации для шлюза GW1

```
!
version 12.4
no service password-encryption
!
crypto ipsec df-bit copy
crypto isakmp identity dn
crypto isakmp keepalive 10
crypto isakmp keepalive retry-count 5
username cscons privilege 15 password 0 csp
aaa new-model
!
!
hostname GW1
enable password csp
!
!
!
!
!
crypto isakmp policy 1
 encr gost
 hash gost341112-256-tc26
 authentication gost-sig
 group vko2
!
ip local pool POOL 192.168.11.1 192.168.11.254
!
crypto ipsec transform-set TSET esp-gost28147-4m-imit
!
ip access-list extended LIST
 permit ip 192.168.1.0 0.0.0.255 192.168.11.0 0.0.0.255
!
!
crypto dynamic-map DMAP 1
 match address LIST
 set transform-set TSET
 set pool POOL
 reverse-route
!
crypto map CMAP 1 ipsec-isakmp dynamic DMAP
!
interface GigabitEthernet0/0
 ip address 192.168.100.2 255.255.255.0
!
interface GigabitEthernet0/1
 ip address 10.1.1.2 255.255.255.0
 crypto map CMAP
!
interface GigabitEthernet0/2
 no ip address
 shutdown
!
interface GigabitEthernet0/3
 no ip address
 shutdown
!
!
ip route 0.0.0.0 0.0.0.0 10.1.1.1
ip route 192.168.1.0 255.255.255.0 192.168.100.1
!
crypto pki trustpoint s-terra_technological_trustpoint
```

```
 revocation-check crl
 crl download group GROUP http://10.0.221.179/certsrv/certcrl.crl
crypto pki certificate chain s-terra_technological_trustpoint
certificate 1F76A0063C0401B94DA17E11D2B4AC8A
...
6278713A76B6046B582F722FDB3854A1F603212EC0FA537A1D2E36C453EFC6

quit
!
end
```

Текст cisco-like конфигурации для шлюза GW2

```
!
version 12.4
no service password-encryption
!
crypto ipsec df-bit copy
crypto isakmp identity dn
crypto isakmp keepalive 10
crypto isakmp keepalive retry-count 5
username cscons privilege 15 password 0 csp
aaa new-model
!
!
hostname GW2
enable password csp
!
!
!
!
!
crypto isakmp policy 1
 encr gost
 hash gost341112-256-tc26
 authentication gost-sig
 group vko2
!
ip local pool POOL 192.168.12.1 192.168.12.254
!
crypto ipsec transform-set TSET esp-gost28147-4m-imit
!
ip access-list extended LIST
 permit ip 192.168.1.0 0.0.0.255 192.168.12.0 0.0.0.255
!
!
crypto dynamic-map DMAP 1
 match address LIST
 set transform-set TSET
 set pool POOL
 reverse-route
!
crypto map CMAP 1 ipsec-isakmp dynamic DMAP
!
interface GigabitEthernet0/0
 ip address 192.168.100.3 255.255.255.0
!
interface GigabitEthernet0/1
 ip address 10.1.1.3 255.255.255.0
 crypto map CMAP
!
interface GigabitEthernet0/2
 no ip address
 shutdown
!
interface GigabitEthernet0/3
```

```
 no ip address
 shutdown
!
!
ip route 0.0.0.0 0.0.0.0 10.1.1.1
ip route 192.168.1.0 255.255.255.0 192.168.100.1
!
crypto pki trustpoint s-terra_technological_trustpoint
 revocation-check crl
 crl download group GROUP http://10.0.221.179/certsrv/certcrl.crl
crypto pki certificate chain s-terra_technological_trustpoint
certificate 1F76A0063C0401B94DA17E11D2B4AC8A
...
6278713A76B6046B582F722FDB3854A1F603212EC0FA537A1D2E36C453EFC6

quit
!
end
```

Текст LSP конфигурации для шлюза GW1

```
#   This is automatically generated LSP
#
#   Conversion Date/Time:   Tue Feb 26 15:18:35 2019

GlobalParameters(
    Title                       = "This LSP was automatically generated by CSP Converter
at Tue Feb 26 15:18:35 2019"
    Version                       = LSP_4_2
    CRLHandlingMode               = ENABLE
    PreserveIPsecSA               = FALSE
)

IKEParameters(
    FragmentSize = 0
)

RoutingTable(
    Routes =
        Route(
            Destination = 0.0.0.0/0
            Gateway = 10.1.1.1
        ),
        Route(
            Destination = 192.168.1.0/24
            Gateway = 192.168.100.1
        )
)

FirewallParameters(
    TCPSynSentTimeout = 30
    TCPFinTimeout = 5
    TCPClosedTimeout = 30
    TCPSynRcvdTimeout = 30
    TCPEstablishedTimeout = 3600
    TCPHalfOpenLow = 400
    TCPHalfOpenMax = 500
    TCPSessionRateLow = 400
    TCPSessionRateMax = 500
)

IKETransform crypto:isakmp:policy:1
(
    CipherAlg   = "G2814789CPRO1-K256-CBC-65534"
    HashAlg     = "GR341112_256TC26-65128"
    GroupID     = VKO2_1B
```

```
    RestrictAuthenticationTo = GOST_SIGN
    LifetimeSeconds = 86400
)

ESPProposal TSET:ESP
(
    Transform* = ESPTransform
    (
        CipherAlg*          = "G2814789CPRO2-K288-CNTMAC-253"
        LifetimeSeconds     = 3600
        LifetimeKilobytes   = 4608000
    )
)

AddressPool POOL
(
    IPAddresses = 192.168.11.1..192.168.11.254
)

AuthMethodGOSTSign GOST:Sign
(
    LocalID        =  IdentityEntry( DistinguishedName* = USER_SPECIFIC_DATA )
    SendRequestMode    =  ALWAYS
    SendCertMode       =  ALWAYS
)

IKERule IKERule:CMAP:1:DMAP:1
(
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = GOST:Sign
    MainModeAuthMethod  = GOST:Sign
    IKECFGPool          = POOL
    DPDIdleDuration     = 10
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 100
)

IPsecAction IPsecAction:CMAP:1:DMAP:1
(
    TunnelingParameters = TunnelEntry(
        DFHandling=COPY
        Assemble=TRUE
    )
    ContainedProposals = ( TSET:ESP )
    ReverseRoute = TRUE
    IKERule = IKERule:CMAP:1:DMAP:1
)

FilterChain IPsecPolicy:CMAP (
    Filters = Filter (
        ProtocolID = 17
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
    Filter (
        SourceIP = 192.168.1.0/24
        DestinationIP = 192.168.11.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:CMAP:1:DMAP:1 >
        LogEventID = "IPsec:Protect:CMAP:1:DMAP:1:LIST"
    )
)

NetworkInterface (
```

```
    LogicalName = "GigabitEthernet0/1"
    IPsecPolicy = IPsecPolicy:CMAP
)
```

Текст LSP конфигурации для шлюза GW2

```
#   This is automatically generated LSP
#
#   Conversion Date/Time:   Tue Feb 26 15:19:41 2019

GlobalParameters(
    Title                     = "This LSP was automatically generated by CSP Converter
at Tue Feb 26 15:19:41 2019"
    Version                     = LSP_4_2
    CRLHandlingMode             = ENABLE
    PreserveIPsecSA             = FALSE
)

IKEParameters(
    FragmentSize = 0
)

RoutingTable(
    Routes =
        Route(
            Destination = 0.0.0.0/0
            Gateway = 10.1.1.1
        ),
        Route(
            Destination = 192.168.1.0/24
            Gateway = 192.168.100.1
        )
)

FirewallParameters(
    TCPSynSentTimeout = 30
    TCPFinTimeout = 5
    TCPClosedTimeout = 30
    TCPSynRcvdTimeout = 30
    TCPEstablishedTimeout = 3600
    TCPHalfOpenLow = 400
    TCPHalfOpenMax = 500
    TCPSessionRateLow = 400
    TCPSessionRateMax = 500
)

IKETransform crypto:isakmp:policy:1
(
    CipherAlg   = "G2814789CPRO1-K256-CBC-65534"
    HashAlg     = "GR341112_256TC26-65128"
    GroupID     = VKO2_1B
    RestrictAuthenticationTo = GOST_SIGN
    LifetimeSeconds = 86400
)

ESPProposal TSET:ESP
(
    Transform* = ESPTransform
    (
        CipherAlg*          = "G2814789CPRO2-K288-CNTMAC-253"
        LifetimeSeconds     = 3600
        LifetimeKilobytes   = 4608000
    )
)
```

```
AddressPool POOL
(
    IPAddresses = 192.168.12.1..192.168.12.254
)

AuthMethodGOSTSign GOST:Sign
(
    LocalID        =  IdentityEntry( DistinguishedName* = USER_SPECIFIC_DATA )
    SendRequestMode    =  ALWAYS
    SendCertMode       =  ALWAYS
)

IKERule IKERule:CMAP:1:DMAP:1
(
    Transform = crypto:isakmp:policy:1
    AggrModeAuthMethod  = GOST:Sign
    MainModeAuthMethod  = GOST:Sign
    IKECFGPool          = POOL
    DPDIdleDuration     = 10
    DPDResponseDuration = 2
    DPDRetries          = 5
    Priority            = 100
)

IPsecAction IPsecAction:CMAP:1:DMAP:1
(
    TunnelingParameters = TunnelEntry(
        DFHandling=COPY
        Assemble=TRUE
    )
    ContainedProposals = ( TSET:ESP )
    ReverseRoute = TRUE
    IKERule = IKERule:CMAP:1:DMAP:1
)

FilterChain IPsecPolicy:CMAP (
    Filters = Filter (
        ProtocolID = 17
        SourcePort = 500, 4500
        Action = PASS
        PacketType = LOCAL_UNICAST, LOCAL_MISDIRECTED
    ),
    Filter (
        SourceIP = 192.168.1.0/24
        DestinationIP = 192.168.12.0/24
        Action = PASS
        ExtendedAction = ipsec< sa = IPsecAction:CMAP:1:DMAP:1 >
        LogEventID = "IPsec:Protect:CMAP:1:DMAP:1:LIST"
    )
)

NetworkInterface (
    LogicalName = "GigabitEthernet0/1"
    IPsecPolicy = IPsecPolicy:CMAP
)
```

Текст LSP конфигурации для клиента Client1

```
GlobalParameters (
      Title = "This LSP was automatically generated by S-Terra Client AdminTool (st)
at 2019.02.26 14:07:24"
      Version = LSP_4_2
      CRLHandlingMode = BEST_EFFORT
)
LDAPSettings (
```

```
	ResponseTimeout = 200
	HoldConnectTimeout = 60
	DropConnectTimeout = 5
)
IdentityEntry auth_identity_01(
	DistinguishedName *= CertDescription(
		Subject *= COMPLETE,"C=RU,O=S-Terra CSP,OU=Research,CN=Client1"
		)
)
CertDescription local_cert_dsc_01(
	Subject *= COMPLETE,"C=RU,O=S-Terra CSP,OU=Research,CN=Client1"
	Issuer *= COMPLETE,"C=RU,O=S-Terra,CN=RootCA"
	SerialNumber = "180000000F07B54F512CB2384C00000000000F"
	FingerprintMD5 = "FFFD5C827C0CD7A99E58FD0A142C1738"
)
CertDescription partner_cert_dsc_01(
)
AuthMethodGOSTSign auth_method_01(
	LocalID = auth_identity_01
	LocalCredential = local_cert_dsc_01
	RemoteCredential = partner_cert_dsc_01
	SendRequestMode = AUTO
	SendCertMode = AUTO
)
IKEParameters (
	DefaultPort = 500
	SendRetries = 5
	RetryTimeBase = 1
	RetryTimeMax = 30
	SessionTimeMax = 60
	InitiatorSessionsMax = 30
	ResponderSessionsMax = 20
	BlacklogSessionsMax = 16
	BlacklogSessionsMin = 0
	BlacklogSilentSessions = 4
	BlacklogRelaxTime = 120
	IKECFGPreferDefaultAddress = FALSE
)
IKETransform ike_trf_01(
	LifetimeSeconds = 28800
	CipherAlg  *= "G2814789CPRO1-K256-CBC-65534"
	HashAlg *= "GR341112_256TC26-65128"
	GroupID *= VKO2_1B
)
IKETransform ike_trf_02(
	LifetimeSeconds = 28800
	CipherAlg  *= "GR341215K-K256-CFB-65528"
	HashAlg *= "GR341112_256TC26-65128"
	GroupID *= VKO2_1B
)
IKETransform ike_trf_03(
	LifetimeSeconds = 28800
	CipherAlg  *= "G2814789CPRO1-K256-CBC-65534"
	HashAlg *= "GR341194CPRO1-65534"
	GroupID *= VKO_1B
)
IKETransform ike_trf_04(
	LifetimeSeconds = 28800
	CipherAlg  *= "G2814789CPRO1-K256-CBC-65534"
	HashAlg *= "GR341194CPRO1-65534"
	GroupID *= MODP_1536
)
ESPTransform esp_trf_01(
	CipherAlg *= "G2814789CPRO2-K288-CNTMAC-253"
	LifetimeSeconds = 3600
	LifetimeKilobytes = 4608000
```

```
)
ESPProposal esp_proposal_01(
	Transform *=esp_trf_01
)
ESPTransform esp_trf_02(
	CipherAlg *= "GR341215K-K256-CFB-248"
	IntegrityAlg *= "GR341215K-K256-MAC-65529"
	LifetimeSeconds = 3600
	LifetimeKilobytes = 4608000
)
ESPProposal esp_proposal_02(
	Transform *=esp_trf_02
)
ESPTransform esp_trf_03(
	CipherAlg *= "G2814789CPRO1-K256-CBC-254"
	IntegrityAlg *= "GR341194CPRO1-H96-HMAC-65534"
	LifetimeSeconds = 3600
	LifetimeKilobytes = 4608000
)
ESPProposal esp_proposal_03(
	Transform *=esp_trf_03
)
ESPTransform esp_trf_04(
	CipherAlg *= "G2814789CPRO1-K256-CBC-254"
	IntegrityAlg *= "G2814789CPRO1-K256-MAC-65535"
	LifetimeSeconds = 3600
	LifetimeKilobytes = 4608000
)
ESPProposal esp_proposal_04(
	Transform *=esp_trf_04
)
ESPTransform esp_trf_05(
	CipherAlg *= "G2814789CPRO1-K256-CBC-254"
	LifetimeSeconds = 3600
	LifetimeKilobytes = 4608000
)
ESPProposal esp_proposal_05(
	Transform *=esp_trf_05
)
IKERule ike_rule_with_ikecfg_01(
	DoNotUseDPD = FALSE
	DPDIdleDuration = 60
	DPDResponseDuration = 5
	DPDRetries = 3
	MainModeAuthMethod *= auth_method_01
	Transform *= ike_trf_01,ike_trf_02,ike_trf_03,ike_trf_04
	IKECFGRequestAddress = TRUE
)
IPsecAction ipsec_action_01(
	PersistentConnection = TRUE
	TunnelingParameters *=
		TunnelEntry(
			PeerAddress = 10.1.1.2
			Assemble = TRUE
			ReRoute = FALSE
			TCPEncapsulation = FALSE
		),
		TunnelEntry(
			PeerAddress = 10.1.1.3
			Assemble = TRUE
			ReRoute = FALSE
			TCPEncapsulation = FALSE
		)
	ShuffleTunnelEntries = FALSE
```

```
	ContainedProposals *= (esp_proposal_01),(esp_proposal_02),(esp_proposal_03),(esp_proposal_04),(esp_proposal_05)
	IKERule = ike_rule_with_ikecfg_01
)
FilterChain filter_chain_input(
	Filters *= Filter(
		ProtocolID *= 17
		DestinationPort *= 500
		Action = PASS
		LogEventID = "pass_action_02_01"
		PacketType = LOCAL_UNICAST,LOCAL_MISDIRECTED
	),Filter(
		ProtocolID *= 17
		DestinationPort *= 4500
		Action = PASS
		LogEventID = "pass_action_02_02"
		PacketType = LOCAL_UNICAST,LOCAL_MISDIRECTED
	),Filter(
		SourceIP *= 10.1.1.2,10.1.1.3
		ProtocolID *= 50
		Action = PASS
		LogEventID = "pass_action_03_01"
		PacketType = LOCAL_UNICAST,LOCAL_MISDIRECTED
	),Filter(
		SourceIP *= 10.1.1.2,10.1.1.3
		ProtocolID *= 51
		Action = PASS
		LogEventID = "pass_action_03_02"
		PacketType = LOCAL_UNICAST,LOCAL_MISDIRECTED
	),Filter(
		Action = PASS
		LogEventID = "pass_action_04"
	)
)
FilterChain filter_chain_output(
	Filters *= Filter(
		ProtocolID *= 17
		SourcePort *= 500
		Action = PASS
		LogEventID = "pass_action_05_01"
		PacketType = LOCAL_UNICAST,LOCAL_MISDIRECTED
	),Filter(
		ProtocolID *= 17
		SourcePort *= 4500
		Action = PASS
		LogEventID = "pass_action_05_02"
		PacketType = LOCAL_UNICAST,LOCAL_MISDIRECTED
	),Filter(
		DestinationIP *= 10.1.1.2,10.1.1.3
		ProtocolID *= 50
		Action = PASS
		LogEventID = "pass_action_06_01"
		PacketType = LOCAL_UNICAST,LOCAL_MISDIRECTED
	),Filter(
		DestinationIP *= 10.1.1.2,10.1.1.3
		ProtocolID *= 51
		Action = PASS
		LogEventID = "pass_action_06_02"
		PacketType = LOCAL_UNICAST,LOCAL_MISDIRECTED
	),Filter(
		Action = PASS
		LogEventID = "pass_action_07"
	)
)
FilterChain filter_chain_classification_input(
```

```
	Filters *= Filter(
		Action = PASS
		LogEventID = "pass_action_08"
	)
)
FilterChain filter_chain_classification_output(
	Filters *= Filter(
		Action = PASS
		LogEventID = "pass_action_09"
	)
)
FilterChain filter_chain_ipsec(
	Filters *= Filter(
		ProtocolID *= 17
		SourcePort *= 500
		Action = PASS
		LogEventID = "pass_action_10_01"
		PacketType = LOCAL_UNICAST,LOCAL_MISDIRECTED
	),Filter(
		ProtocolID *= 17
		SourcePort *= 4500
		Action = PASS
		LogEventID = "pass_action_10_02"
		PacketType = LOCAL_UNICAST,LOCAL_MISDIRECTED
	),Filter(
		DestinationIP *= 10.1.1.2,10.1.1.3
		ProtocolID *= 50
		Action = PASS
		LogEventID = "pass_action_11_01"
		PacketType = LOCAL_UNICAST,LOCAL_MISDIRECTED
	),Filter(
		DestinationIP *= 10.1.1.2,10.1.1.3
		ProtocolID *= 51
		Action = PASS
		LogEventID = "pass_action_11_02"
		PacketType = LOCAL_UNICAST,LOCAL_MISDIRECTED
	),Filter(
		DestinationIP *= 192.168.1.0/24
		Action = PASS
		ExtendedAction *= ipsec<sa=ipsec_action_01>
		LogEventID = "ipsec_action_01"
	),Filter(
		Action = PASS
		LogEventID = "pass_action_12"
	)
)
NetworkInterface(
	InputFilter = filter_chain_input
	OutputFilter = filter_chain_output
	InputClassification = filter_chain_classification_input
	OutputClassification = filter_chain_classification_output
	IPsecPolicy = filter_chain_ipsec
)
```

Текст конфигурации ripd.conf для шлюза GW1

```
hostname GW1
password csp
log file /var/log/quagga/rip.log debugging
!
router rip
version 2
redistribute kernel
network eth0
distribute-list acl-in in
distribute-list acl-out out
```

```
!
access-list acl-in deny any
access-list acl-out permit 192.168.11.0/24
access-list acl-out deny any
line vty
```

Текст конфигурации ripd.conf для шлюза GW2

```
hostname GW2
password csp
log file /var/log/quagga/rip.log debugging
!
router rip
version 2
redistribute kernel
network eth0
distribute-list acl-in in
distribute-list acl-out out
!
access-list acl-in deny any
access-list acl-out permit 192.168.12.0/24
access-list acl-out deny any
line vty
```