Обзорная статья на тему использования современных приватных тоннелей в роутерах популярной латвийской марки. Я расскажу о том, как настроить vpn сервер в mikrotik на базе таких технологий как l2tp, ipsec, openvpn, pptp, gre и eoip. Попутно кратко расскажу о том, что это за технологии, чем они отличаются, а так же проведу сравнение производительности микротика со всеми указанными тоннелями.

> Если у вас есть желание научиться работать с роутерами микротик и стать специалистом в этой области, рекомендую по программе, основанной на информации из официального курса **MikroTik Certified Network Associate**. Курс стоящий, все подробности читайте по ссылке.

Содержание:

- 1 Введение
- 2 Варианты vpn сервера в микротике
- 3 Настройка l2tp туннеля в mikrotik
  - 3.1 L2tp клиент
  - 3.2 Настраиваем ipsec
- 4 Настройка pptp сервера в mikrotik
  - 4.1 pptp клиент
- 5 Настройка openvpn server в микротик
  - 5.1 openvpn client
- 6 Настройка EOIP Tunnel + Ipsec
- 7 GRE туннель + Ipsec в mikrotik, создание и настройка
- 8 Сравнение скорости L2tp, Pptp, EoIP, GRE и OpenVPN туннелей
- 9 Заключение

Введение

Сразу хочу обратить внимание, что эта статья будет скорее обзорной, нежели передачей реального опыта, так как сам я чаще всего использую в качестве vpn сервера openvpn. Тем не менее с vpn в микротик тоже приходилось сталкиваться. Настраивал как pptp сервера для подключения удаленных клиентов, так и l2tp для объединения двух и более микротиков в общую приватную сеть. В основном по дефолту, не вникая в тонкости настроек.

Сегодня хочу рассмотреть этот вопрос более внимательно и посмотреть, что вообще предлагает микротик из коробки для настройки vpn соединений. Своими исследованиями я и хочу поделиться с вами, написав небольшой обзор на тему средств организации vpn сервера в mikrotik. А попутно хочу собрать отзывов и исправлений на тему написанного, чтобы укрепить свои знания. В комментариях к своим статьям я черпаю массу советов, за что благодарен всем писавшим полезные вещи. Так что замечания, дополнения и исправления категорически приветствуются.

Для тех, кто хочет хорошо разбираться в сетях, но пока по какой-то причине не умеет этого, рекомендую вот этот цикл статей — сети для самых маленьких.

Варианты vpn сервера в микротике

С вариантами vpn сервера в микротике все сложно :) В том плане, что есть много реализаций vpn, которую не так просто выбрать, если не разбираешься детально в сетевых технологиях. Я не сильно в них разбираюсь, но как мне кажется, немного улавливаю суть. Постараюсь вам объяснить своими словами, в чем отличия.

Существуют 2 принципиально разных решения для организации соединений между двумя микротиками и внешними абонентами:

1. Создание l2 туннеля типа site-to-site с помощью **EOIP Tunnel**. Самый простой и быстрый способ объединить два микротика. Если не будет использовано шифрование, то получатся самые быстрые vpn подключения. Необходимы выделенные белые ip адреса на обоих устройствах. Такие соединения используют для объединения офисов или филиалов по vpn. В общем случае не работает через NAT. Сюда так же добавлю **GRE Tunnel**, хотя он работает в l3 и использует маршрутизацию, но работает так же по принципу site-to-site.
2. VPN соединения уровня l3 на технологии Клиент-Сервер, типа **PPTP**, **L2TP**, **SSTP, OpenVPN**. Такие соединения используются как для объединения офисов, так и для подключения удаленных сотрудников. Достаточно только одного белого ip адреса на стороне сервера для создания vpn соединений. Работает через NAT.

Расскажу немного подробнее о каждом из типов vpn соединений отдельно.

- GRE Tunnel — использует простой протокол gre для построения базового незащищенного site-to-site VPN. Разработан компанией CISCO. Позволяет инкапсулировать пакеты различного типа внутри ip туннелей. Простыми словами вот что он делает. Берет ваши данные со всеми заголовками, упаковывает в пакет, передает по интернету на другой конец, где этот пакет обратно разбирается на исходные данные. Для конечных пользователей сети все это выглядит, как-будто они общаются через локальную сеть.
- EOIP Tunnel — Ethernet over IP — это проприетарный протокол MikroTik RouterOS, который создает туннель Ethernet между двумя маршрутизаторами поверх IP-соединения. Для передачи данных использует GRE протокол. Принципиальное отличие eoip tunnnel в том, что он работает в l2 и передает напрямую фреймы, тогда как gre tunnel оперирует пакетами и использует маршрутизацию. Надеюсь правильно объяснил и не соврал. Для чего mikrotik решили создать свою реализацию туннеля через gre протокол, не знаю. Возможно, похожих решений просто нет, вот они и придумали свою реализацию.
- PPTP — туннельный протокол типа точка-точка (Point-to-Point Tunneling Protocol). Для работы использует GRE протокол, поддерживает шифрование. В свое время pptp обрел большую популярность из-за того, что его из коробки поддерживала Windows начиная с версии 95. На сегодняшний день pptp использовать не рекомендуется, так как он очень легко взламывается. Из дампа трафика за короткое время (несколько часов) достается ключ шифрования и расшифровывается весь трафик. Возможно, с этим как-то можно бороться, используя разные протоколы шифрования, но я не разбирался подробно с этой темой. Для себя решил, что pptp можно использовать как самое простое решение там, где нет повышенных требований к безопасности и расшифровка трафика, если таковая и случится, не принесет никаких проблем. PPTP поддерживает из коробки не только Windows но и Android, что очень удобно. Настраивается очень просто.
- L2TP — Layer 2 Tunneling Protocol. Несмотря на то, что в названии указано l2, реально в ip сети он работает на сеансовом уровне, то есть l3. Использует в работе udp порт 1701. Может работать не только в IP сетях. Из коробки, как и pptp, поддерживает аутентификацию пользователей. Сам по себе не обеспечивает шифрование. Для шифрования трафика может использовать ipsec, который считается очень безопасным и не имеет серьезных уязвимостей. В настоящее время поддерживается практически всеми устройствами и системами из коробки, как и pptp. Настраивать не сильно сложнее. В общем случае, для организации vpn рекомендую использовать именно этот тип шифрованного тоннеля.
- OpenVPN — это очень популярная реализация шифрованных соединений. Главное достоинство — гибкость настроек. К примеру, очень крутая возможность openvnp — пушить маршруты напрямую клиенту при подключении. Я долгое время использовал openvpn серверы. Когда первый раз понадобилось передать клиенту pptp маршрут, никак не мог понять, как это настроить. Оказалось, что никак, он это просто не умеет. Пришлось настраивать сторонними инструментами. К сожалению, по непонятным причинам, в mikrotik openvpn не поддерживает протокол udp, что очень сужает возможности использования этого vpn сервера. По tcp он работает гораздо медленнее, чем по udp. Так же не работает сжатие заголовков пакетов. Так что в общем случае использовать openvpn сервер в микротик не имеет смысла, если только он не нужен вам по каким-то конкретным причинам.
- SSTP — Протокол безопасного туннелирования сокетов (Secure Socket Tunneling Protocol) – был представлен Microsoft в Windows Vista SP1. Основной плюс в том, что он интегрирован в Windows, может использовать 443 порт, что иногда помогает обходить фаерволы. Считается очень безопасным, использует SSL 3.0. Из минусов, насколько я знаю, в микротике очень требователен к ресурсам процессора. На слабеньких железках будет выдавать самую низкую скорость по сравнению со всеми остальными соединениями по vpn. По этой причине я его не буду рассматривать в своем обзоре совсем.

Из всего написанного можно сделать такой вывод. В общем случае лучше всего в микротике использовать vpn на базе l2tp + ipsec. Основные причины:

1. Простота и удобство настройки.
2. Надежное шифрование.
3. Поддержка l2tp соединений практически всеми современными устройствами и системами. Нет необходимости ставить дополнительное программное обеспечение.
4. Подходит как для объединения офисов, так и для удаленных сотрудников — site-to-site и client-to-site подключения.

Если вам нужно максимальное быстродействие без шифрования, то стройте соединения между сетями или офисами с помощью EOIP Tunnel — фирменной разработки компании Mikrotik.

Дальше я покажу, как настроить все описанные туннели, кроме SSTP и произведу замеры скорости для сравнения. Мой тестовый стенд из двух Mikrotik RB951G-2hnD будет иметь следующие настройки.

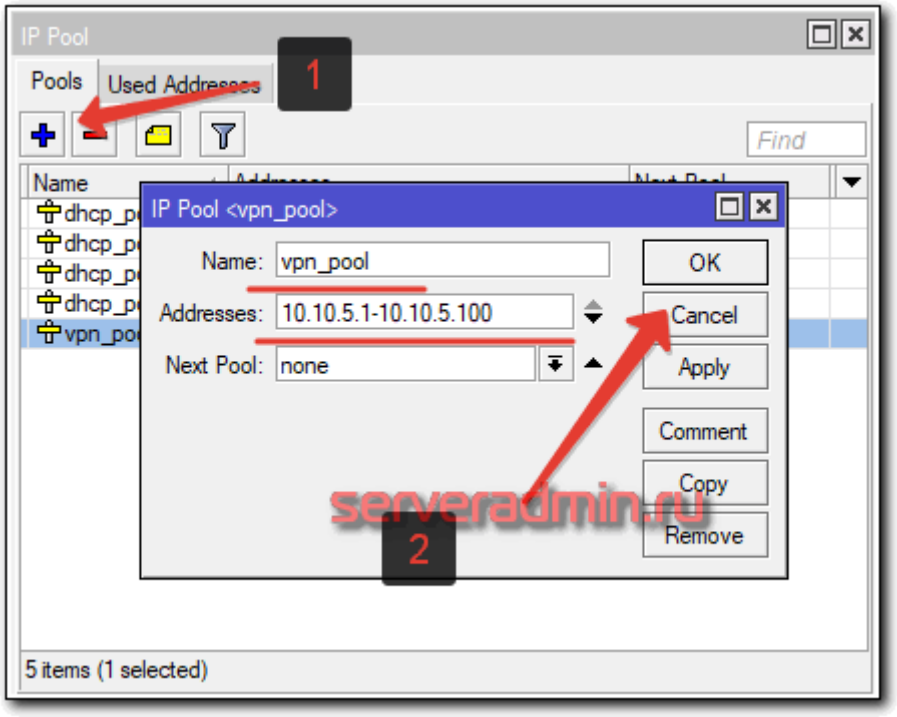
m-server	имя микротика, выступающего в роли сервера
m-remote	имя микротика, выступающего в роли удаленного маршрутизатора
192.168.13.1	WAN ip адрес на m-server
192.168.13.197	WAN ip адрес на m-remote
10.20.1.0/24	локальная сеть за m-server
10.30.1.0/24	локальная сеть за m-remote
10.10.5.1-10.10.5.100	vpn сеть

Приступим к настройке и тестированию vpn соединений в mikrotik.

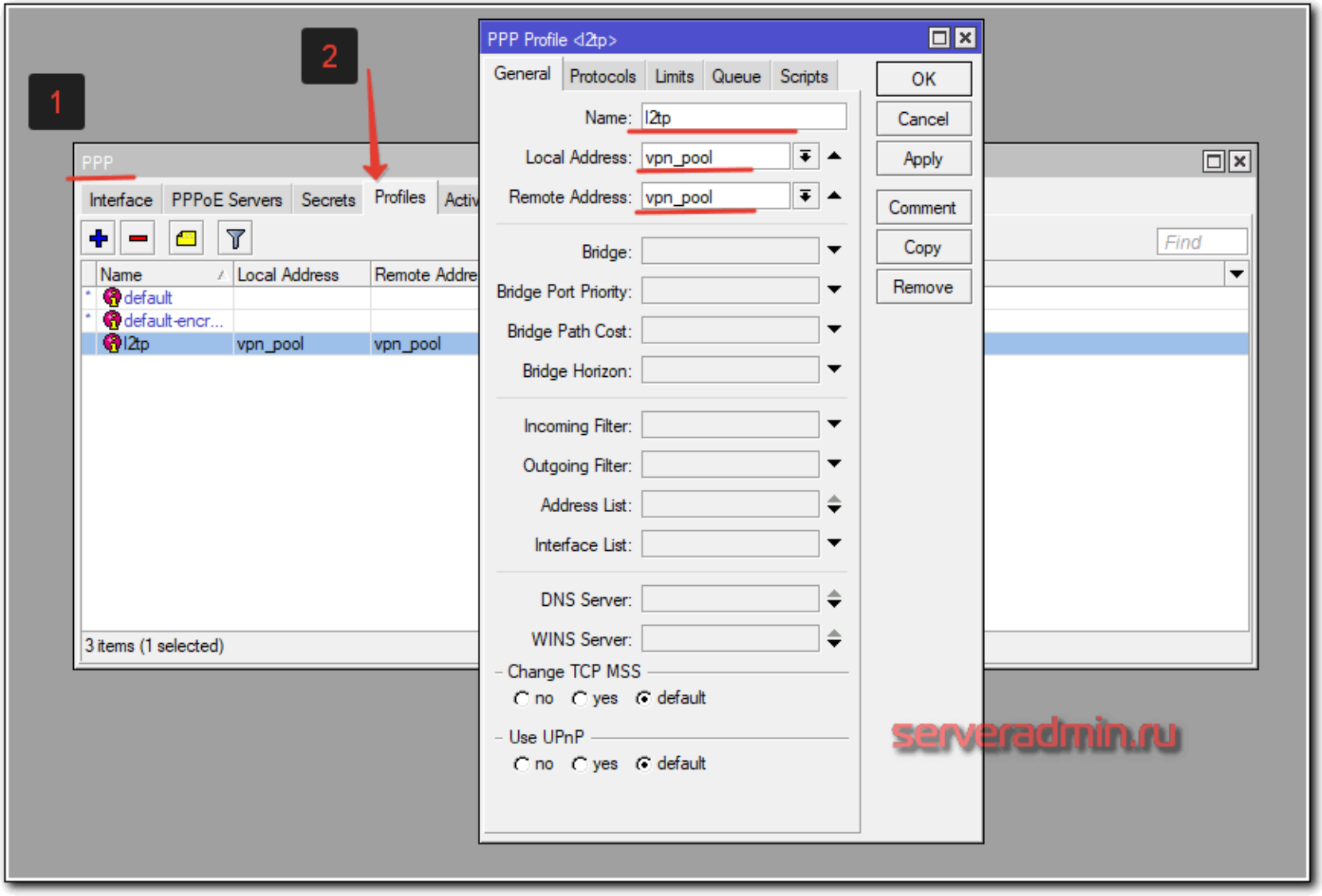
Настройка l2tp туннеля в mikrotik

Для начала настроим простой l2tp туннель без шифрования и замерим скорость. Для настройки l2tp vpn в mikrotik выполните следующую последовательность действий.

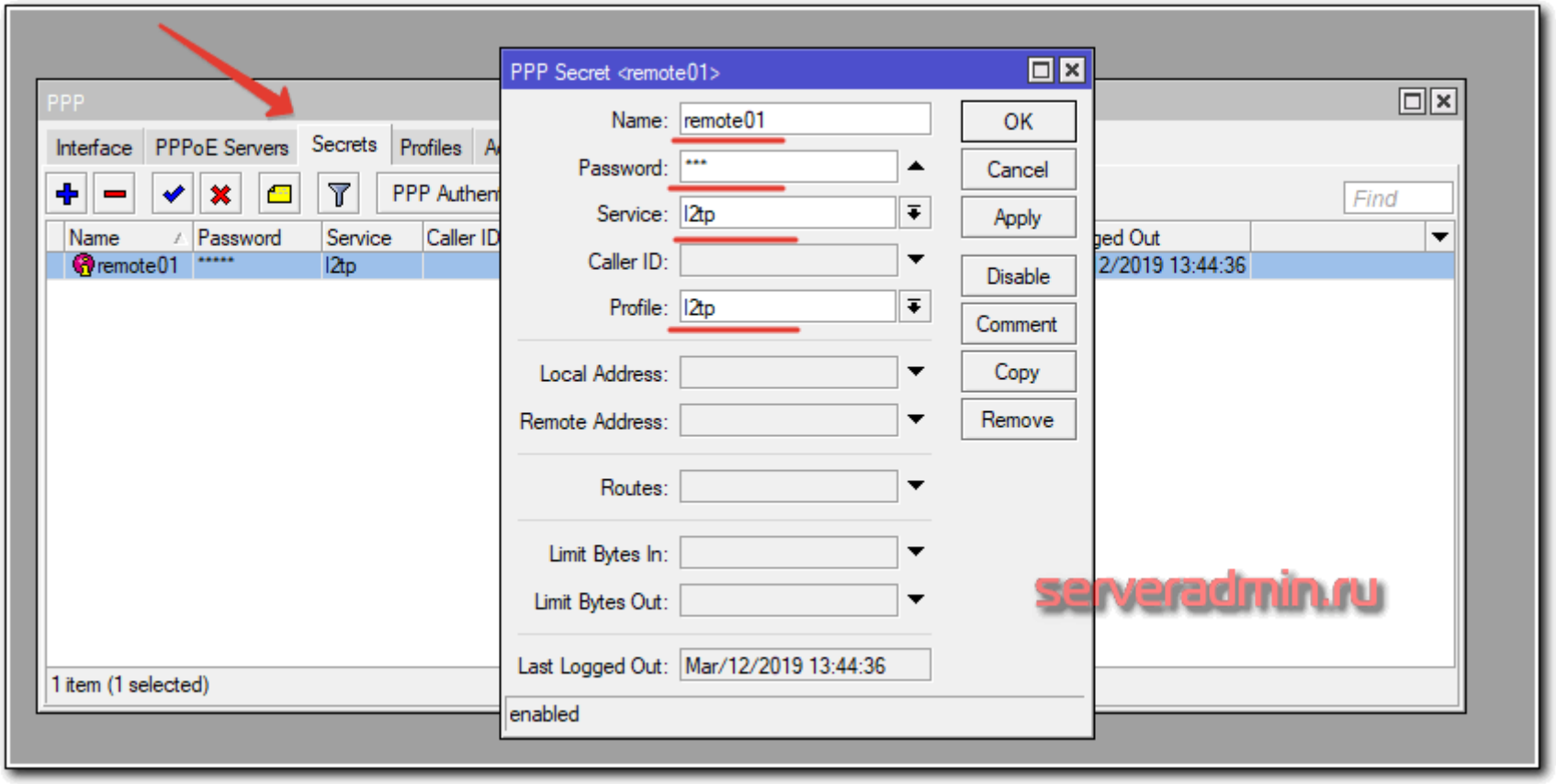
Идем в раздел **IP -> Pool** и добавляем пул ip адресов для vpn туннеля.



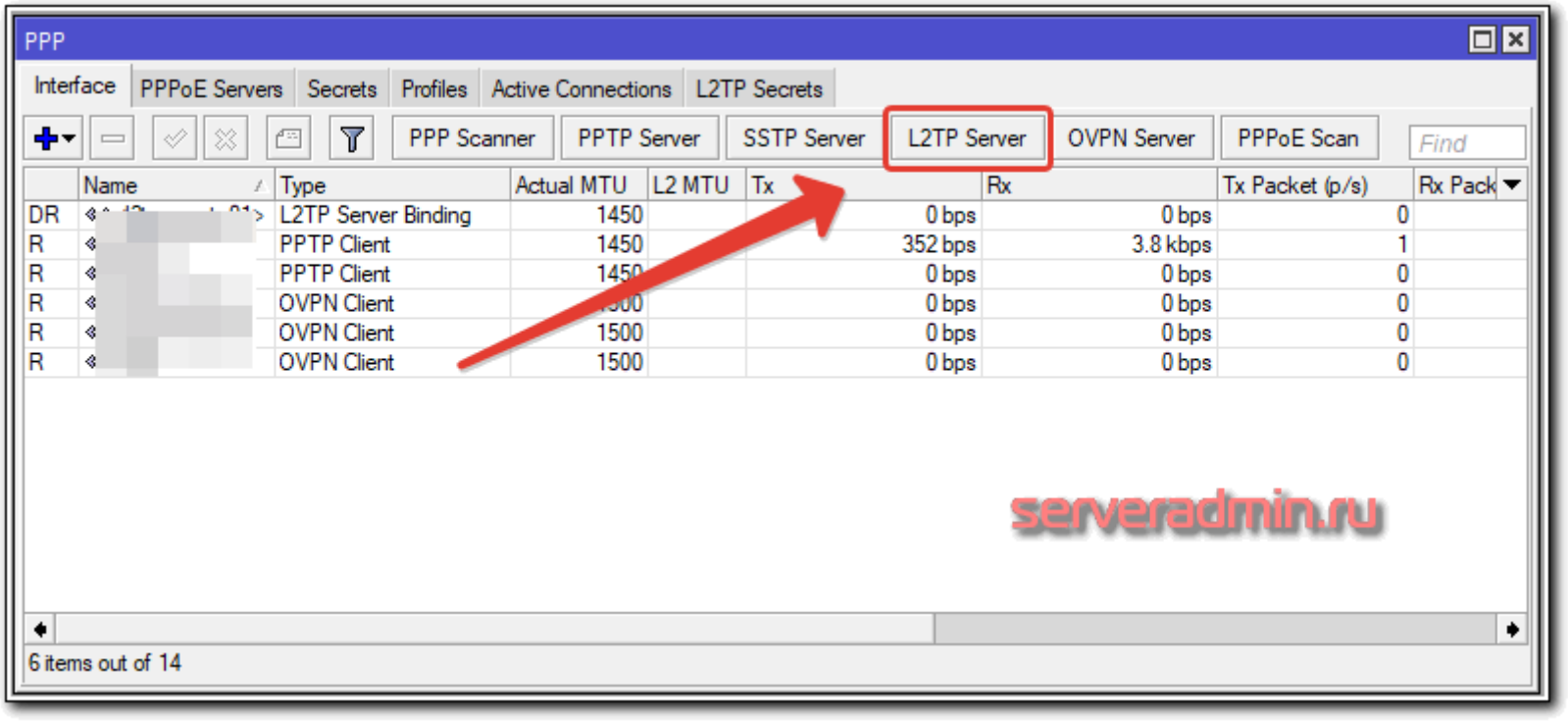
Создаем профиль для туннеля в **PPP -> Profiles**.


1
2
PPP
Interface
PPPoE Servers
Secrets
Profiles
Activ
Name
Local Address
Remote Addre
default
default-encr...
l2tp
vpn_pool
vpn_pool
3 items (1 selected)
Find
PPP Profile <l2tp>
General
Protocols
Limits
Queue
Scripts
Name: l2tp
Local Address: vpn_pool
Remote Address: vpn_pool
Bridge:
Bridge Port Priority:
Bridge Path Cost:
Bridge Horizon:
Incoming Filter:
Outgoing Filter:
Address List:
Interface List:
DNS Server:
WINS Server:
Change TCP MSS
no yes default
Use UPnP
no yes default
OK
Cancel
Apply
Comment
Copy
Remove
serveradmin.ru

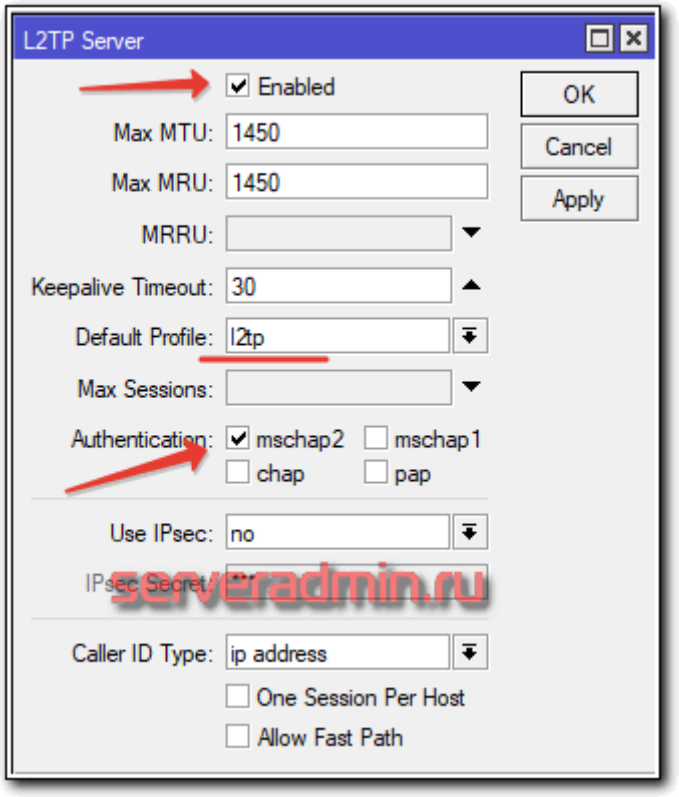
На остальных вкладках настройки дефолтные. Далее создаем пользователя в **PPP -> Secrets**.



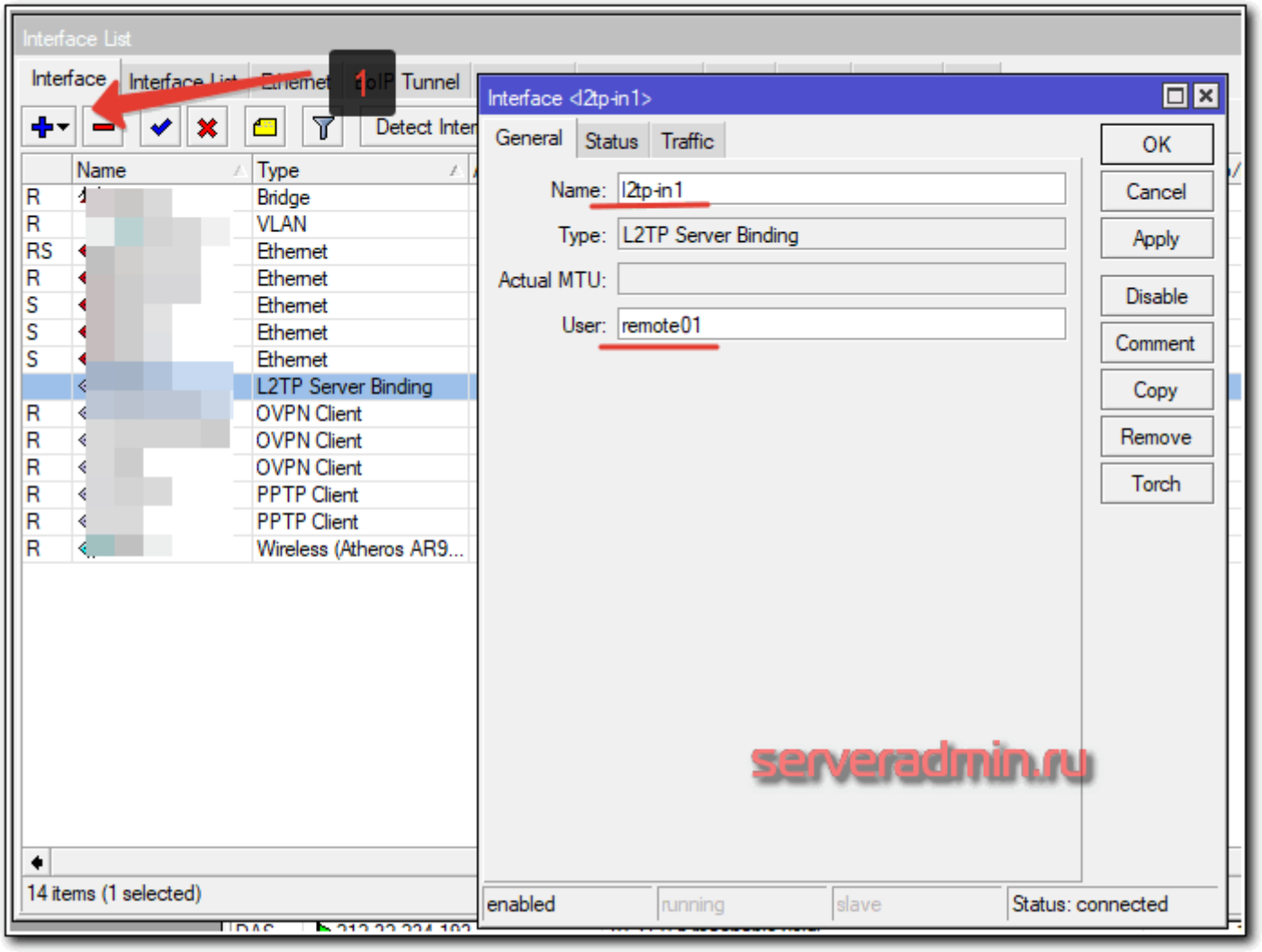
Теперь запускаем l2tp сервер. Идем в **PPP** и жмем в кнопку **L2TP Server**.



Устанавливаем настройки для l2tp сервера. ipsec пока не включаем.

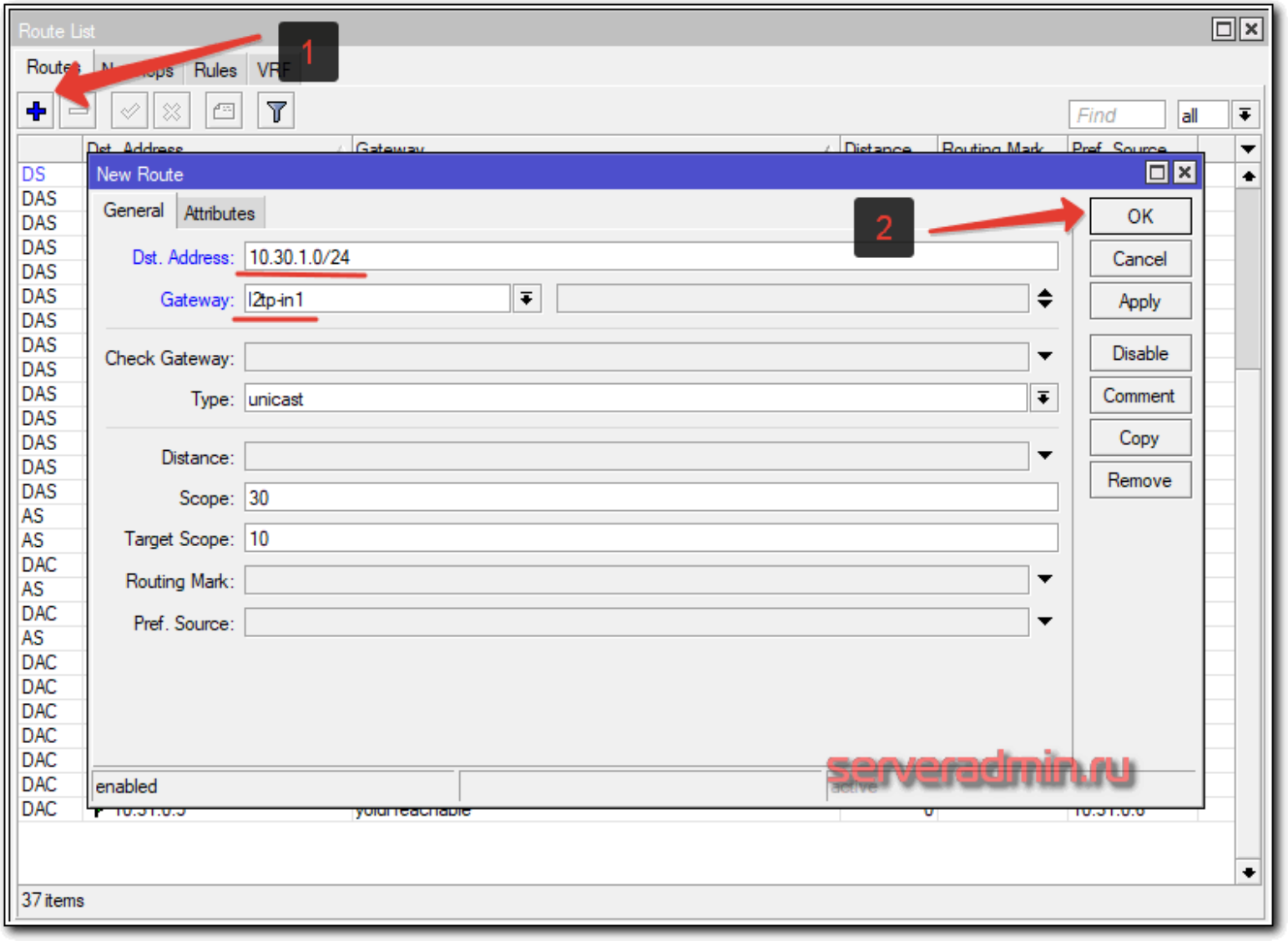


VPN сервер настроен. Теперь создадим для него постоянный интерфейс, чтобы на его основе создавать статические маршруты. Идем в **Interfaces** и создаем **L2tp Server Binding**.



Последний штрих. Создаем статический маршрут, с помощью которого абоненты локальной сети сервера смогут подключаться к абонентом локальной

сети за удаленным роутером через vpn. Идем в **IP -> Routes** и добавляем маршрут.


Route List
Rules
VRF
1
Find
all
New Route
General
Attributes
2
OK
Cancel
Apply
Disable
Comment
Copy
Remove
Dst. Address: 10.30.1.0/24
Gateway: l2tp-in1
Check Gateway:
Type: unicast
Distance:
Scope: 30
Target Scope: 10
Routing Mark:
Pref. Source:
enabled
serveradmin.ru
37 items

Я не рассмотрел вопрос настройки firewall, так как не хочется раздувать и так объемную статью. Напрямую это не относится к указанной теме. Подробнее читайте о настройке фаервола отдельно по приведенной ссылке. Здесь же только укажу, что необходимо открыть на firewall для корректной настройки l2tp.

На сервере необходимо создать следующие правила для фаерволла, чтобы мы могли достучаться до нашего L2TP сервера. **IP -> Firewall -> Filter Rules**. Необходимо создать разрешающее правило в цепочке input для следующих портов и протоколов:

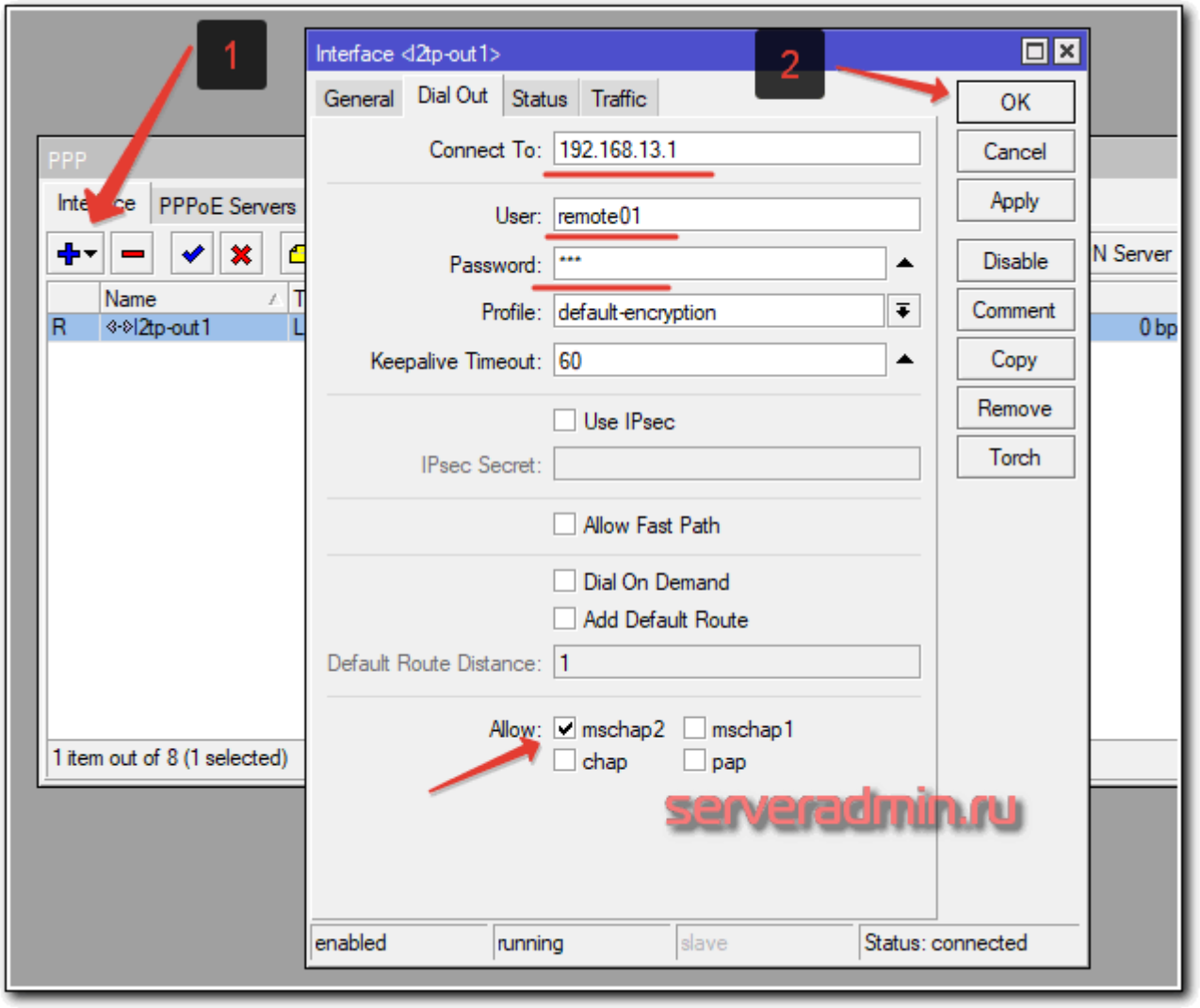
- Протокол: UDP
- Разрешаем порты: 1701,500,4500
- В качестве In.Interface указываем тот, через который происходит l2tp подключение.

Отдельно добавляем еще одно правило, разрешающее протокол **ipsec-esc**.

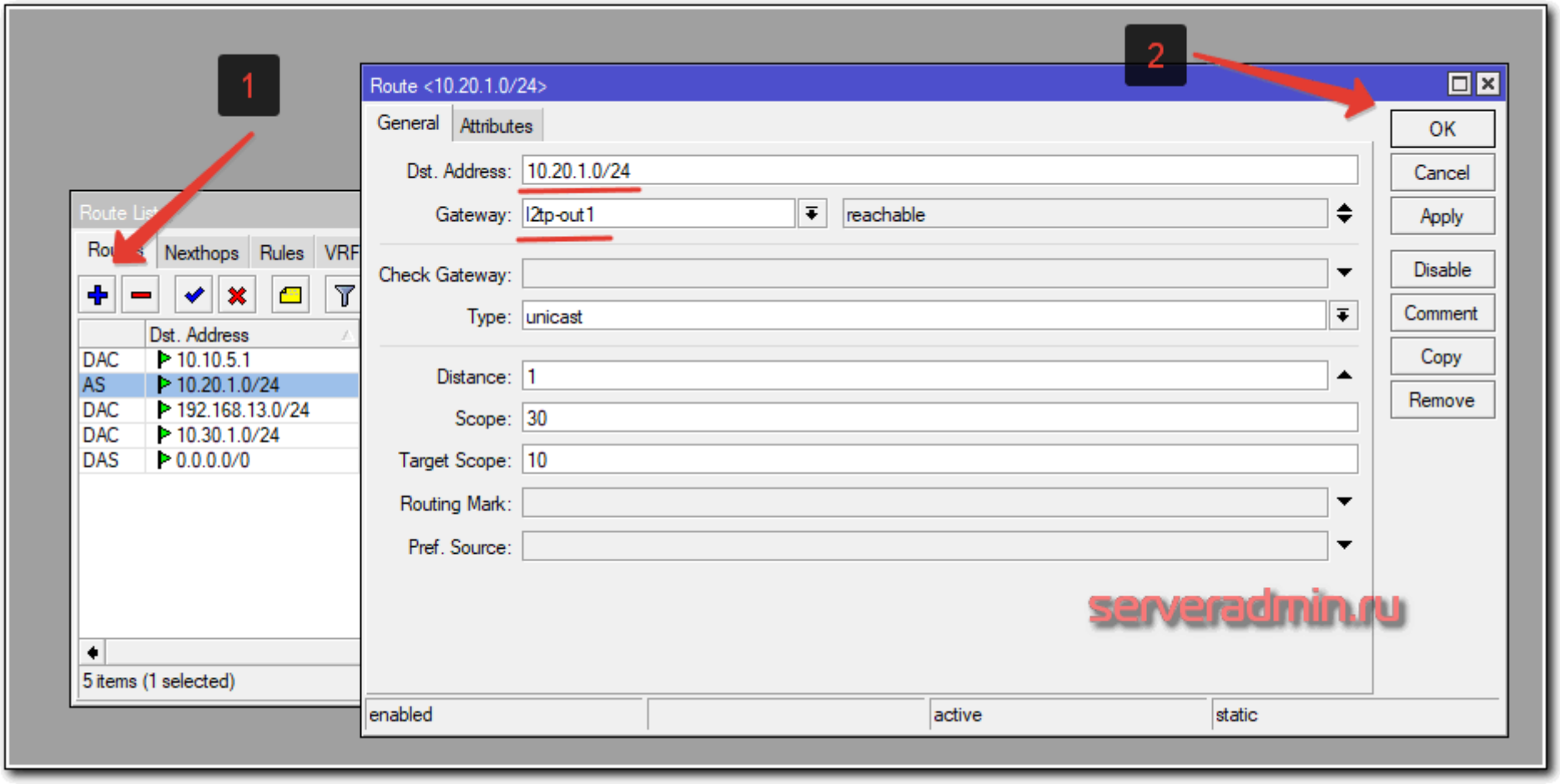
На сервере все готово. Идем настраивать l2pt клиент на удаленном микротике.

L2tp клиент

Здесь все достаточно просто. Идем в **PPP** и добавляем **L2TP Client**. Указываем настройки, которые задавали ранее на сервере.

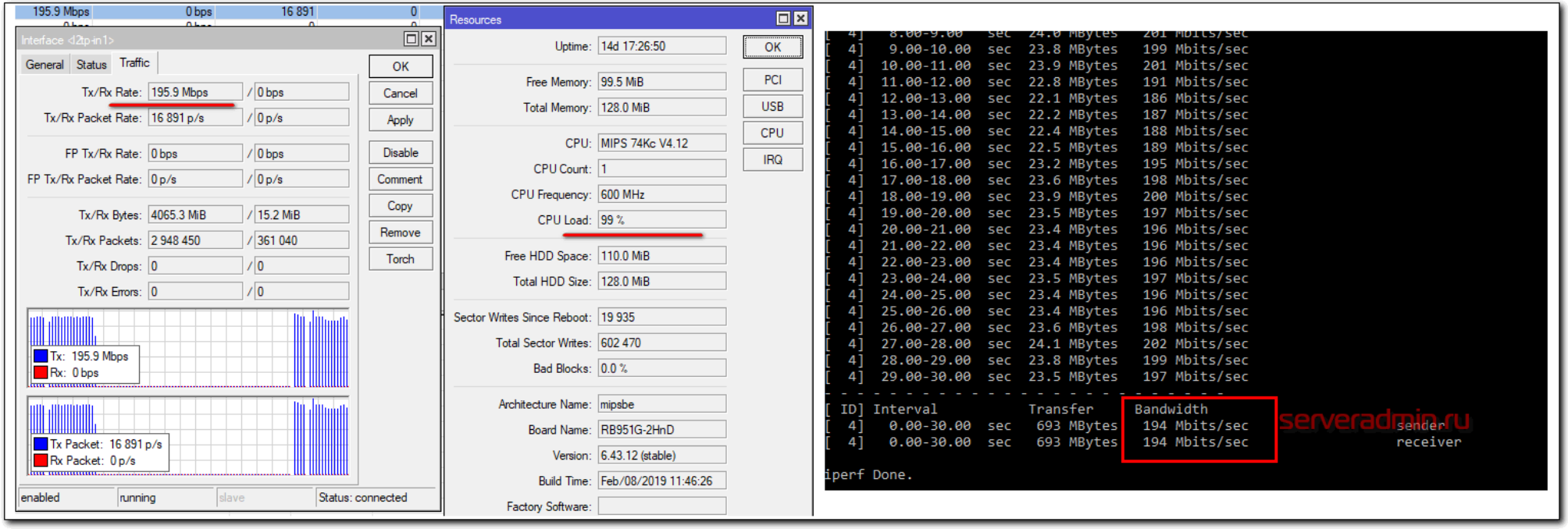

1
2
PPP
PPPoE Servers
Name
R
l2tp-out1
1 item out of 8 (1 selected)
Interface <l2tp-out1>
General
Dial Out
Status
Traffic
Connect To: 192.168.13.1
User: remote01
Password: ***
Profile: default-encryption
Keepalive Timeout: 60
Use IPsec
IPsec Secret:
Allow Fast Path
Dial On Demand
Add Default Route
Default Route Distance: 1
Allow: mschap2 mschap1 chap pap
OK
Cancel
Apply
Disable
Comment
Copy
Remove
Torch
enabled
running
slave
Status: connected
serveradmin.ru

Добавляем статический маршрут, чтобы клиенты этого роутера знали, куда обращаться к абонентам удаленной локальной сети за vpn.



На этом все. Мы настроили l2tp на удаленном микротике и таким образом объединили 2 локальных сети с помощью vpn. В списке ip адресов при активном l2tp соединении на сервере и клиенте вы должны увидеть ip адреса из заданного на сервере диапазона для vpn сети — 10.10.5.1-10.10.5.100. Теперь можно пропинговать с обоих сетей противоположные.

У меня для теста к обоим микротикам подключены ноутбуки. Сейчас я измерю скорость соединения с помощью iperf3. За роутером **m-remote** на ноутбуке 10.30.1.254 запускаю сервер, а на 10.20.1.3 агента. Запускаем тест скорости vpn соединения:

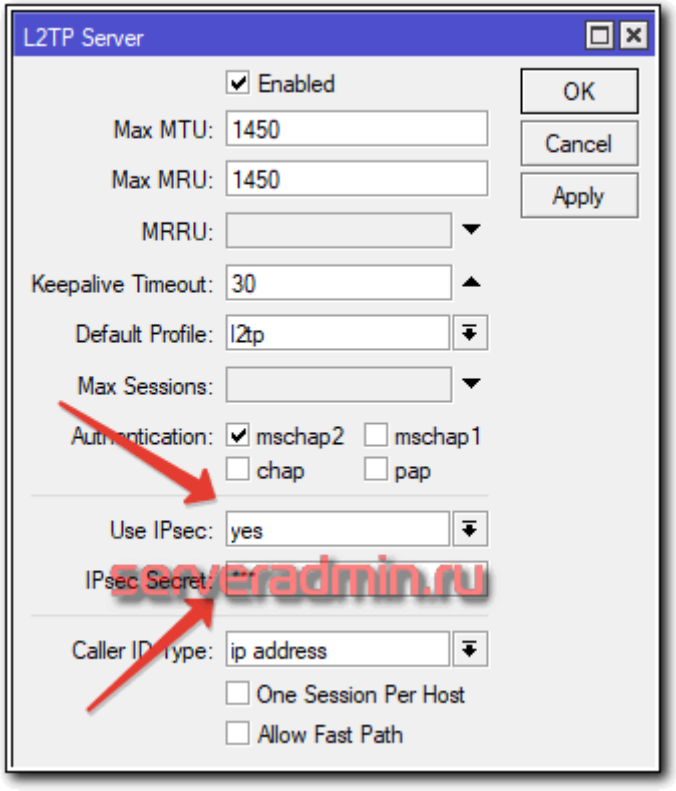


Средняя скорость **194 мбит/сек**. Откровенно говоря, я не понял, почему такая низкая скорость. Мой тестовый стенд собран на двух роутерах микротиках и гигабитного микротик свитча между ними. Ожидал увидеть что-то в районе 500 мбит/сек. Напомню, что туннель пока без шифрования. При этом загрузка процессоров на роутерах была в районе 90-95%. То есть фактически потолок этих железок.

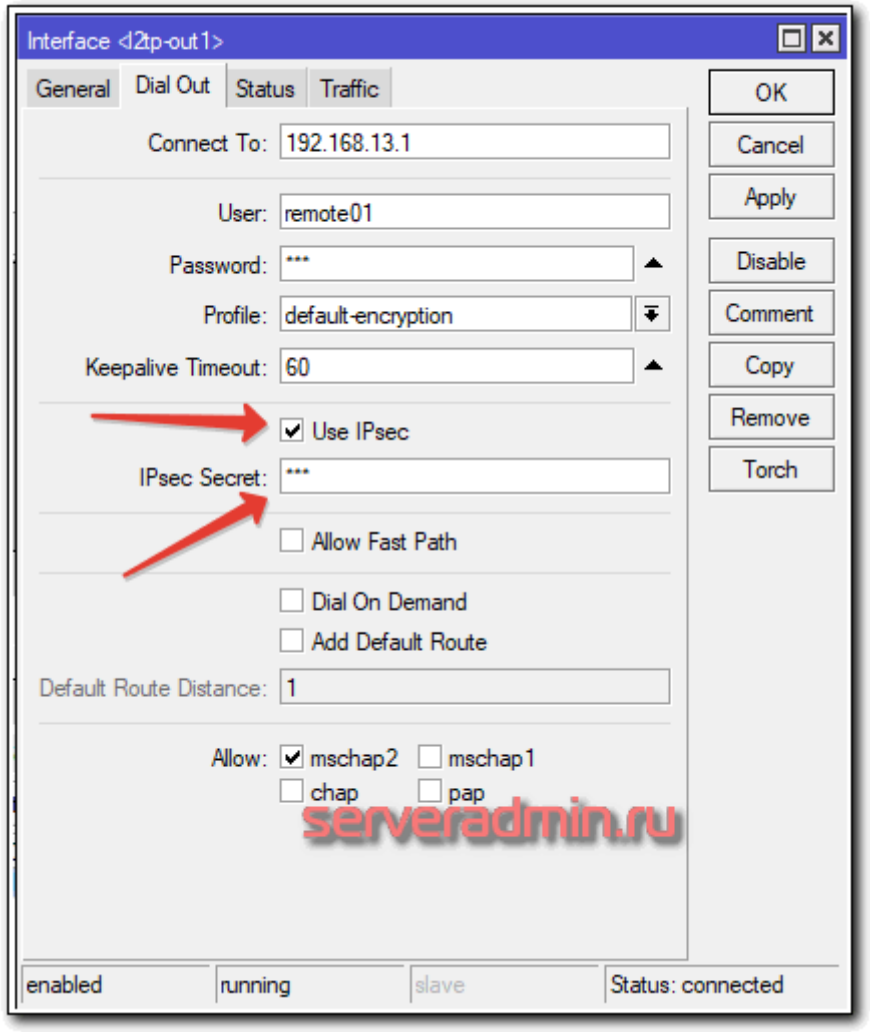
Попробуем теперь включить шифрование ipsec и замерить скорость с ним.

Настраиваем ipsec

С настройкой ipsec для l2tp я залип на некоторое время. В сети много инструкций, но все они устарели. Как оказалось, в последних версиях прошивок, запустить ipsec в дефолтных настройках не просто, а очень просто. Для этого надо всего лишь в свойствах l2tp сервера указать **Use IPsec** — yes и задать пароль.



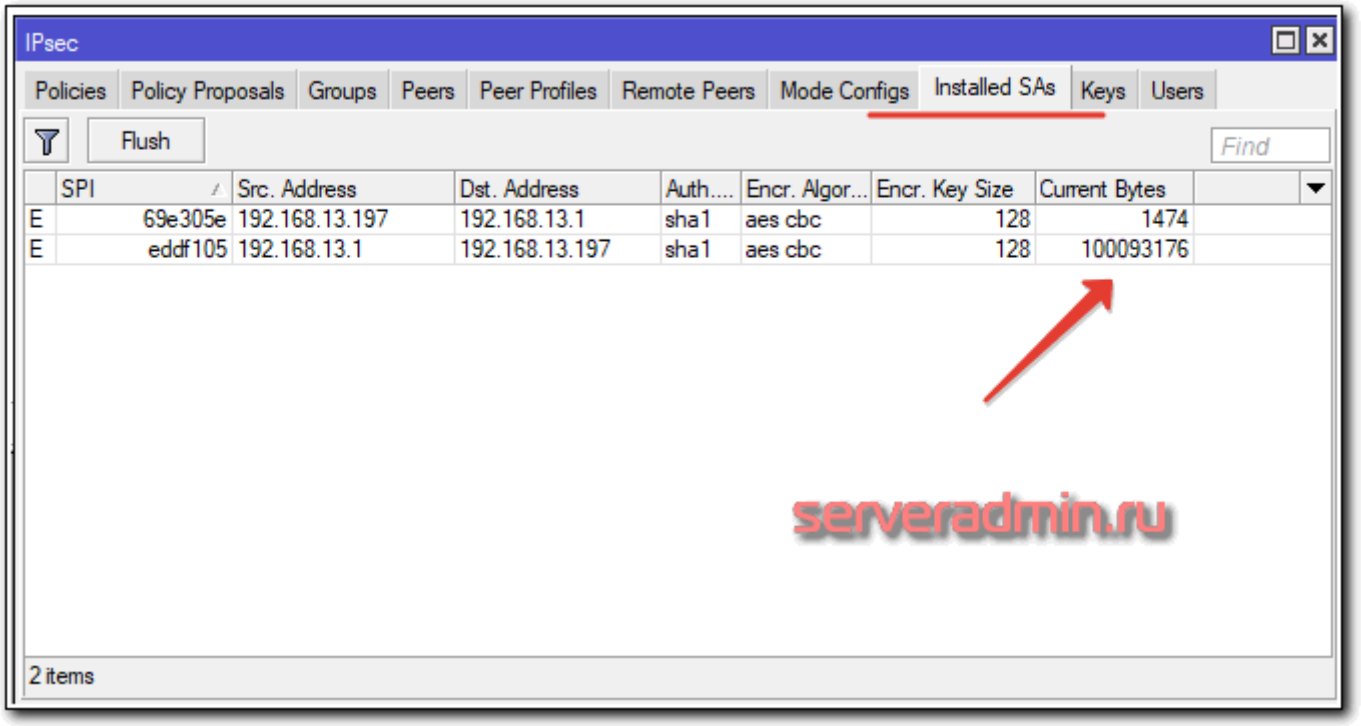
Все необходимые настройки ipsec будут созданы автоматически. На агенте сделать то же самое — включить ipsec шифрование и указать пароль.



После подключения l2tp клиента увидите в логе похожие строки:

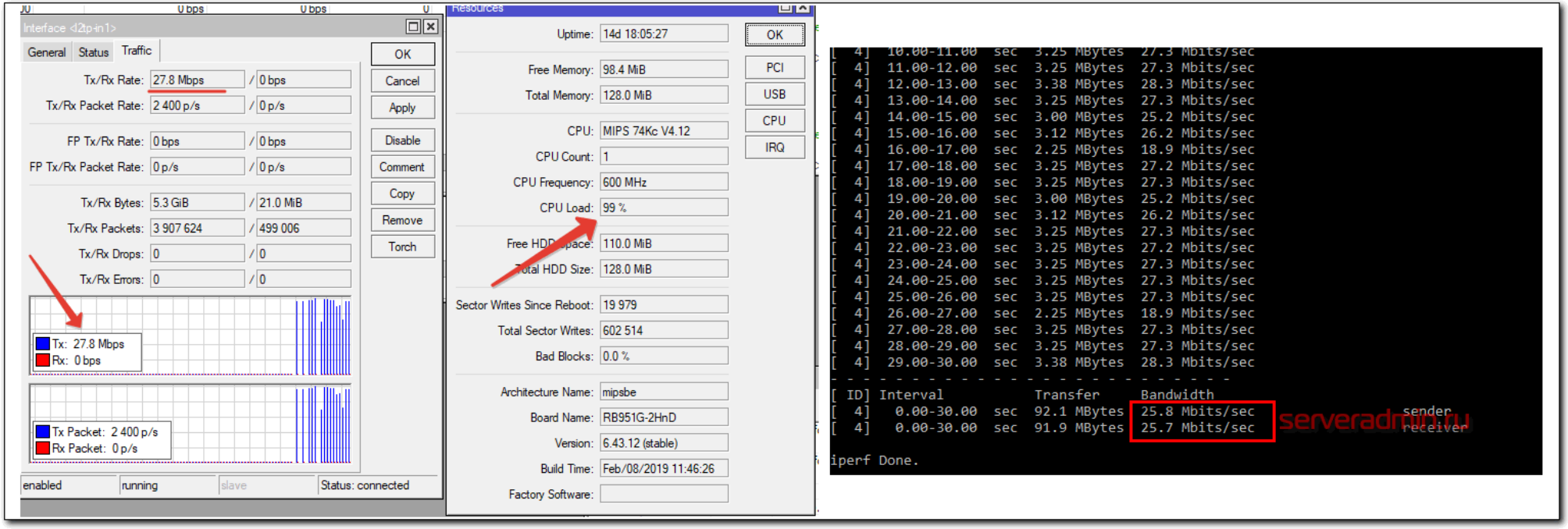
```
19:17:00 l2tp,ppp,info l2tp-out1: initializing...
19:17:00 l2tp,ppp,info l2tp-out1: connecting...
19:17:03 ipsec,info initiate new phase 1 (Identity Protection): 192.168.13.197[500]<=>192.168.13.1[500]
19:17:04 ipsec,info ISAKMP-SA established 192.168.13.197[500]-192.168.13.1[500] spi:407844c0ceb5d2ab:46ce7ffb25495efd
19:17:07 l2tp,ppp,info l2tp-out1: authenticated
19:17:07 l2tp,ppp,info l2tp-out1: connected
```

Для того, чтобы убедиться, что шифрование ipsec работает, можно зайти в раздел **IP -> Ipsec -> Installed SAs** и посмотреть на счетчик зашифрованных пакетов. Если он растет, значит все в порядке, трафик шифруется.



Там же в разделе **Remote Peers** можно посмотреть список удаленных клиентов, для которых работает ipsec шифрование, посмотреть используемые алгоритмы. Все дефолтные настройки ipsec живут в этом разделе. Вы можете посмотреть их, изменить или добавить новые профили. По-умолчанию используется алгоритм авторизации sha1 и шифрование AES. Можете изменить эти параметры, если разбираетесь в теме. Я умничать не буду, тему шифрования не копал. Какие алгоритмы максимально быстры и защищены — не знаю.

Проведем тесты скорость vpn соединения l2tp + ipsec.



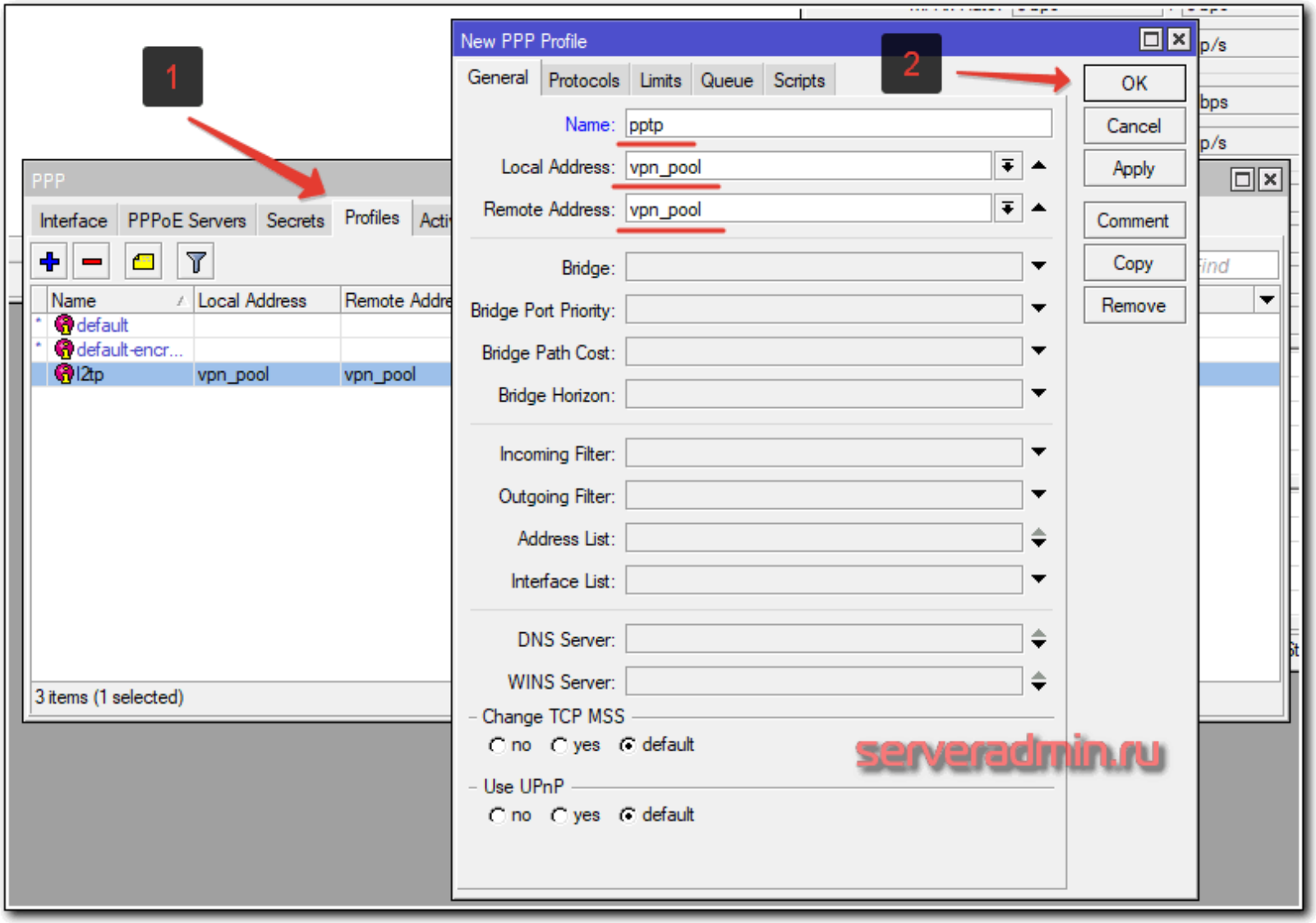
У меня получилось вот так — **26 мбит/сек** в среднем. При этом загрузка процессора 100%. Не густо. Данные железки для шифрованных каналов пригодны очень слабо. В данных тестах они ничем, кроме непосредственно теста не нагружены. В реальных условиях скорость будет еще ниже.

С настройками vpn на базе l2tp + ipsec закончили. Продолжим настройку остальных vpn туннелей и сравним их скорость.

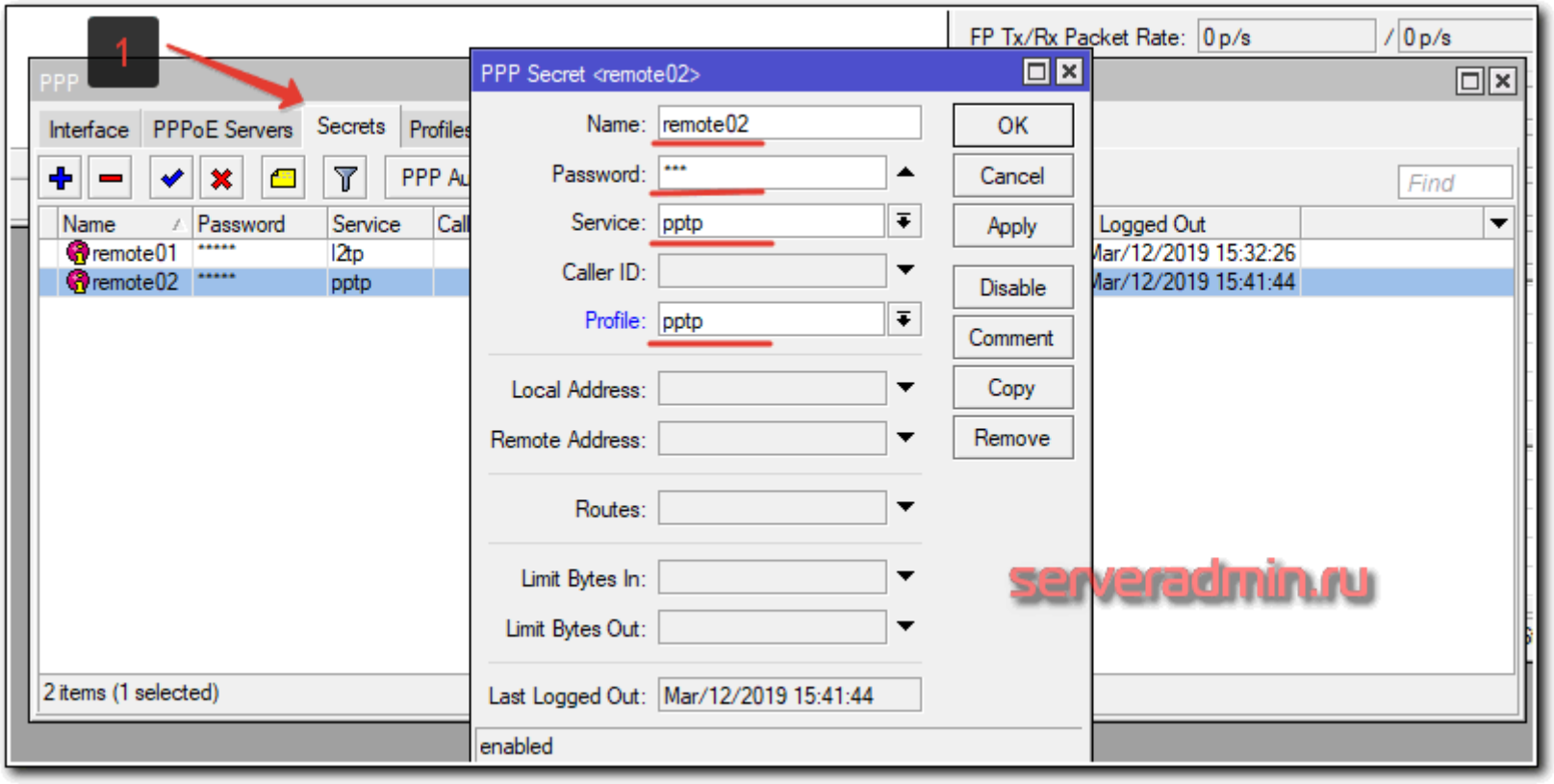
Настройка pptp сервера в mikrotik

Настройка pptp сервера не отличается принципиально от l2tp. Логика и последовательность действий та же самая. Сначала создаем pool адресов в **IP -> Pool** для vpn сети. Я буду использовать тот же пул, что мы создали ранее.

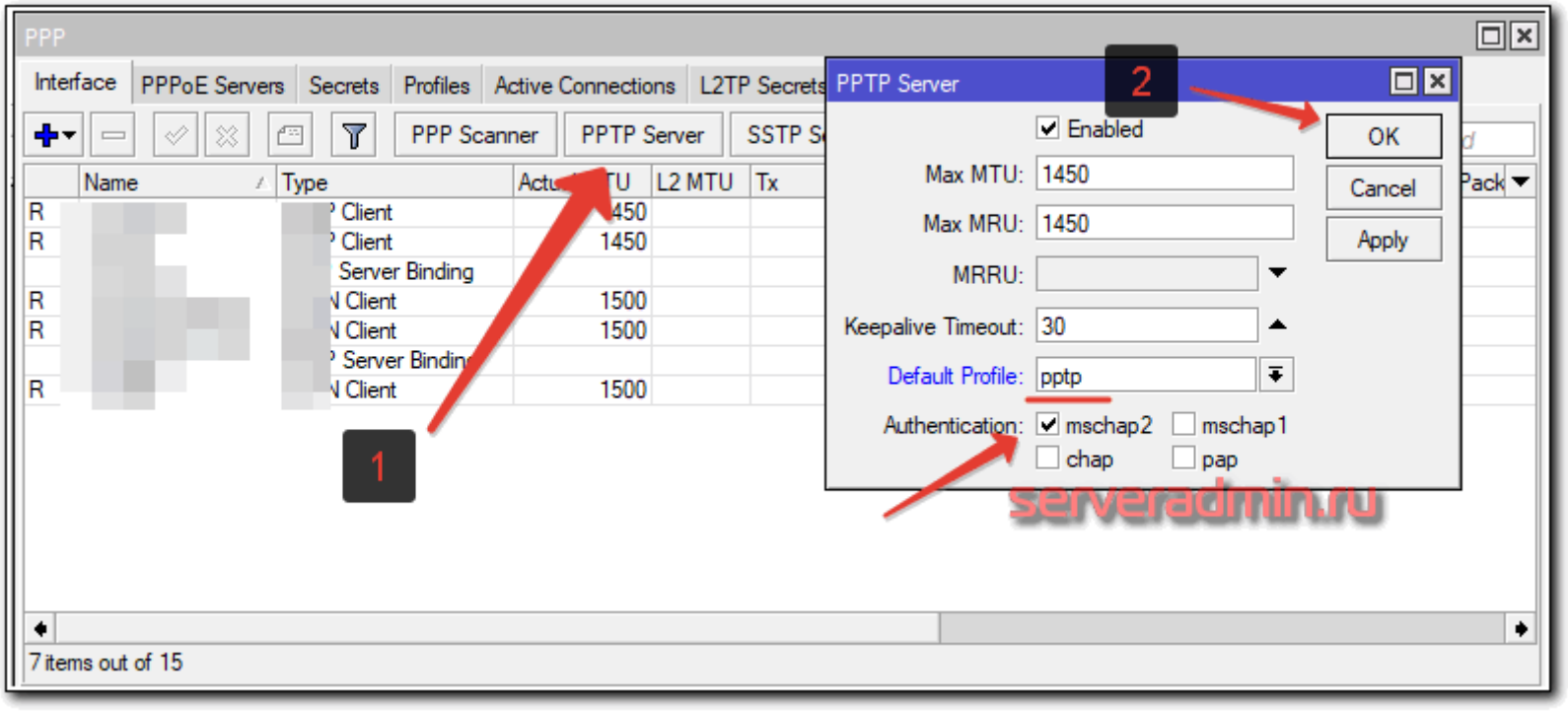
Далее создаем профиль для pptp туннеля в разделе **PPP -> Profiles**.


1
2
PPP
Interface
PPPoE Servers
Secrets
Profiles
Name
Local Address
Remote Addr
default
default-encr...
l2tp
vpn_pool
vpn_pool
3 items (1 selected)
New PPP Profile
General
Protocols
Limits
Queue
Scripts
Name: pptp
Local Address: vpn_pool
Remote Address: vpn_pool
Bridge:
Bridge Port Priority:
Bridge Path Cost:
Bridge Horizon:
Incoming Filter:
Outgoing Filter:
Address List:
Interface List:
DNS Server:
WINS Server:
Change TCP MSS
no
yes
default
Use UPnP
no
yes
default
OK
Cancel
Apply
Comment
Copy
Remove
serveradmin.ru

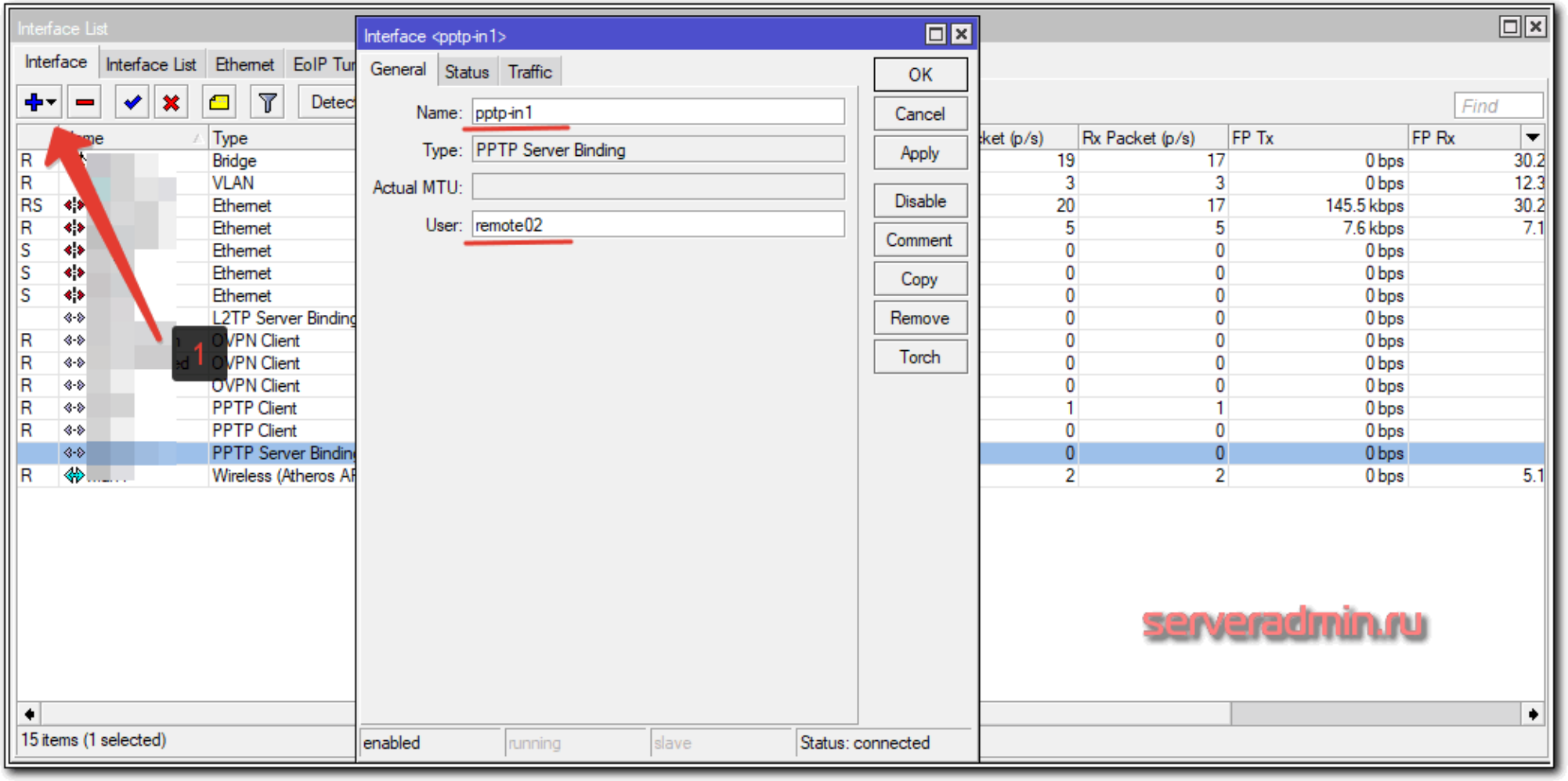
В этом профиле указаны дефолтные настройки шифрования, при которых оно отключено. Проверим сначала скорость vpn канала без них. Создаем нового пользователя для удаленного pptp подключения.



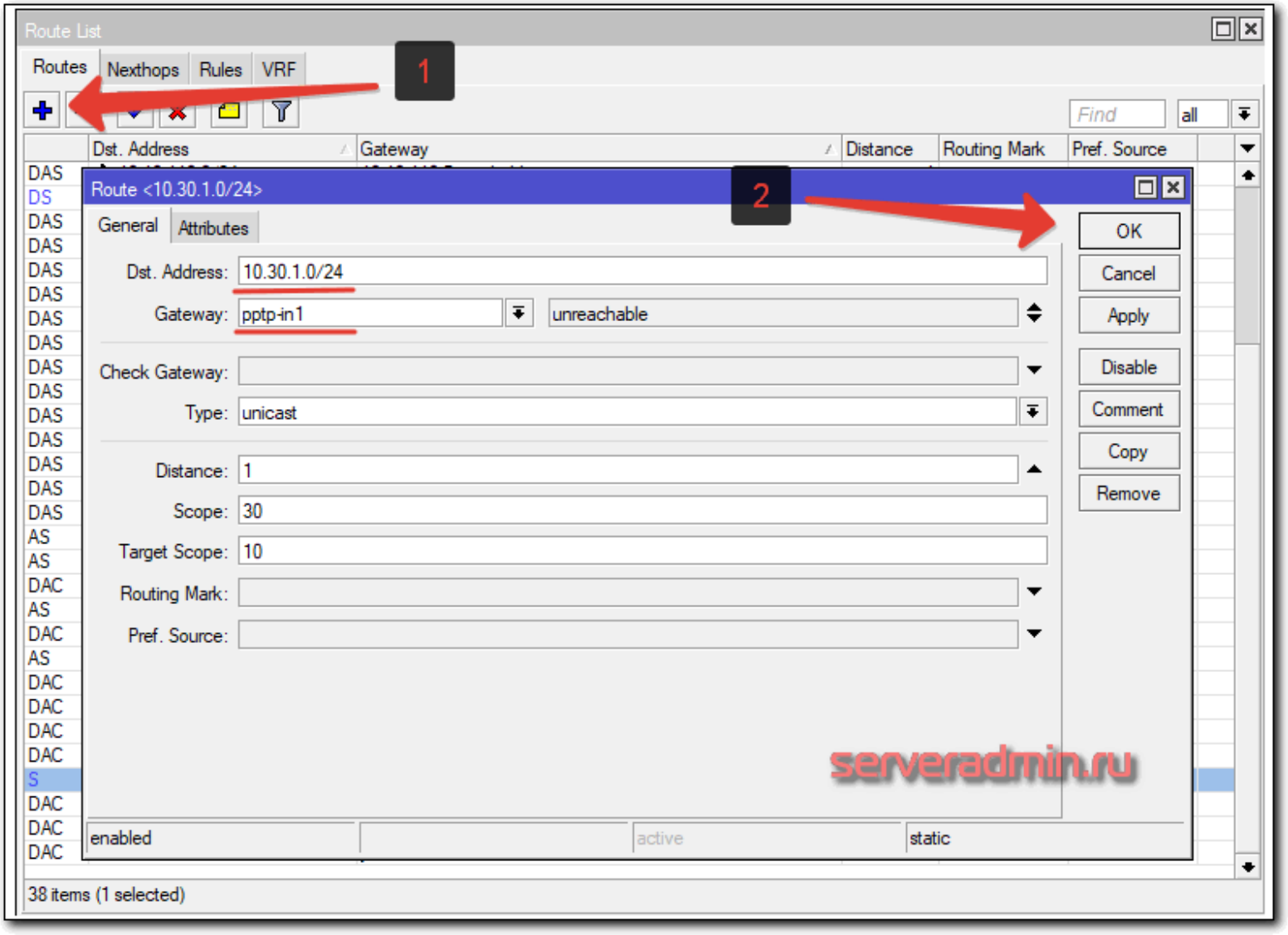
Включаем pptp сервер в разделе PPP.



Теперь создадим в Interface List **PPTP Server Binding** по аналогии с предыдущим разделом.



И в завершение добавляем статический маршрут до удаленной сети через pptp подключение.


Route List
Routes
Nexthops
Rules
VRF
1
Find
all
Dst. Address
Gateway
Distance
Routing Mark
Pref. Source
Route <10.30.1.0/24>
2
General
Attributes
OK
Cancel
Apply
Disable
Comment
Copy
Remove
Dst. Address: 10.30.1.0/24
Gateway: pptp-in1
unreachable
Check Gateway:
Type: unicast
Distance: 1
Scope: 30
Target Scope: 10
Routing Mark:
Pref. Source:
serveradmin.ru
enabled
active
static
38 items (1 selected)

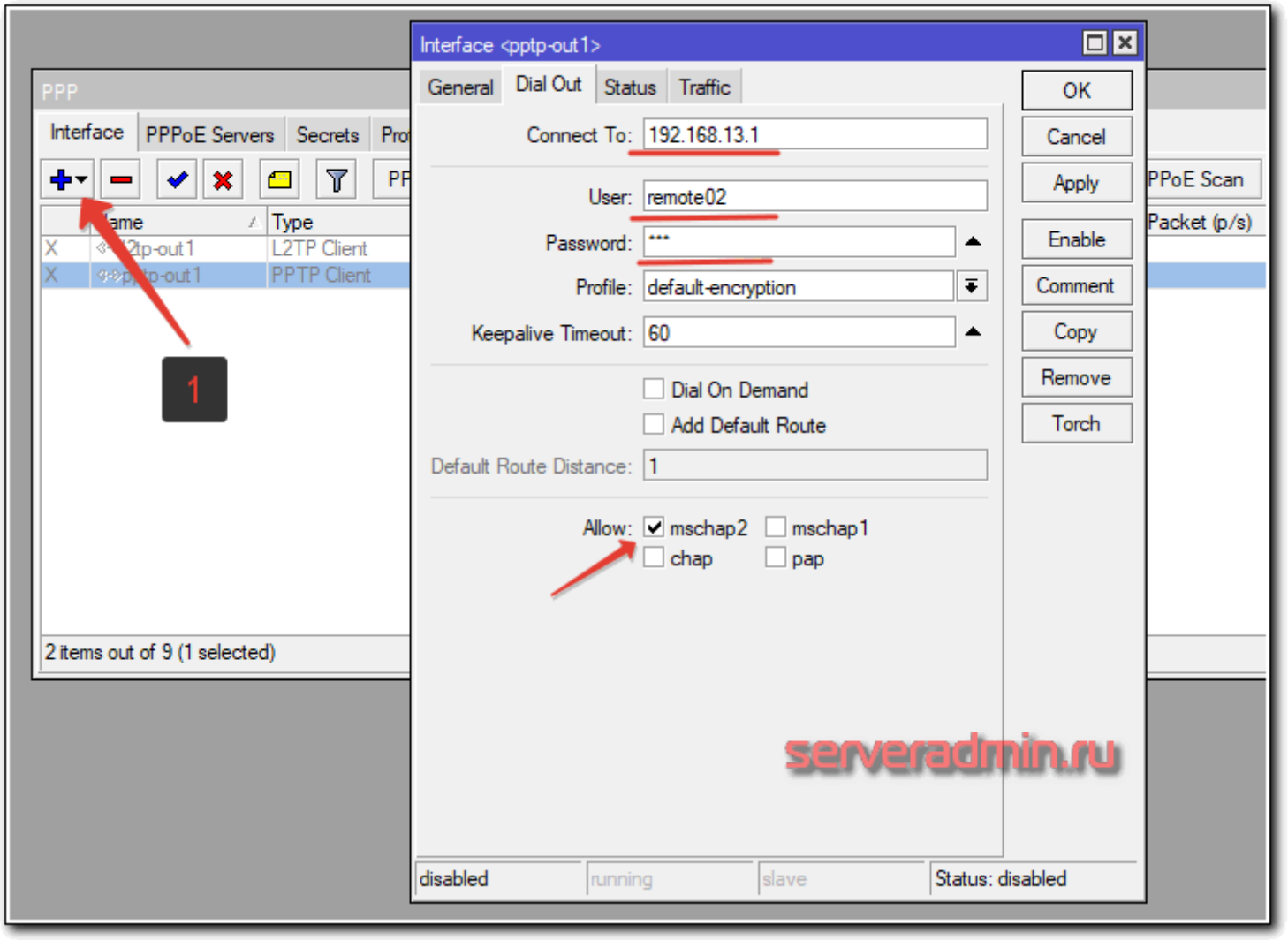
Настройка pptp сервера закончена. На фаерволе нужно будет открыть для входящих подключений внешнего интерфейса следующие вещи:

- TCP port 1723
- GRE протокол

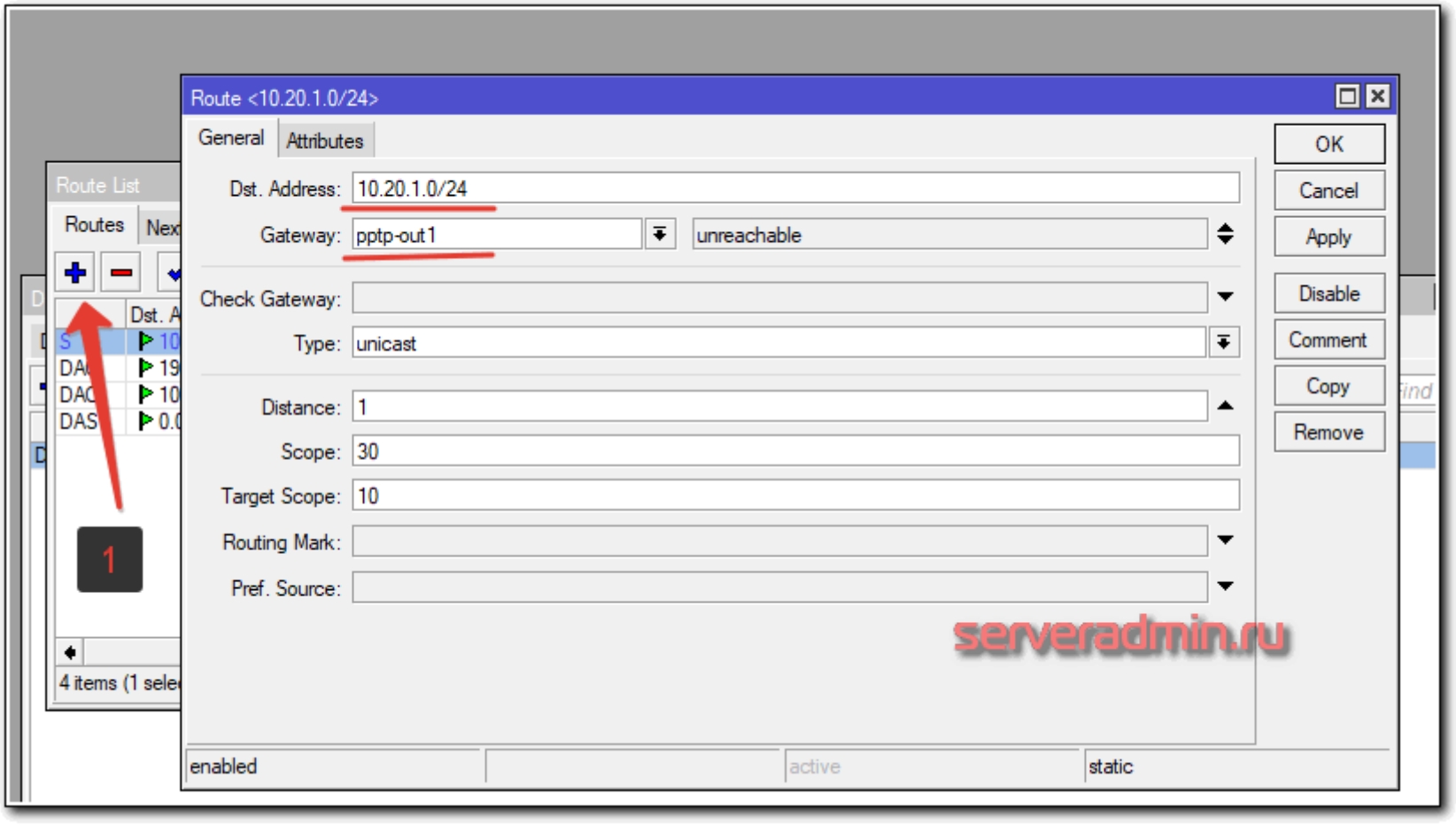
Отправляемся настраивать pptp клиент.

pptp клиент

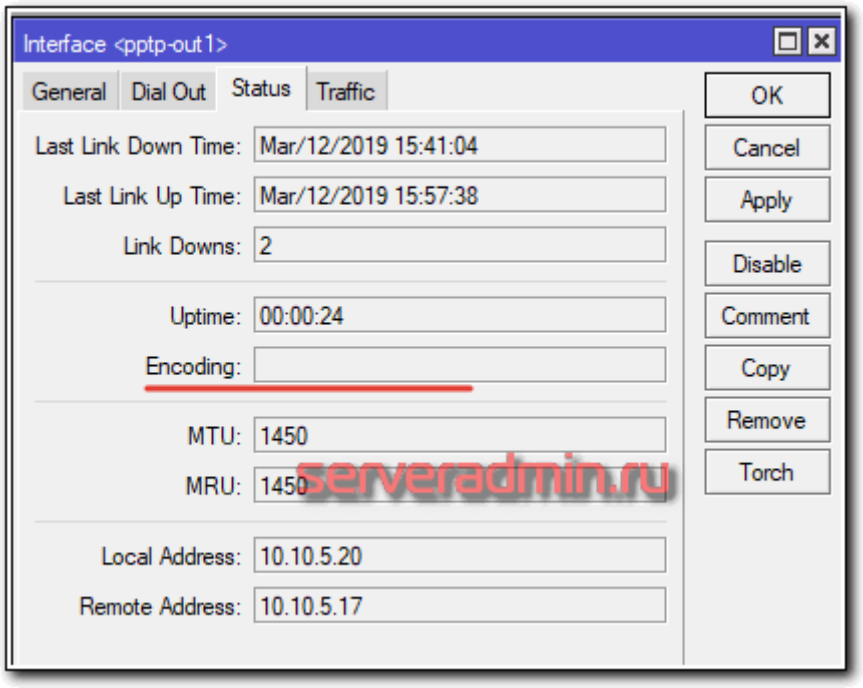
Отправляемся на удаленный роутер и там настраивает подключение через pptp client. Идем, как обычно, в раздел **PPP** и добавляем **PPTP Client**. На вкладке General ничего не трогаем, а на Dial Out указываем адрес pptp сервера и имя пользователя для подключения.



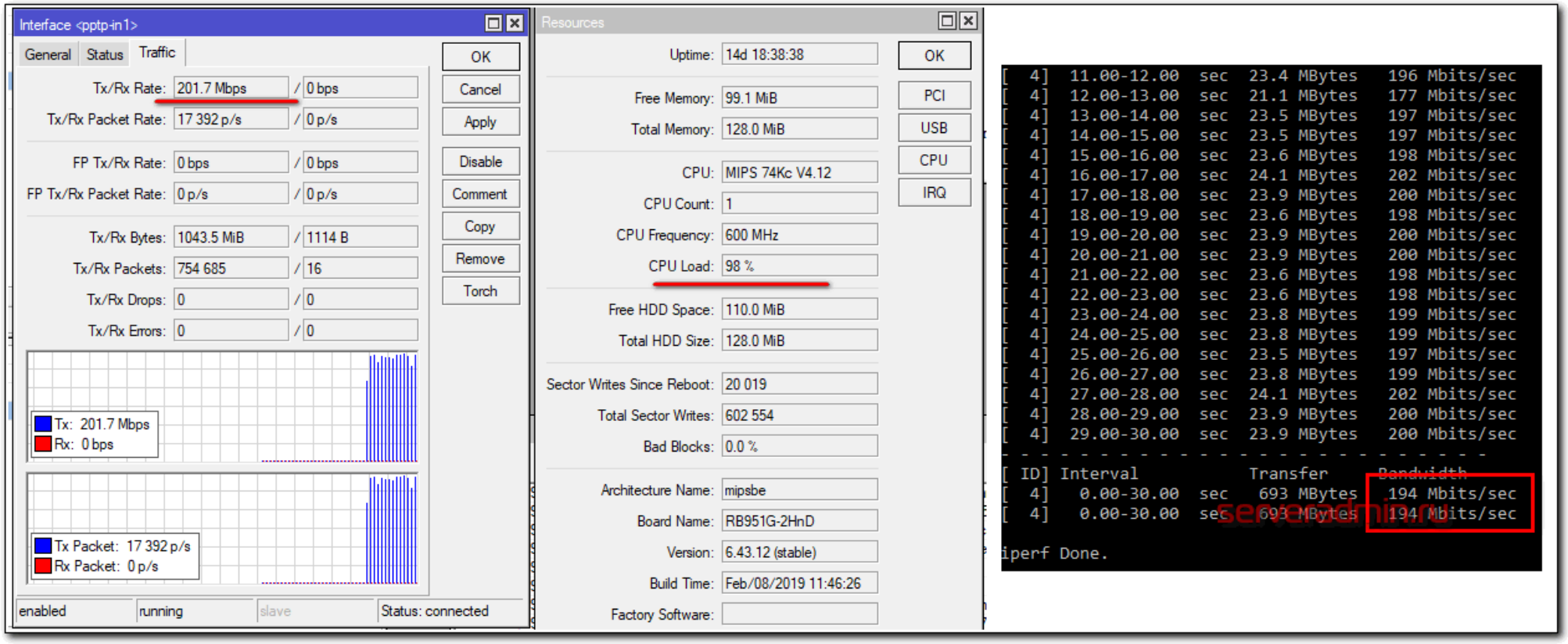
Добавляем статический маршрут до удаленного офиса через vpn туннель.



Все готово. Активируем pptp подключение и пробуем пинговать адреса в локальной сети. Убедиться в том, что шифрование отключено можно в статуте pptp соединения на клиенте.

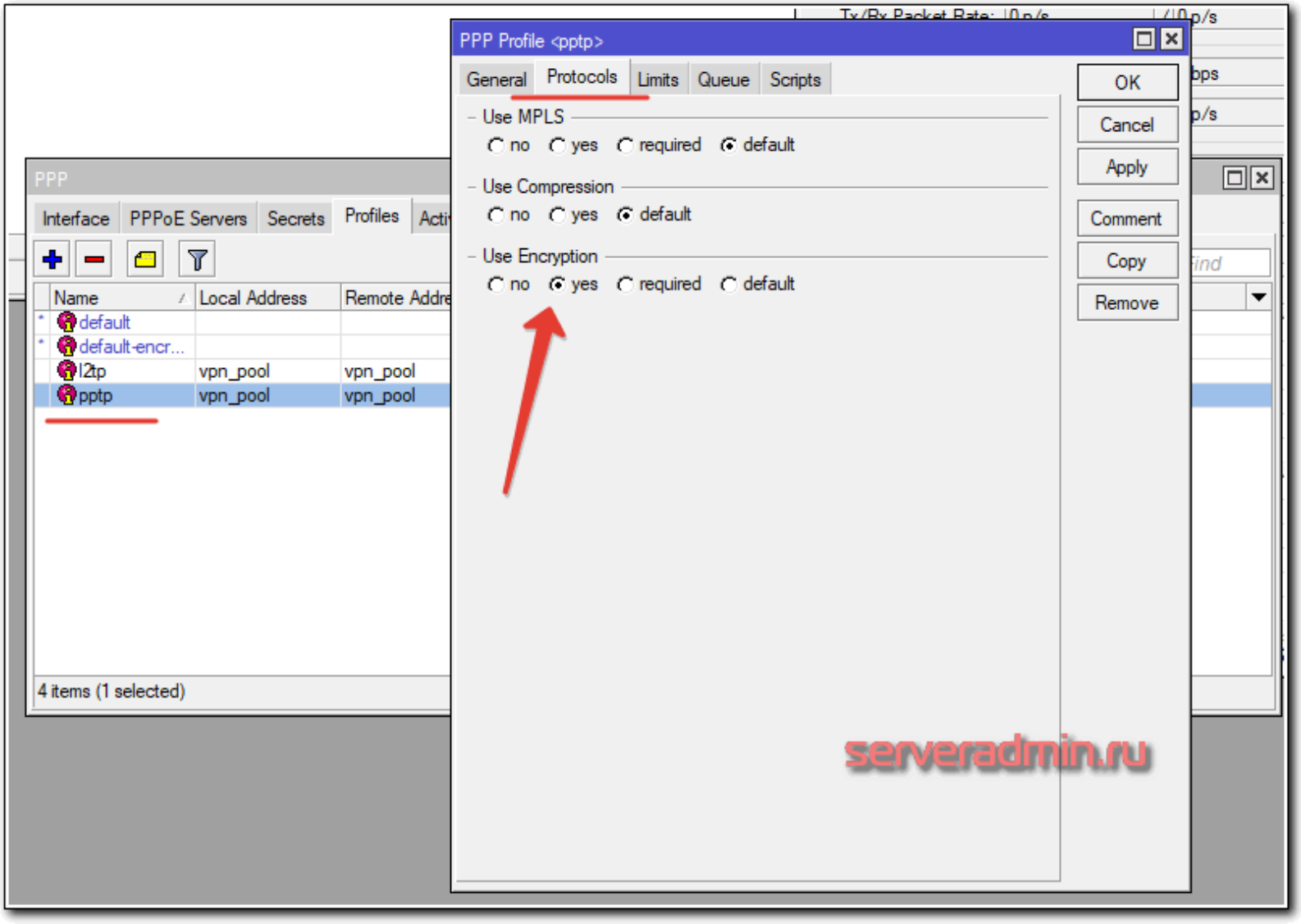


Проверим теперь скорость vpn соединения по pptp.

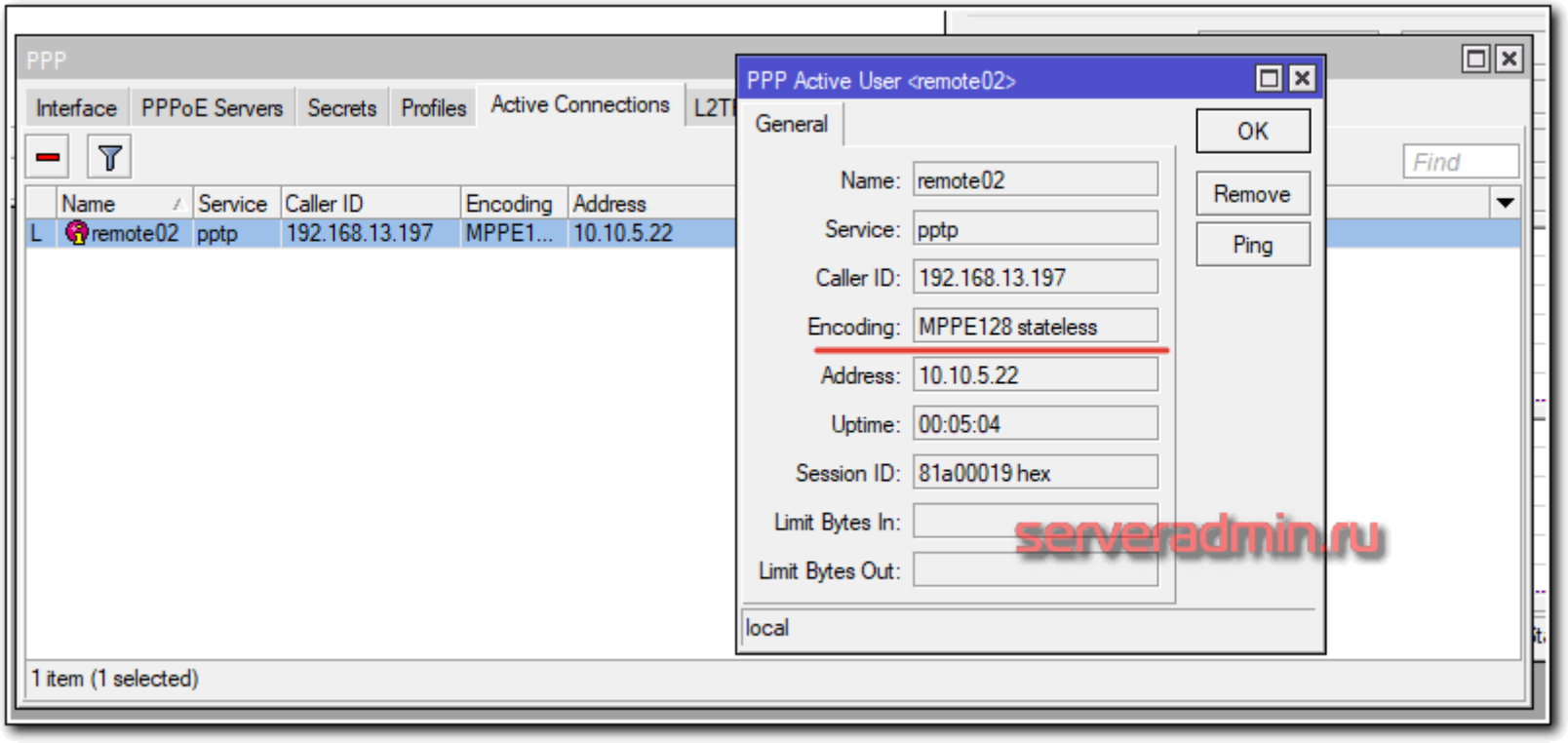


Те же самые **194 мбит/сек**, что на нешифрованном l2tp при 100% загрузке процессора. Вообще, было немного странно увидеть абсолютно такие же цифры. Проводил тесты несколько раз, но везде был стабильно один и тот же результат. Без шифрования нет разницы по скорости между l2tp и pptp соединением.

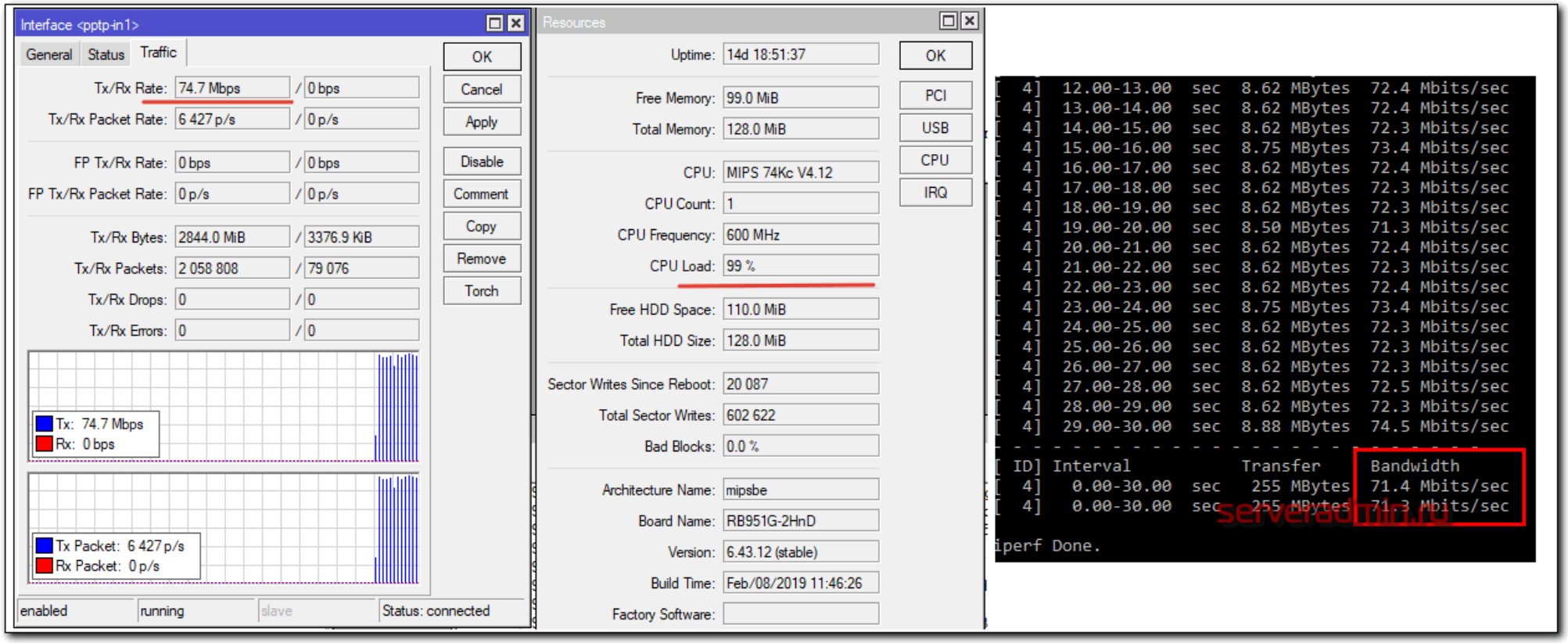
Теперь включим шифрование в pptp на сервере и посмотрим на скорость. Для этого указываем в pptp профиле явно, чтобы использовалось шифрование. Идем в **PPP -> Profiles** и редактируем наш профиль.


PPP
Interface
PPPoE Servers
Secrets
Profiles
Name
Local Address
Remote Addre
default
default-encr...
l2tp
vpn_pool
vpn_pool
pptp
vpn_pool
vpn_pool
4 items (1 selected)
PPP Profile <pptp>
General
Protocols
Limits
Queue
Scripts
OK
Cancel
Apply
Comment
Copy
Remove
Use MPLS
no
yes
required
default
Use Compression
no
yes
default
Use Encryption
no
yes
required
default
serveradmin.ru

Убедимся в статусе клиента, что шифрование работает.



Тестирую скорость vpn соединения по pptp с включенным шифрованием.



Получилось в среднем **71 мбит/сек**. Неплохой результат в сравнении с шифрованием ipsec в l2tp. Как я и говорил ранее, pptp сервер хорошо подходит там, где шифрование либо совсем не нужно, либо допускается возможность, что зашифрованный трафик будет расшифрован. Но при этом он все равно закрыт шифрованием и каждый проходящий не сможет ничего увидеть. Нужно как минимум снять дампт трафика и каким-то образом подбирать ключ по словарю или перебором. Не знаю точно, как это реализуется на практике. Не изучал вопрос.

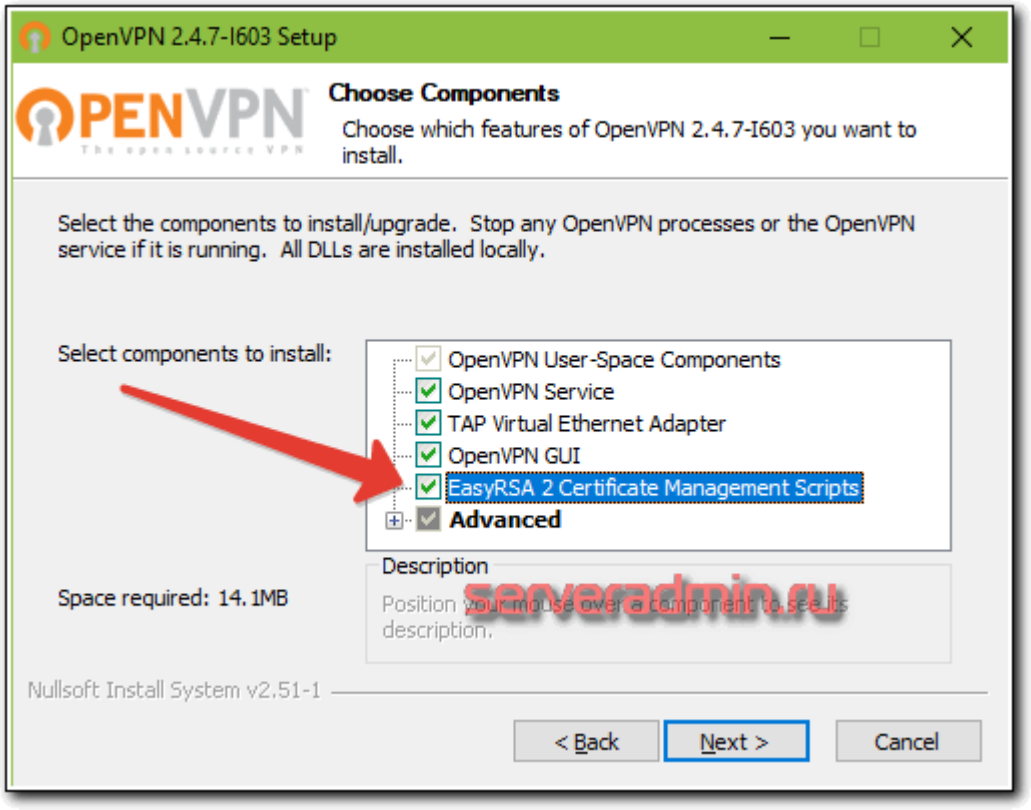
Перейдем теперь к openvpn серверу в микротик. Очень любопытно посмотреть на тесты скорости этого типа vpn соединений.

Настройка openvpn server в микротик

В настройке openvpn сервера на mikrotik нет ничего сложного, кроме нюанса с сертификатами. Тому, кто с ними никогда не работал, может показаться все слишком замороченным. К тому же в самом микротике нет никаких средств для создания сертификатов сервера и клиента. Необходимо использовать сторонние утилиты. Если у вас есть linux машина, можете воспользоваться моей инструкцией по созданию сертификатов для openvpn на linux.

Если у вас нет linux машины, но вы все же настроены поднять vpn туннель с помощью openvpn в микротике, то давайте разбираться с настройкой дальше. Прежде всего нам понадобится дистрибутив openvpn для windows. Скачать его можно по ссылке — https://openvpn.net/community-downloads/. Нас будет интересовать Windows Installer.

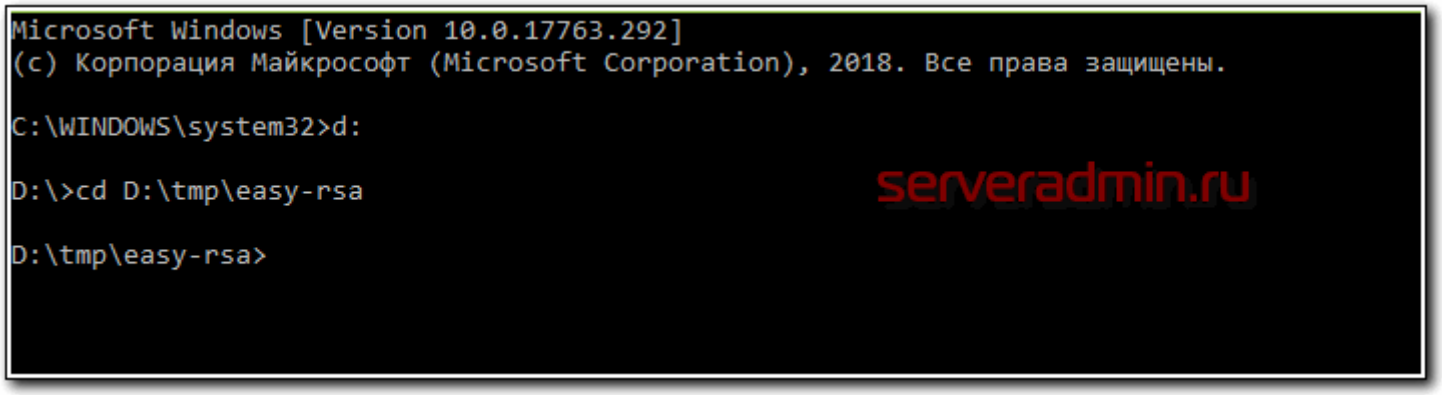
Выполняем установку от имени администратора и указываем в процессе компонент под названием **EasyRSA 2 Certificate Management Scripts**.



Идем в директорию *C:\Program Files\OpenVPN*. Переносим оттуда папку *easy-rsa* куда-нибудь в другое место, чтобы не приходилось постоянно спотыкаться об UAC, который не даст спокойно работать в Program files. Я перенес в *D:\tmp\easy-rsa*. Переименовываем файл *vars.bat.sample* в *vars.bat*. Открываем его на редактирование и приводим примерно к следующему виду.

```
set KEY_COUNTRY=RU
set KEY_PROVINCE=MSK
set KEY_CITY=Moscow
set KEY_ORG=Mikrotik
set KEY_EMAIL=root@serveradmin.ru
set KEY_CN=changeme
set KEY_NAME=changeme
set KEY_OU=changeme
set PKCS11_MODULE_PATH=changeme
set PKCS11_PIN=1234
```


Для тех, кто не понял, это просто переменные, которые я указал под свои нужды. Там писать можно все, что угодно, не принципиально для нашей задачи. Можно вообще ничего не менять, а оставить как есть. Создаем в директории папку *keys*. Далее запускаем командную строку от администратора и перемещаемся в указанную директорию *D:\tmp\easy-rsa*.



Далее в командной строке пишем vars и жмем enter. Этим мы загрузим переменные из файла vars.bat, потом вводим **clean-all**. Дальше генерируем Root CA командой — **build-ca**.

```
C:\WINDOWS\system32>d:

D:\>cd D:\tmp\easy-rsa

D:\tmp\easy-rsa>vars

D:\tmp\easy-rsa>clean-all
Скопировано файлов:         1.
Скопировано файлов:         1.

D:\tmp\easy-rsa>build-ca
Generating a RSA private key
.......................++++
.............................................................++++
writing new private key to 'keys\ca.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [RU]:
State or Province Name (full name) [MSK]:
Locality Name (eg, city) [Moscow]:
Organization Name (eg, company) [Mikrotik]:
Organizational Unit Name (eg, section) [changeme]:
Common Name (eg, your name or your server's hostname) [changeme]:
Name [changeme]:
Email Address [root@serveradmin.ru]:

D:\tmp\easy-rsa>
```

Отвечаем на задаваемые вопросы и завершаем создание корневого сертификата. Он появится в папке *D:\tmp\easy-rsa\keys*. Дальше создаем сертификат openvpn сервера командой — **build-key-server имя_сервера**.

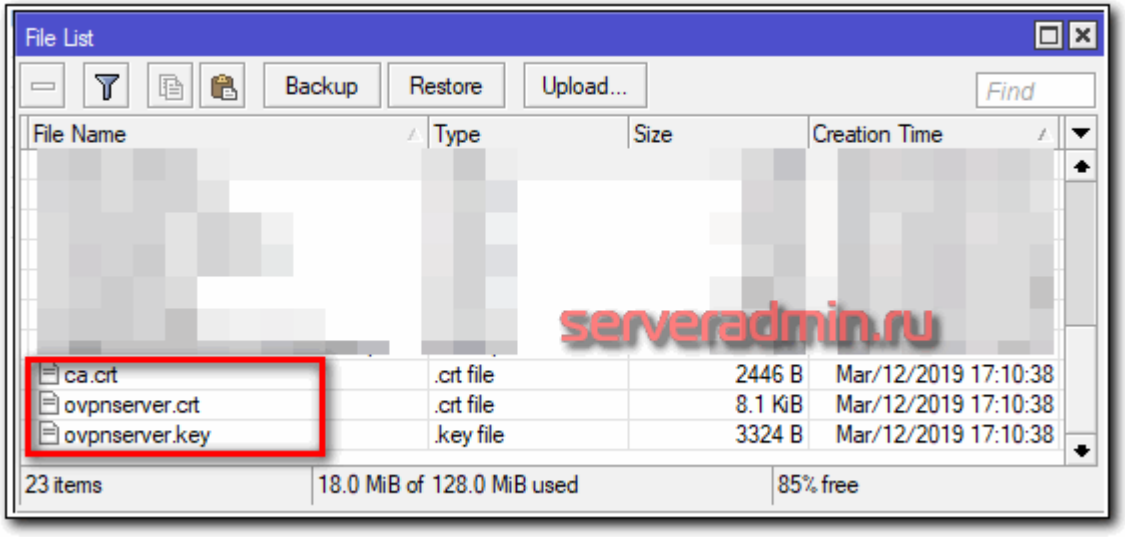
```
D:\tmp\easy-rsa>build-key-server ovpnserver
Generating a RSA private key
.....................................................................................
.....................................................................................
writing new private key to 'keys\ovpnserver.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [RU]:
State or Province Name (full name) [MSK]:
Locality Name (eg, city) [Moscow]:
Organization Name (eg, company) [Mikrotik]:
Organizational Unit Name (eg, section) [changeme]:
Common Name (eg, your name or your server's hostname) [changeme]:opvnserver
Name [changeme]:
Email Address [root@serveradmin.ru]:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
Using configuration from openssl-1.0.0.cnf
Can't open keys/index.txt.attr for reading, No such file or directory
9308:error:02001002:system library:fopen:No such file or directory:crypto/bio/bss_file.c:74:fopen
'r')
9308:error:2006D080:BIO routines:BIO_new_file:no such file:crypto/bio/bss_file.c:81:
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName           :PRINTABLE:'RU'
stateOrProvinceName   :PRINTABLE:'MSK'
localityName          :PRINTABLE:'Moscow'
organizationName      :PRINTABLE:'Mikrotik'
organizationalUnitName:PRINTABLE:'changeme'
commonName            :PRINTABLE:'opvnserver'
name                  :PRINTABLE:'changeme'
emailAddress          :IA5STRING:'root@serveradmin.ru'
Certificate is to be certified until Mar  9 14:02:24 2029 GMT (3650 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
```

Теперь сгенерируем сертификат для клиента. У меня только один клиент в виде удаленного микротика. Вы создаете ровно столько, сколько вам нужно. Используем команду **build-key имя_сертификата**.

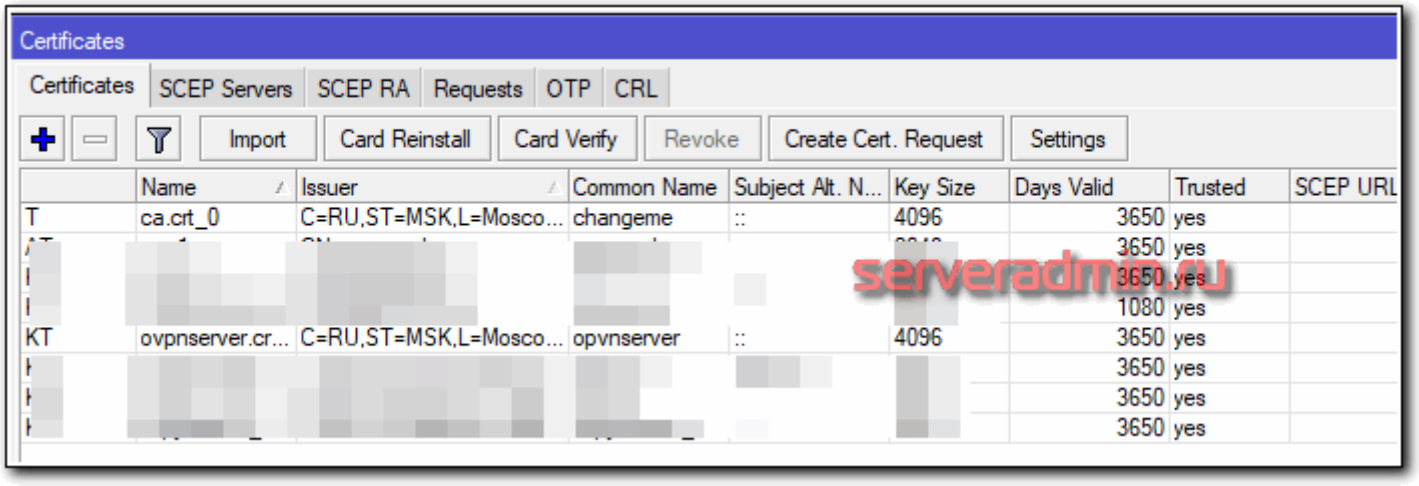
```
D:\tmp\easy-rsa>build-key m-remote
Generating a RSA private key
.....................................................................................
.....................................+++
.....................................................................................
writing new private key to 'keys\m-remote.key'
-----
You are about to be asked to enter information that will be incorporated
into your certificate request.
What you are about to enter is what is called a Distinguished Name or a DN.
There are quite a few fields but you can leave some blank
For some fields there will be a default value,
If you enter '.', the field will be left blank.
-----
Country Name (2 letter code) [RU]:
State or Province Name (full name) [MSK]:
Locality Name (eg, city) [Moscow]:
Organization Name (eg, company) [Mikrotik]:
Organizational Unit Name (eg, section) [changeme]:
Common Name (eg, your name or your server's hostname) [changeme]:m-remote
Name [changeme]:
Email Address [root@serveradmin.ru]:

Please enter the following 'extra' attributes
to be sent with your certificate request
A challenge password []:
An optional company name []:
Using configuration from openssl-1.0.0.cnf
Check that the request matches the signature
Signature ok
The Subject's Distinguished Name is as follows
countryName           :PRINTABLE:'RU'
stateOrProvinceName   :PRINTABLE:'MSK'
localityName          :PRINTABLE:'Moscow'
organizationName      :PRINTABLE:'Mikrotik'
organizationalUnitName:PRINTABLE:'changeme'
commonName            :PRINTABLE:'m-remote'
name                  :PRINTABLE:'changeme'
emailAddress          :IA5STRING:'root@serveradmin.ru'
Certificate is to be certified until Mar  9 14:04:33 2029 GMT (3650 days)
Sign the certificate? [y/n]:y


1 out of 1 certificate requests certified, commit? [y/n]y
Write out database with 1 new entries
Data Base Updated

D:\tmp\easy-rsa>
```

С созданием сертификатов закончили. Они у нас все лежат в директории keys. На микротик, который будет выступать в качестве openvpn сервера, нужно передать файлы:

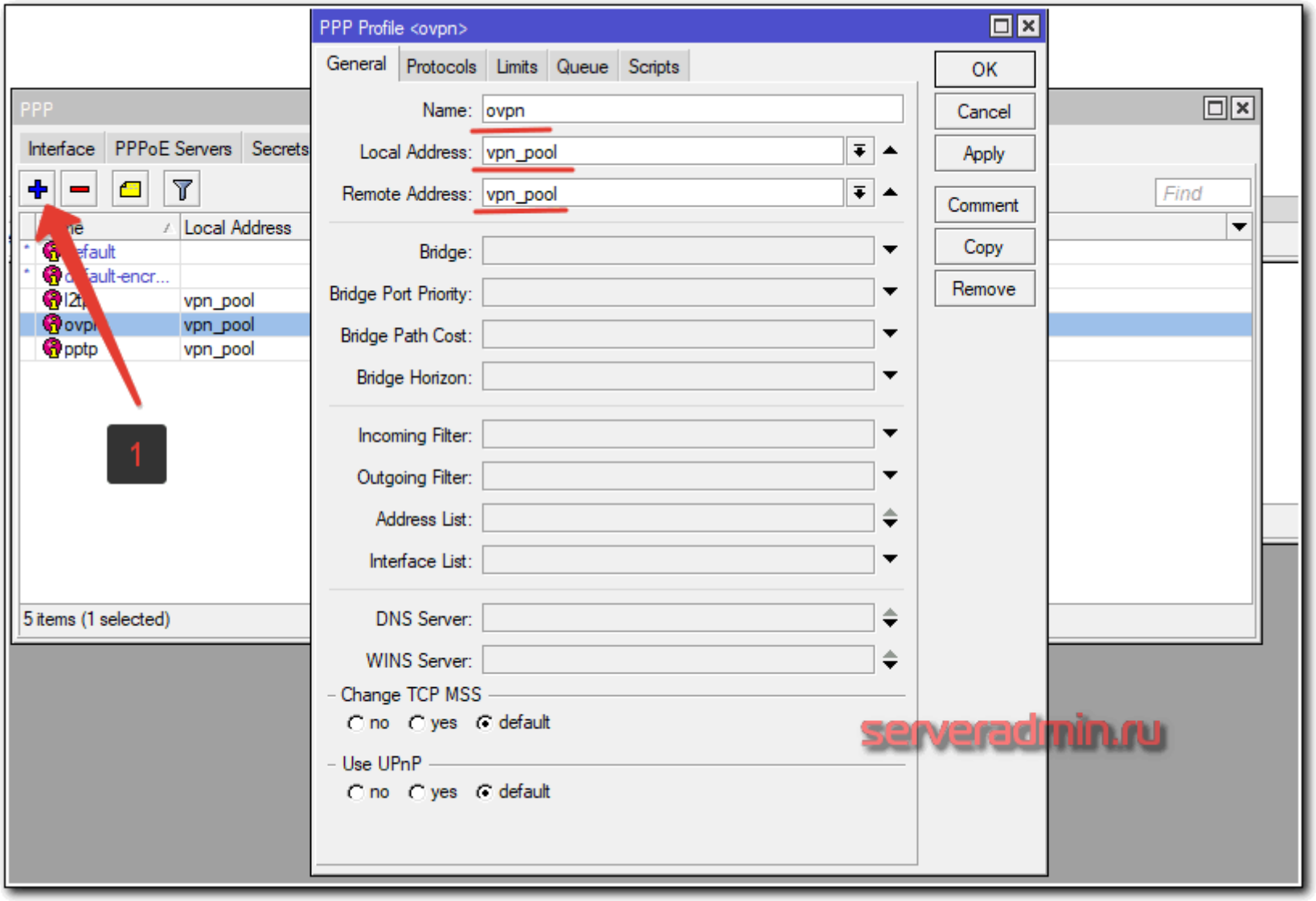
- ca.crt
- ovpnserver.crt
- ovpnserver.key



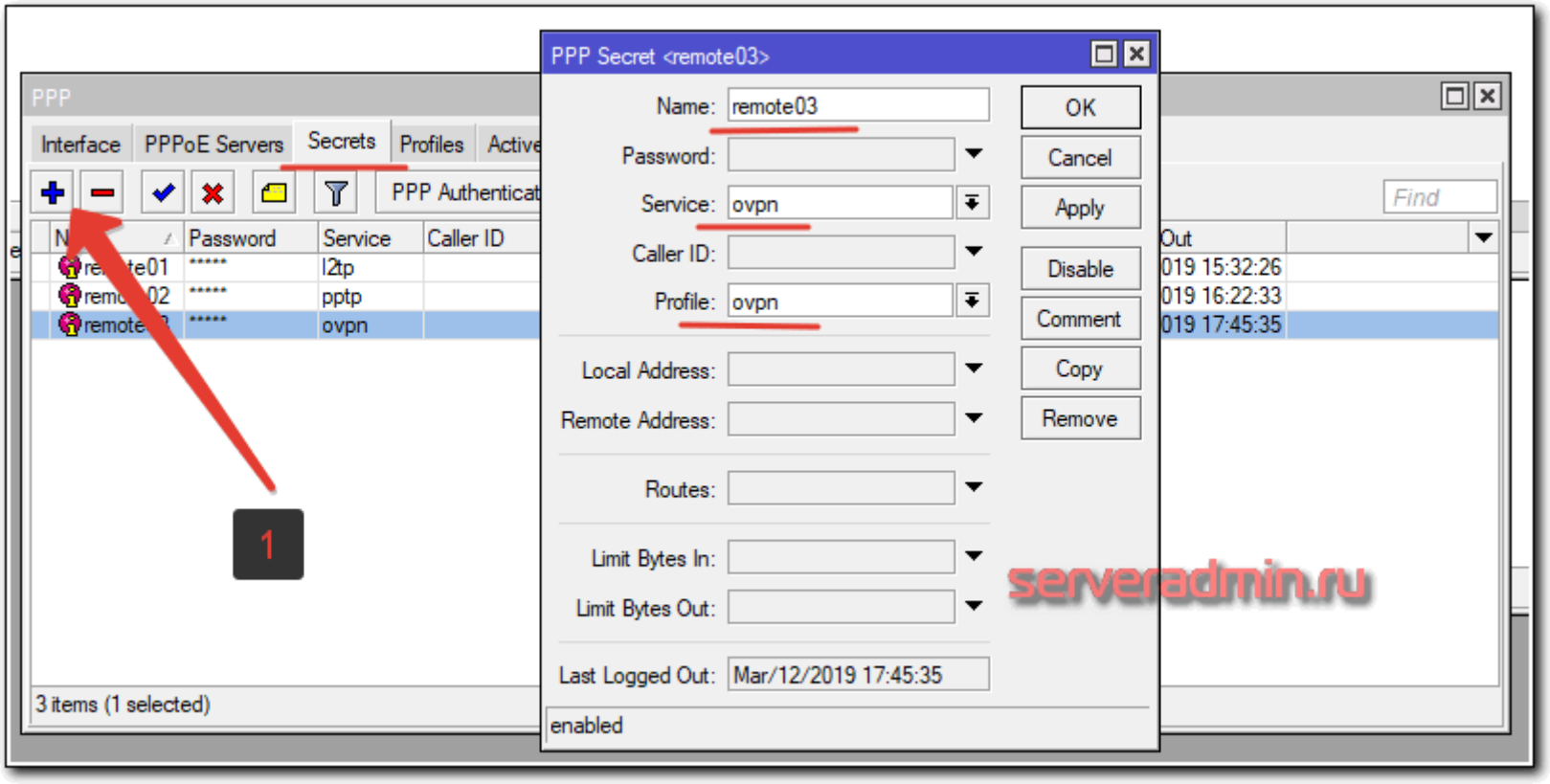
Импортируем сертификаты из добавленных файлов. Идем в **System -> Certificates** и импортируем сначала *ca.crt*, потом *ovpnserver.crt* и *ovpnserver.key*.



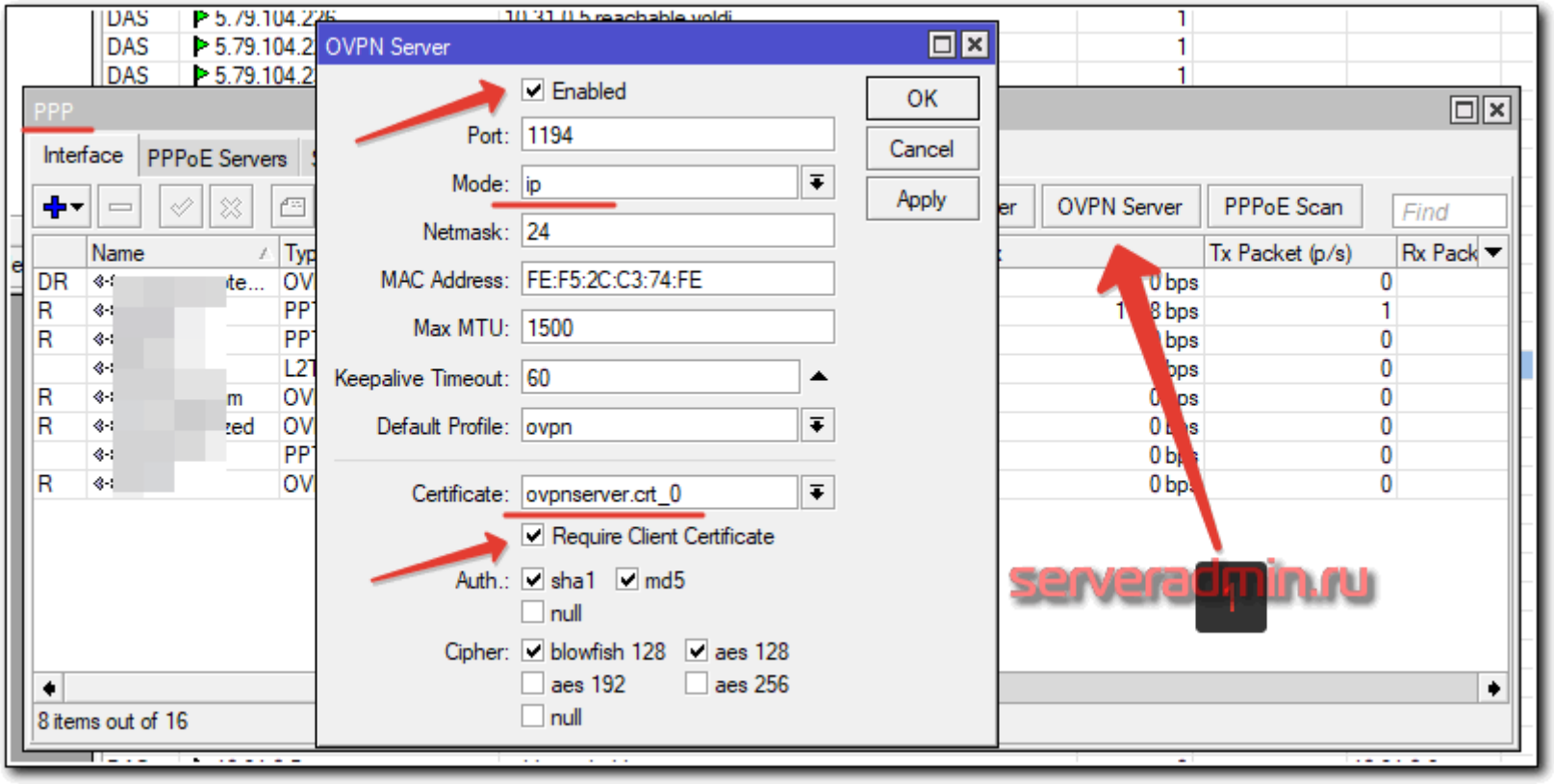
Должно получиться примерно так. Теперь приступаем к настройке openvpn сервера в mikrotik. Создадим для него отдельный профиль в **PPP -> Profiles**.


PPP
Interface
PPPoE Servers
Secrets
Local Address
vpn_pool
vpn_pool
vpn_pool
5 items (1 selected)
1
PPP Profile <ovpn>
General
Protocols
Limits
Queue
Scripts
OK
Cancel
Apply
Comment
Copy
Remove
Name: ovpn
Local Address: vpn_pool
Remote Address: vpn_pool
Bridge:
Bridge Port Priority:
Bridge Path Cost:
Bridge Horizon:
Incoming Filter:
Outgoing Filter:
Address List:
Interface List:
DNS Server:
WINS Server:
Change TCP MSS
no
yes
default
Use UPnP
no
yes
default
serveradmin.ru

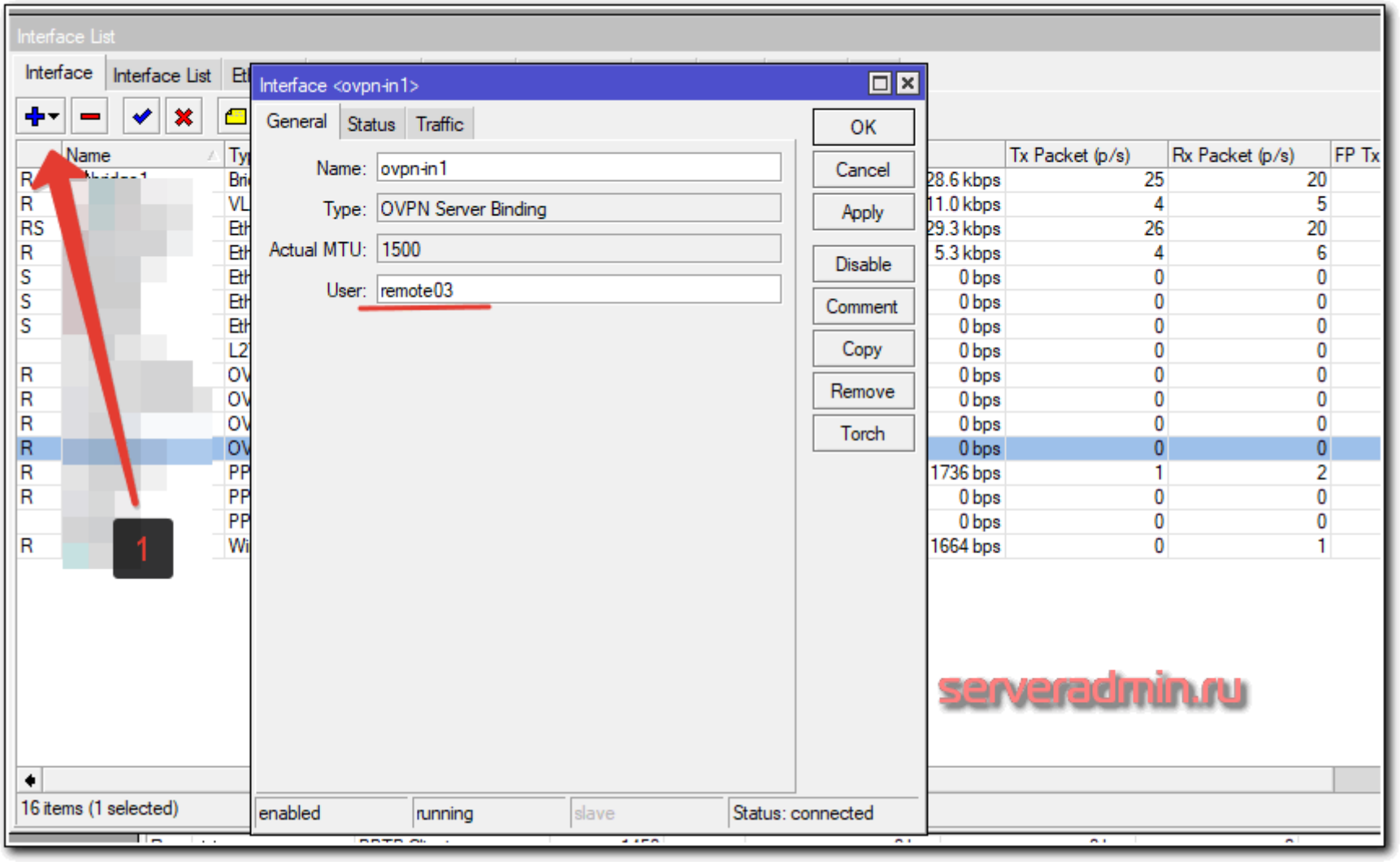
Все настройки дефолтные. В качестве локального и удаленного адреса использую Ip Pool, который создал в самом начале настройки l2tp. Добавим удаленного пользователя для openvpn в **PPP ->Secrets**.

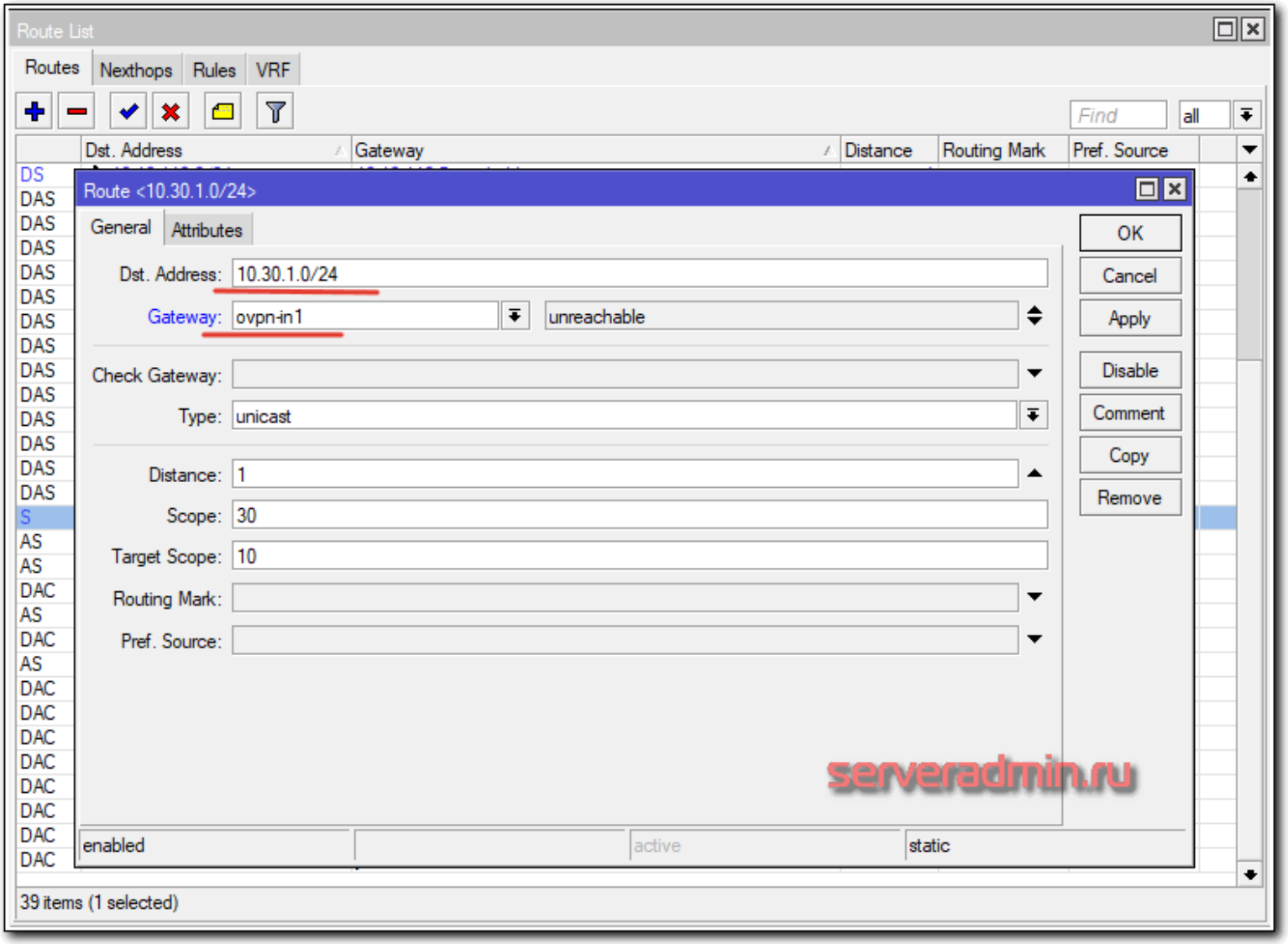


Идем в раздел **PPP** и жмем **OVPN Server**. Указываем настройки и загруженный ca сертификат.



Далее добавляем по аналогии с остальными vpn серверами **OVPN Server Binding** и статические маршруты.


Interface List
Interface
Interface List
Interface <ovpn-in1>
General
Status
Traffic
Name: ovpn-in1
Type: OVPN Server Binding
Actual MTU: 1500
User: remote03
OK
Cancel
Apply
Disable
Comment
Copy
Remove
Torch
Name
Tx Packet (p/s)
Rx Packet (p/s)
FP Tx
1
16 items (1 selected)
enabled
running
slave
Status: connected
serveradmin.ru


Route List
Routes
Nexthops
Rules
VRF
Find
all
Dst. Address
Gateway
Distance
Routing Mark
Pref. Source
Route <10.30.1.0/24>
General
Attributes
OK
Cancel
Apply
Disable
Comment
Copy
Remove
Dst. Address: 10.30.1.0/24
Gateway: ovpn-in1
unreachable
Check Gateway:
Type: unicast
Distance: 1
Scope: 30
Target Scope: 10
Routing Mark:
Pref. Source:
serveradmin.ru
enabled
active
static
39 items (1 selected)

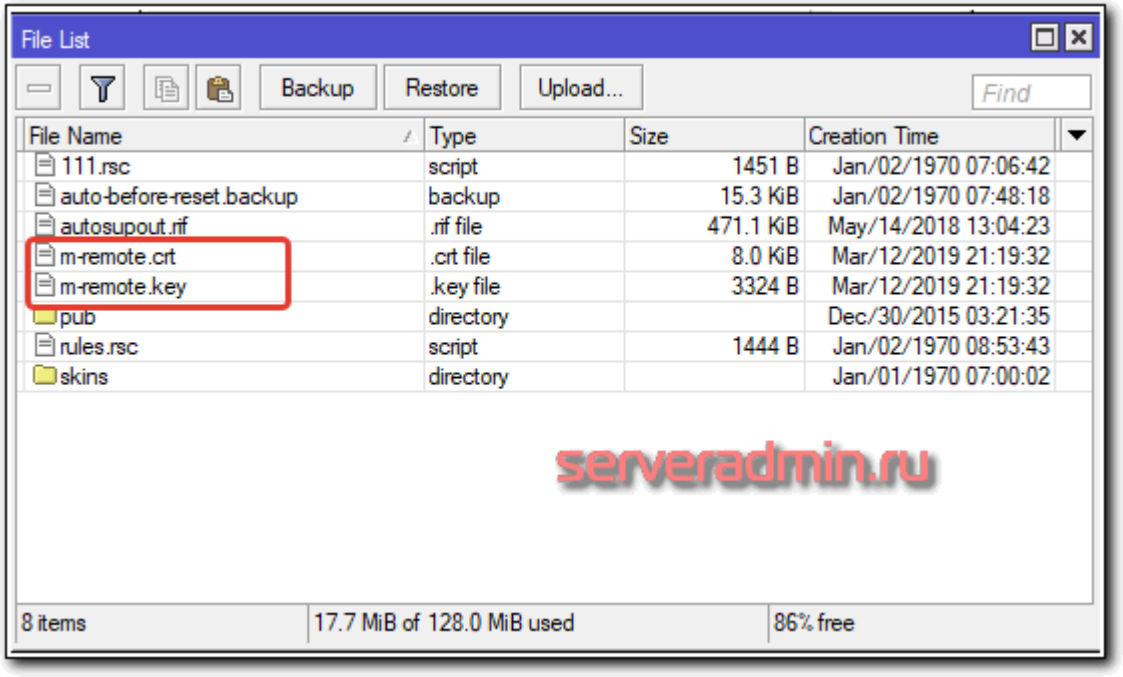
На этом настройка openvpn server в микротик завершена. По дефолту будет использоваться протокол шифрования **BF-128-CBC**. Его можно поменять в свойствах клиента, а список всех поддерживаемых шифров в свойствах vpn сервера.

Для работы указанной настройки openvpn сервера необходимо открыть входящий tcp порт 1194 на фаерволе. Теперь настроим openvpn клиент и протестируем скорость соединения через vpn на основе openvpn.

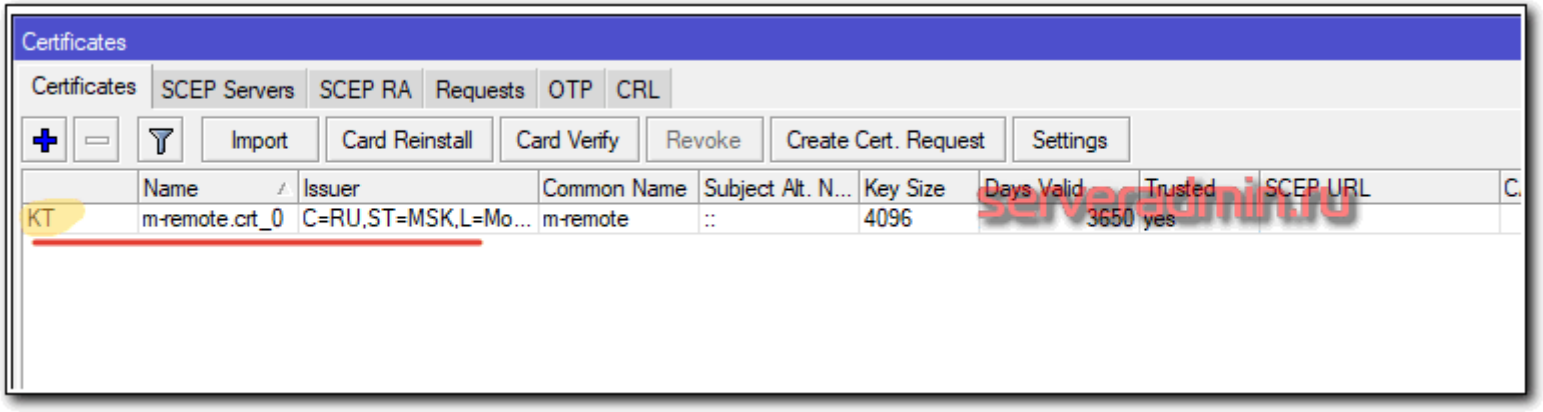
openvpn client

Для настройки openvpn client на mikrotik, туда нужна передать сертификаты, сгенерированные на предыдущем шаге. Конкретно вот эти файлы:

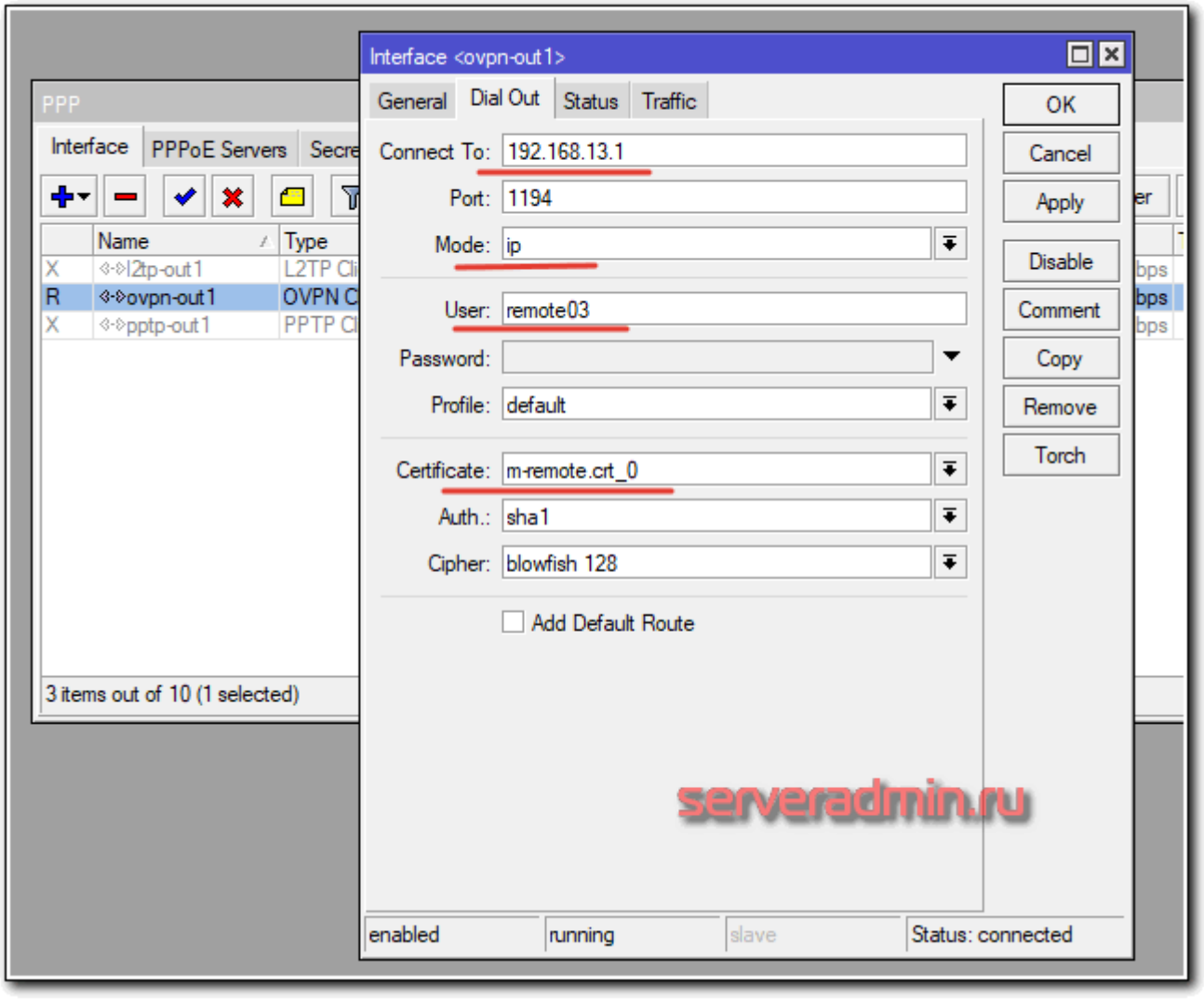
- m-remote.crt
- m-remote.key



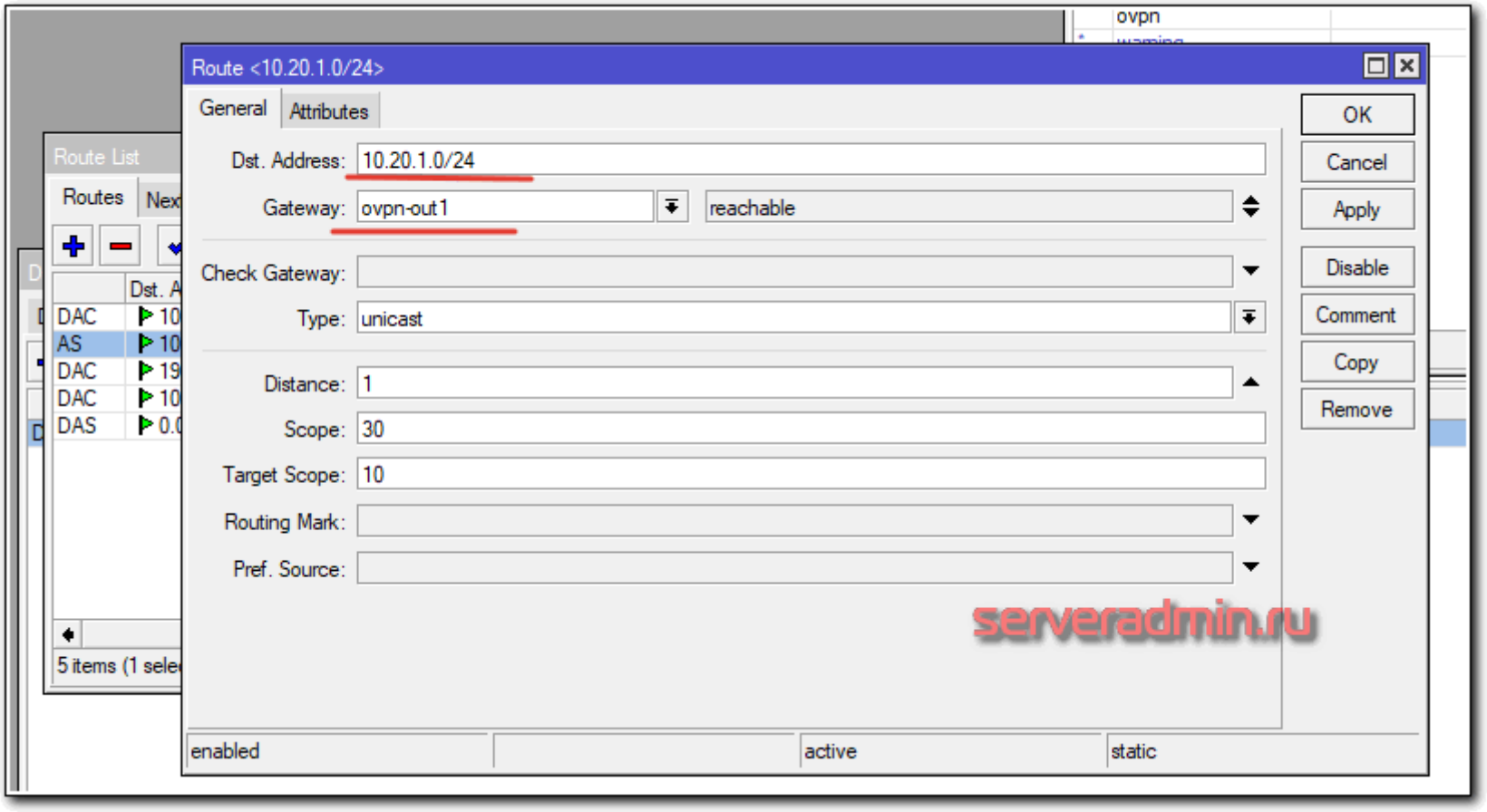
Импортируем, как и на сервере сертификат из этих файлов. Обращаю внимание, что должны быть символы KT напротив имени сертификата.



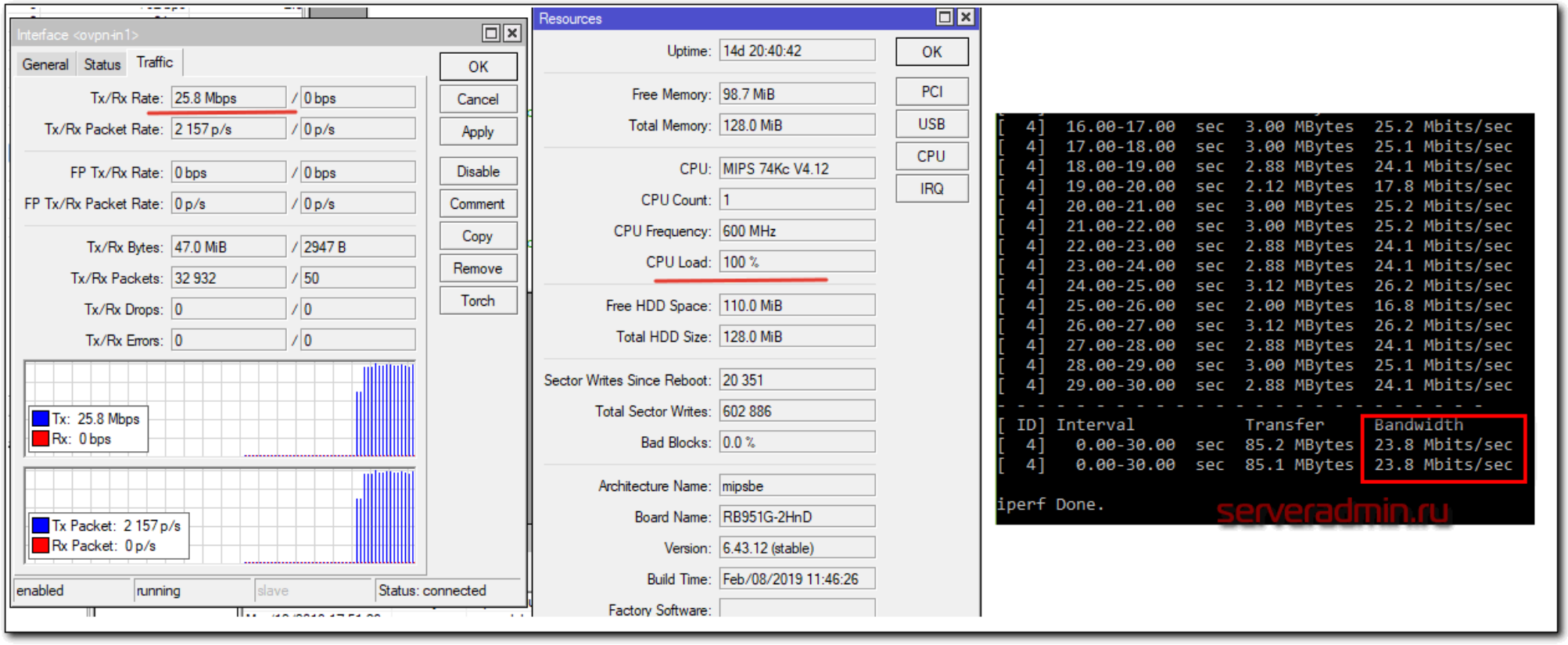
Теперь настраивает openvpn клиента. Идем в **PPP** и добавляем **OVPN Client**.


PPP
Interface
PPPoE Servers
Secre
Name
Type
X
l2tp-out1
L2TP Cli
R
ovpn-out1
OVPN C
X
pptp-out1
PPTP Cl
3 items out of 10 (1 selected)
Interface <ovpn-out1>
General
Dial Out
Status
Traffic
Connect To: 192.168.13.1
Port: 1194
Mode: ip
User: remote03
Password:
Profile: default
Certificate: m-remote.crt_0
Auth.: sha1
Cipher: blowfish 128
Add Default Route
OK
Cancel
Apply
Disable
Comment
Copy
Remove
Torch
serveradmin.ru
enabled
running
slave
Status: connected

Добавляем статический маршрут для доступа к ресурсам удаленной сети за openvpn сервером.

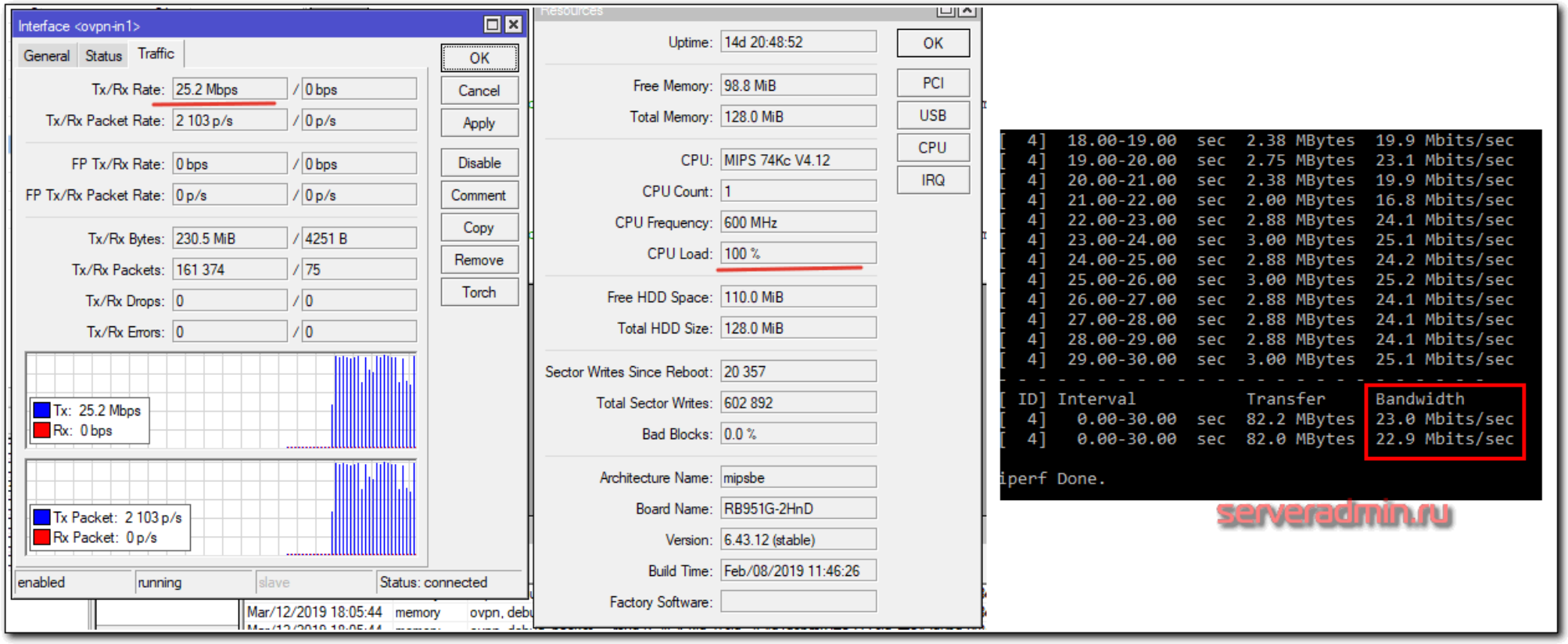


Все готово. Можно подключаться и тестировать скорость vpn соединения через openvpn.



Получилось в среднем **24 мбит/сек** при 100% загрузке процессора. Результат сопоставим с l2tp + ipsec. Немного удивил результат. Я думал, будет хуже, чем l2tp, а на деле то же самое. Мне лично вариант с openvpn в целом нравится больше, хотя из-за ограниченности настроек openvpn в микротике преимущества openvpn трудно реализовать. Напомню, что тестировал с шифрованием BF-128-CBC, то есть blowfish.

Вот результат с AES-128-CBC — 23 мбит/сек, примерно то же самое.

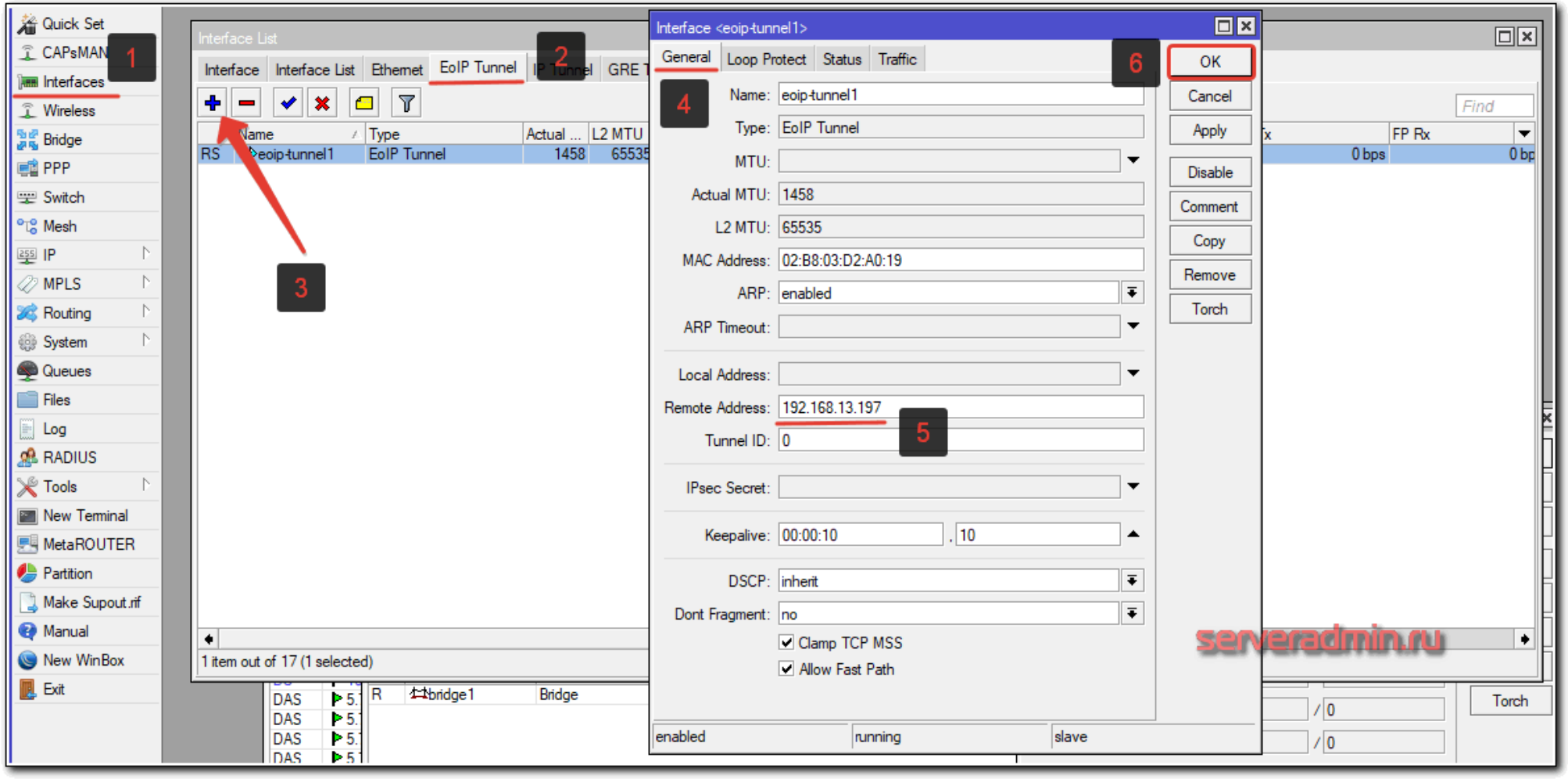


С клиент-серверными реализациями vpn сервера в mikrotik разобрались. Теперь просмотрим на скорость l2-vpn в виде eoip tunnel.

Настройка EOIP Tunnel + Ipsec

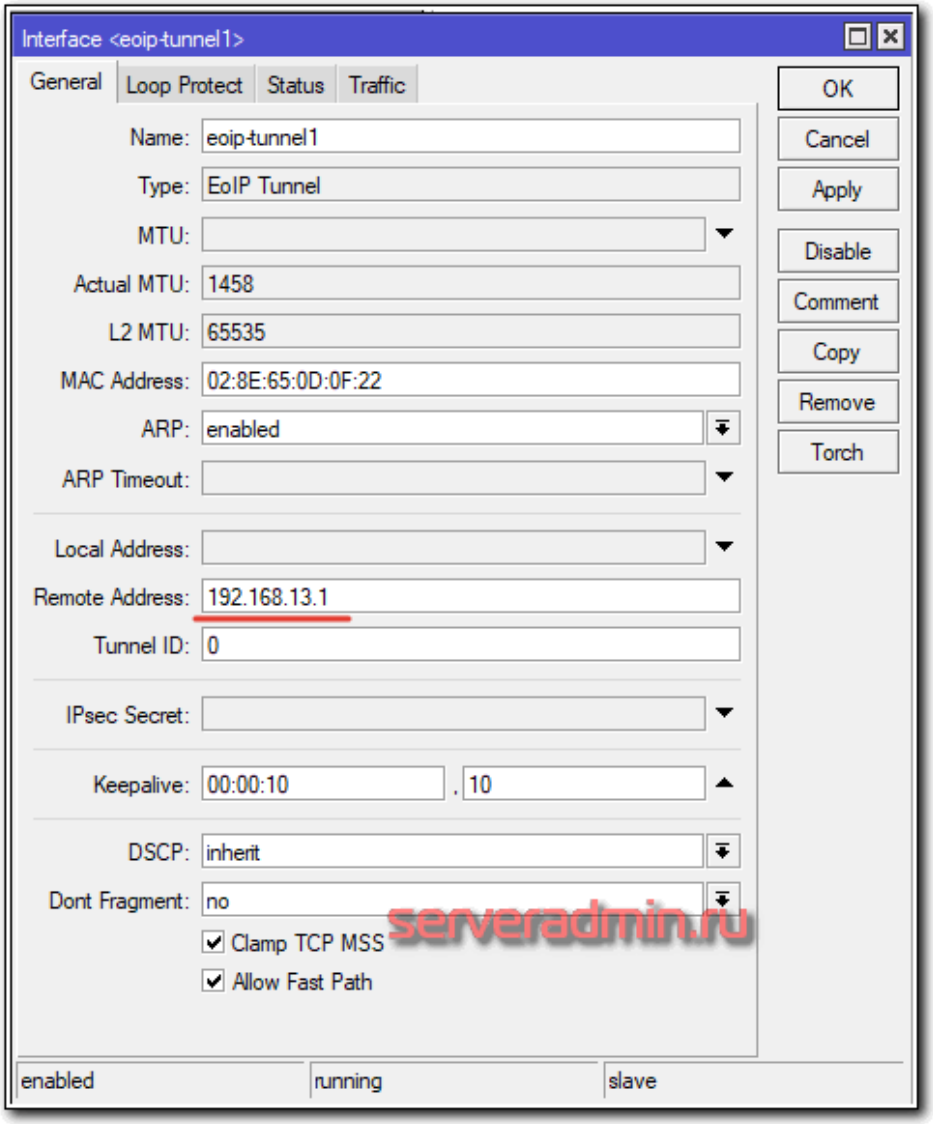
Настроим vpn сеть на базе EOIP в Mikrotik. Тут нужно понимать одно важное отличие от всех предыдущих настроек, которые мы делали ранее. EOIP туннель работает на уровне l2, то есть оба сегмента сети будут считать, что находятся в одной физической сети. Адресное пространство для обоих будет одно и то же. В моем примере это 10.20.1.0/24. DHCP сервер должен остаться только один для обоих сетей. В моем случае он останется на **m-server**.

Создаем EOIP туннель на m-server. Идем в **Interface list -> EoIP Tunnel** и добавляем новый.

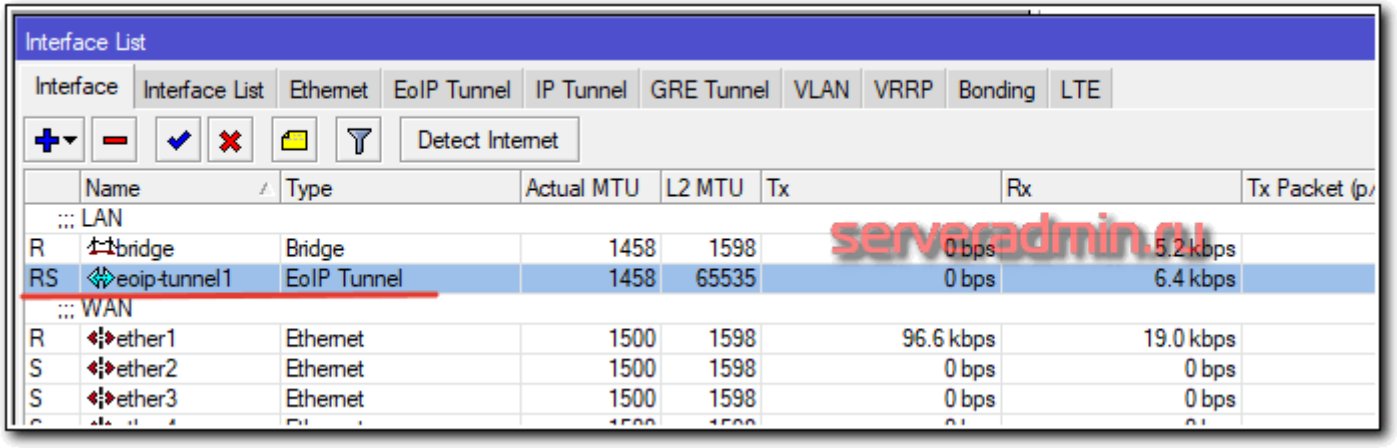


Из настроек достаточно указать только удаленный адрес второго микротика. Новый EoIP интерфейс необходимо добавить в локальный бридж вместе с физическими интерфейсами.

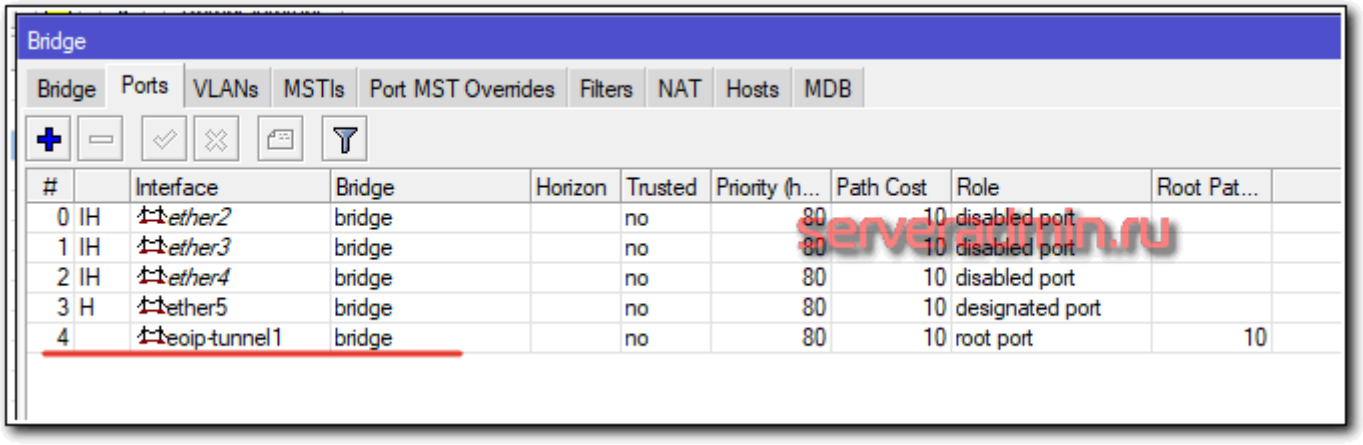
Идем на удаленный микротик и там делаем все то же самое, только Remote Address указываем другой.


Interface <eoip-tunnel1>
General
Loop Protect
Status
Traffic
OK
Cancel
Apply
Disable
Comment
Copy
Remove
Torch
Name: eoip-tunnel1
Type: EoIP Tunnel
MTU:
Actual MTU: 1458
L2 MTU: 65535
MAC Address: 02:8E:65:0D:0F:22
ARP: enabled
ARP Timeout:
Local Address:
Remote Address: 192.168.13.1
Tunnel ID: 0
IPsec Secret:
Keepalive: 00:00:10 , 10
DSCP: inherit
Dont Fragment: no
Clamp TCP MSS
Allow Fast Path
serveradmin.ru
enabled
running
slave

Этого достаточно, чтобы EoIP туннель сразу же заработал. Его состояние будет RS.

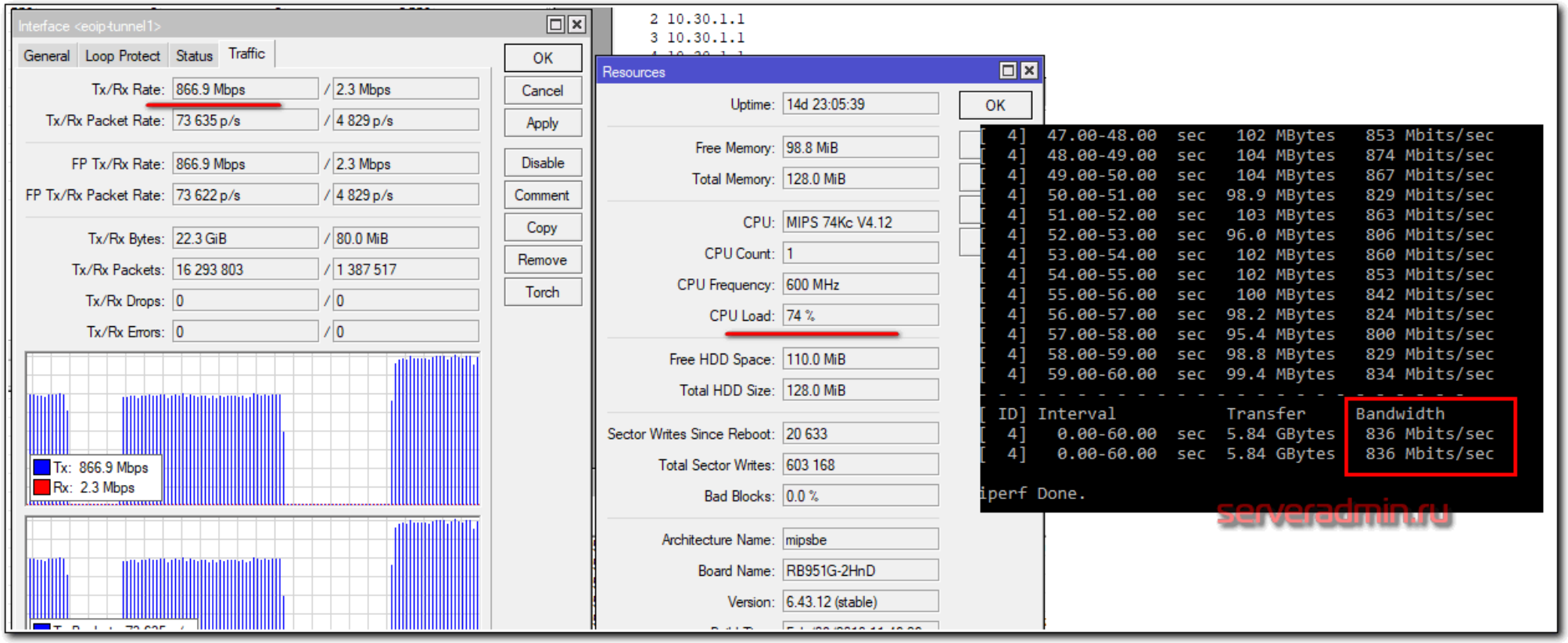


На втором микротике EoIP интерфейс так же нужно добавить в локальный бридж с остальными интерфейсами.



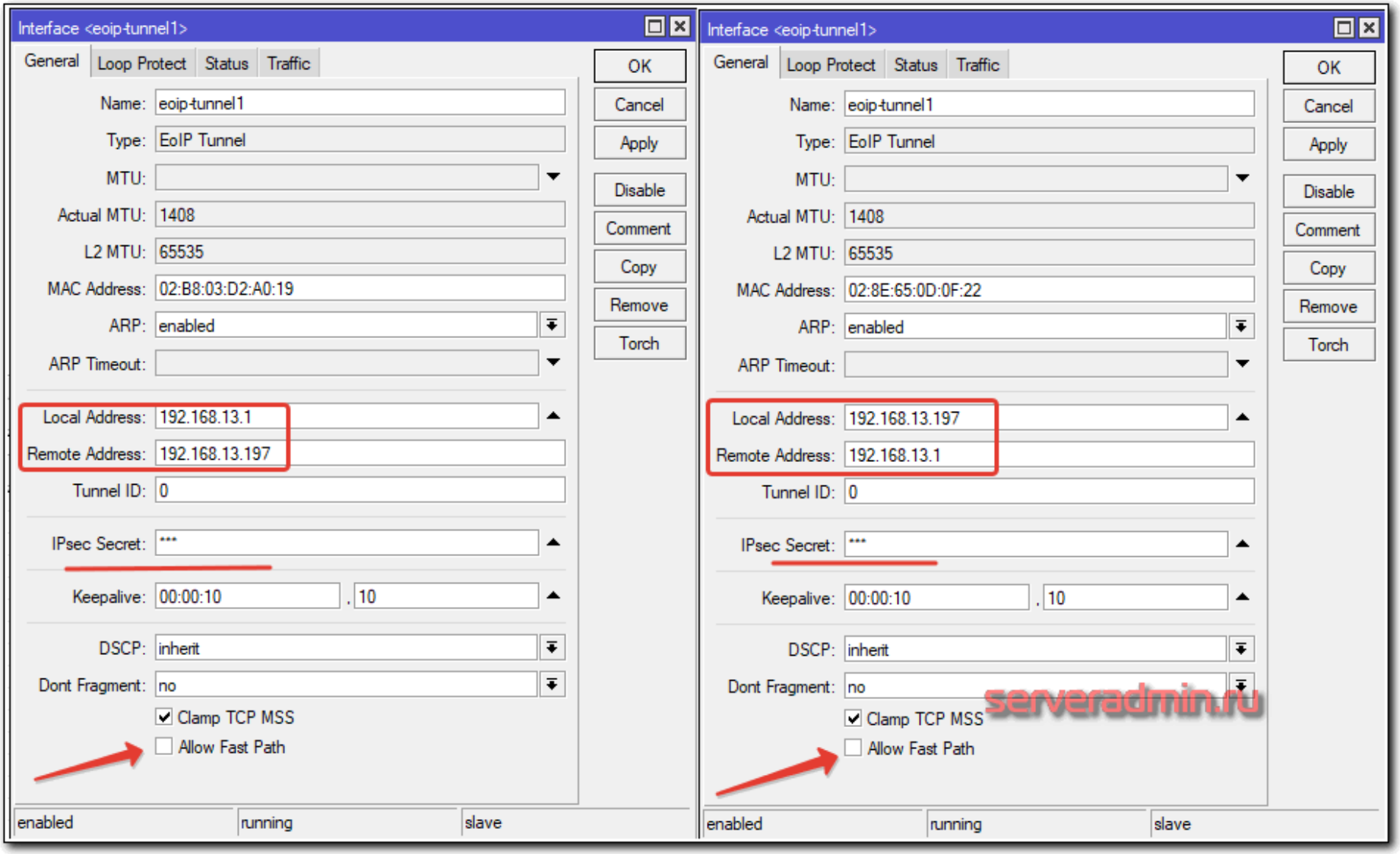
Проще всего проверить, что все в порядке, это запросить по dhcp на m-slave ip адрес для интерфейса bridge. Он должен получить ip адрес от dhcp сервера на m-server, при условии, что в сети больше нет других dhcp серверов. То же самое будет и с локальными машинами в сети за m-slave. Они будут получать ip адреса от dhcp сервера на m-server.

Проверим теперь быстродействие такого vpn туннеля на основе EoIP.

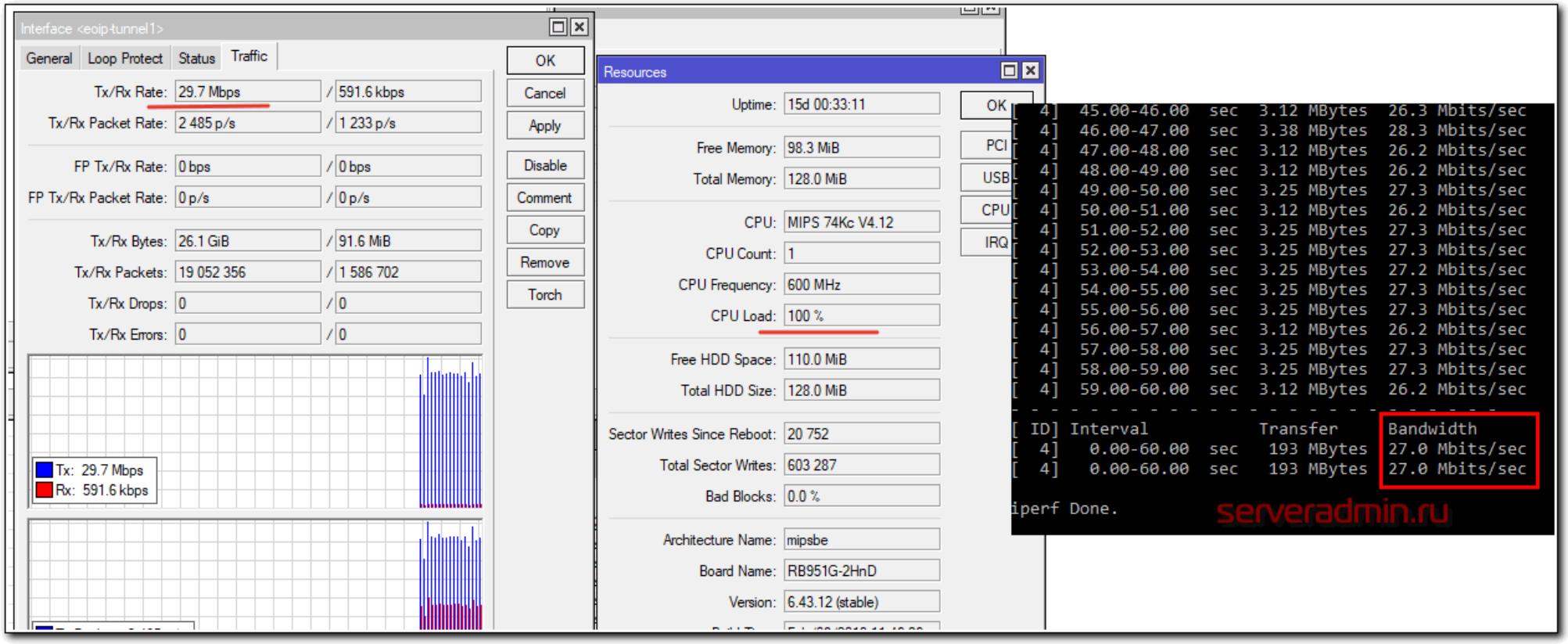


Показываю максимальный результат, который у меня получился — **836 мбит/сек**. По какой-то причине в разных тестах скорость плавала в интервале между 600-850 мбит/сек. Для того, чтобы скорость изменилась, необходимо было отключить и заново включить EoIP интерфейс. Скорость впечатляет. При этом, процессор не загружен на 100%. То есть узкое место не он. Похоже я уперся в производительность сети. Напомню, что тут нет никакого шифрования и маршрутизации трафика. Прямой l2 канал между двумя микротиками через EoIP vpn.

Добавим в EoIP туннель шифрование Ipsec и посмотрим на скорость. Для этого меняем настройки каналов на обоих микротиках. Добавляем пароль Ipsec и локальные адреса, отключаем Fast Path.


Interface <eoip-tunnel1>
General
Loop Protect
Status
Traffic
OK
Cancel
Apply
Disable
Comment
Copy
Remove
Torch
Name: eoip-tunnel1
Type: EoIP Tunnel
MTU:
Actual MTU: 1408
L2 MTU: 65535
MAC Address: 02:B8:03:D2:A0:19
ARP: enabled
ARP Timeout:
Local Address: 192.168.13.1
Remote Address: 192.168.13.197
Tunnel ID: 0
IPsec Secret: ***
Keepalive: 00:00:10 , 10
DSCP: inherit
Dont Fragment: no
Clamp TCP MSS
Allow Fast Path
enabled
running
slave
Interface <eoip-tunnel1>
General
Loop Protect
Status
Traffic
OK
Cancel
Apply
Disable
Comment
Copy
Remove
Torch
Name: eoip-tunnel1
Type: EoIP Tunnel
MTU:
Actual MTU: 1408
L2 MTU: 65535
MAC Address: 02:8E:65:0D:0F:22
ARP: enabled
ARP Timeout:
Local Address: 192.168.13.197
Remote Address: 192.168.13.1
Tunnel ID: 0
IPsec Secret: ***
Keepalive: 00:00:10 , 10
DSCP: inherit
Dont Fragment: no
Clamp TCP MSS
Allow Fast Path
serveradmin.ru
enabled
running
slave

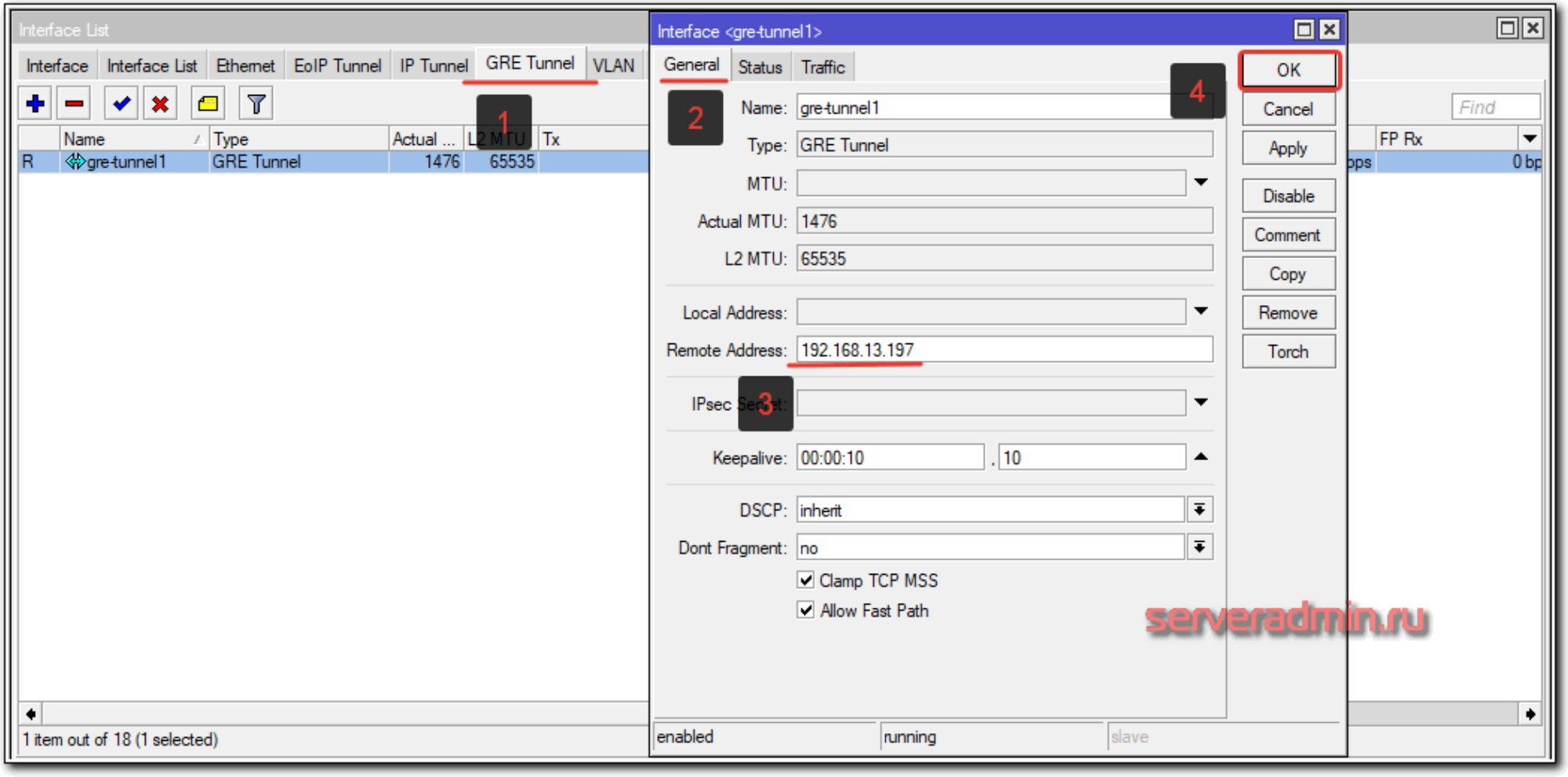
Измеряем скорость соединения.



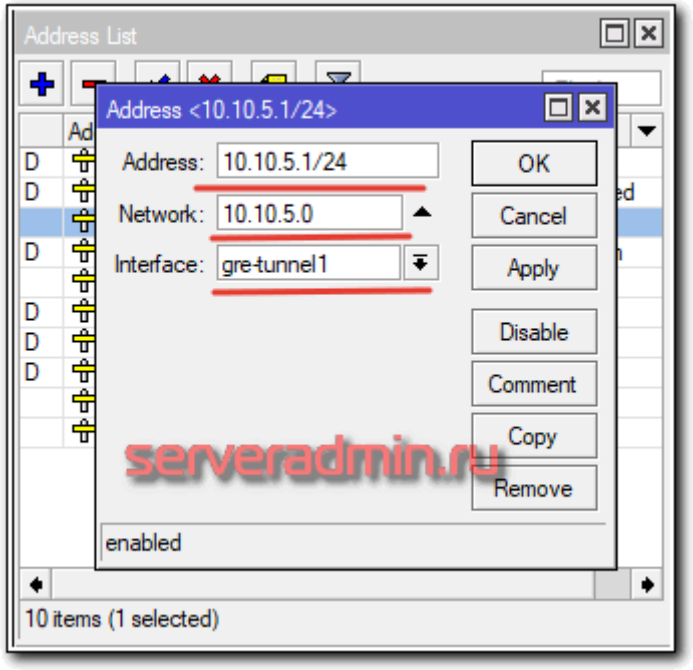
У меня получилась скорость vpn при использовании EoIP + Ipsec в среднем **27 мбит/сек**. Скорость сопоставима с шифрованными туннелями L2tp и Openvpn. В этом плане никаких приятных сюрпризов не получилось. Шифрование очень тяжело дается этой железке. Можно сказать она для него не предназначена практически совсем.

GRE туннель + Ipsec в mikrotik, создание и настройка

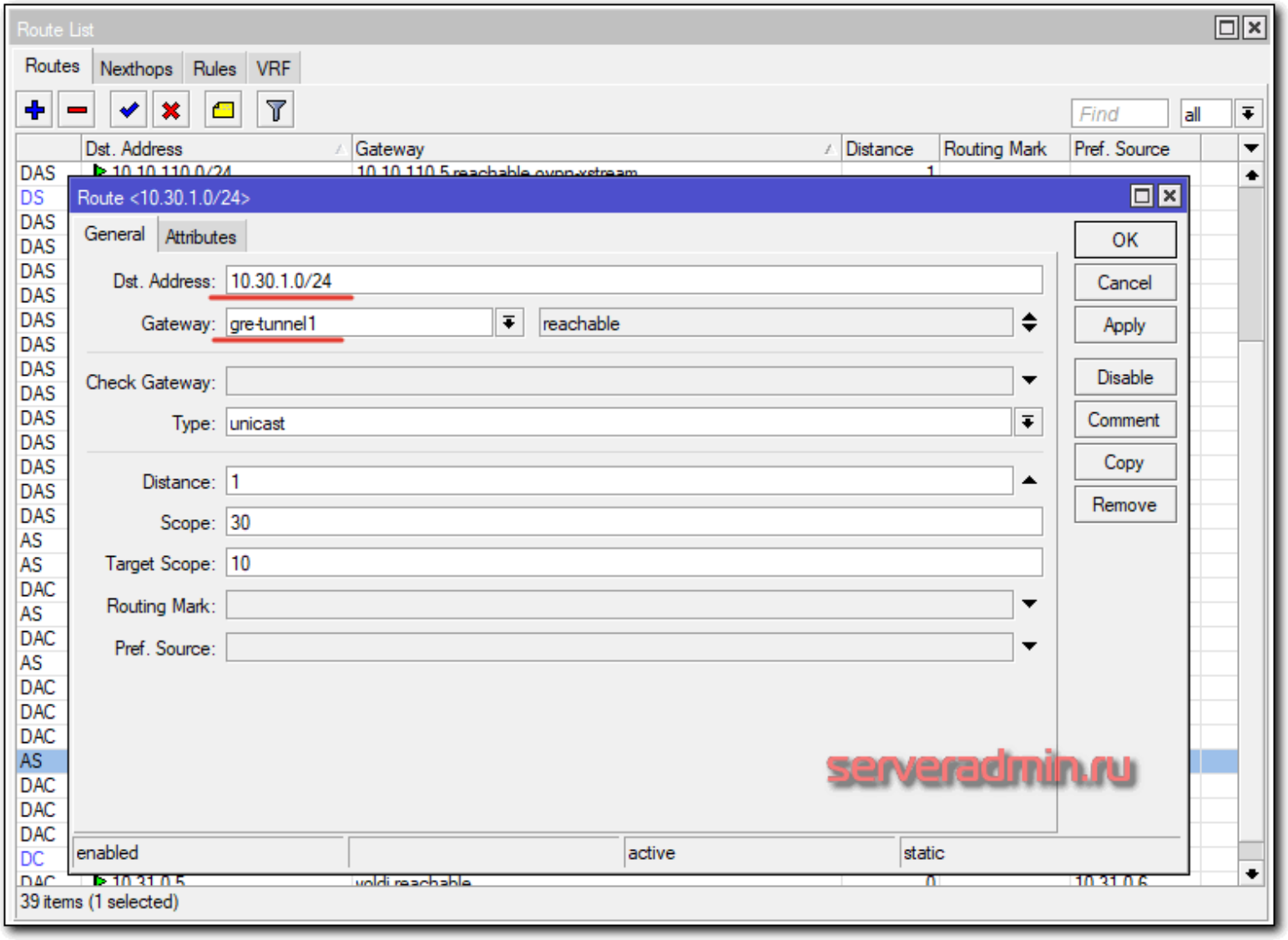
Для настройки GRE туннеля в Mikrotik идем в раздел **Interfaces -> GRE Tunnel** и добавляем новый со следующими настройками:



Назначаем GRE туннелю ip адрес в **IP -> Adresses**.

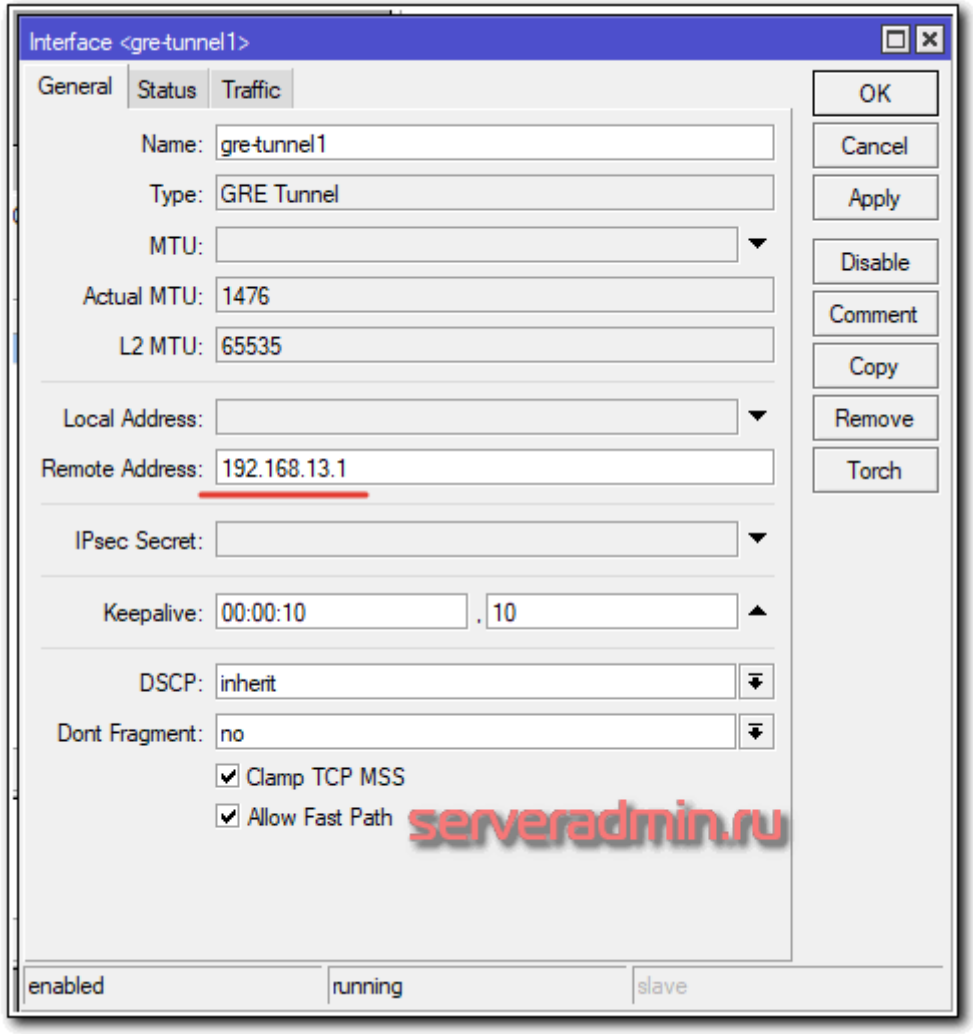


Сразу же создаем статический маршрут для доступа к ресурсам удаленной сети.

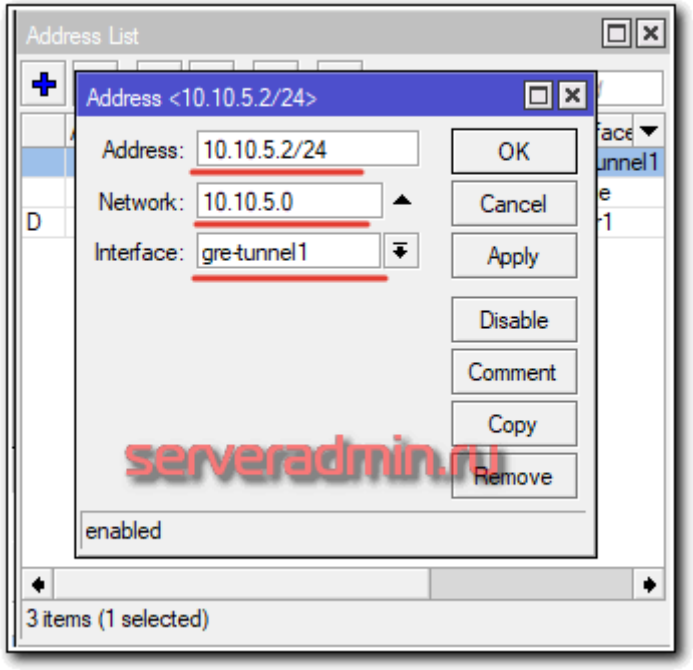

Route List
Routes
Nexthops
Rules
VRF
Find
all
Dst. Address
Gateway
Distance
Routing Mark
Pref. Source
Route <10.30.1.0/24>
General
Attributes
OK
Cancel
Apply
Disable
Comment
Copy
Remove
Dst. Address: 10.30.1.0/24
Gateway: gre-tunnel1
reachable
Check Gateway:
Type: unicast
Distance: 1
Scope: 30
Target Scope: 10
Routing Mark:
Pref. Source:
serveradmin.ru
enabled
active
static
39 items (1 selected)

Для организации vpn соединения через GRE tunnel то же самое проделываем на удаленном микротике, только меняем соответствующие адреса.

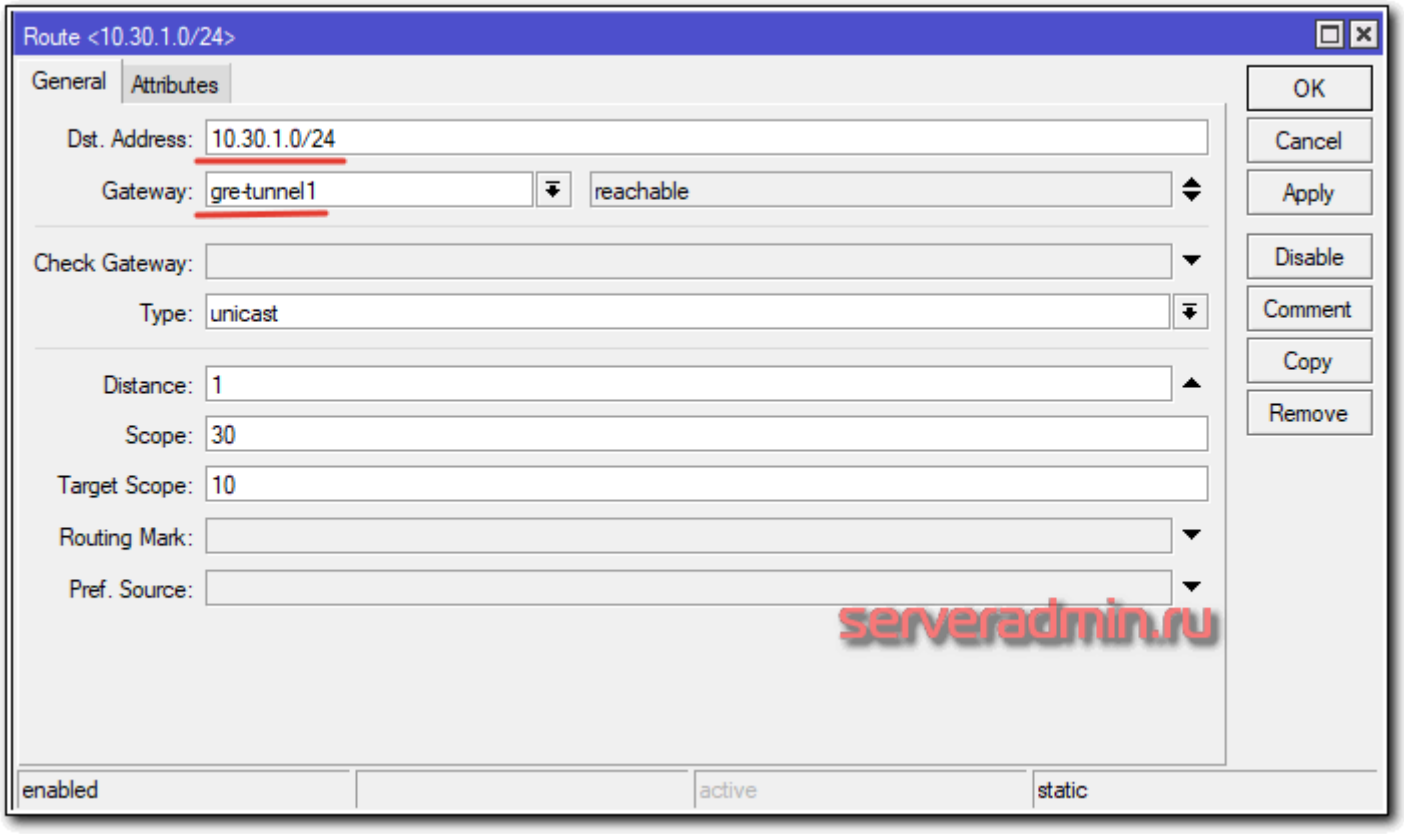
Создаем GRE Tunnel.



Назначаем ip адрес.

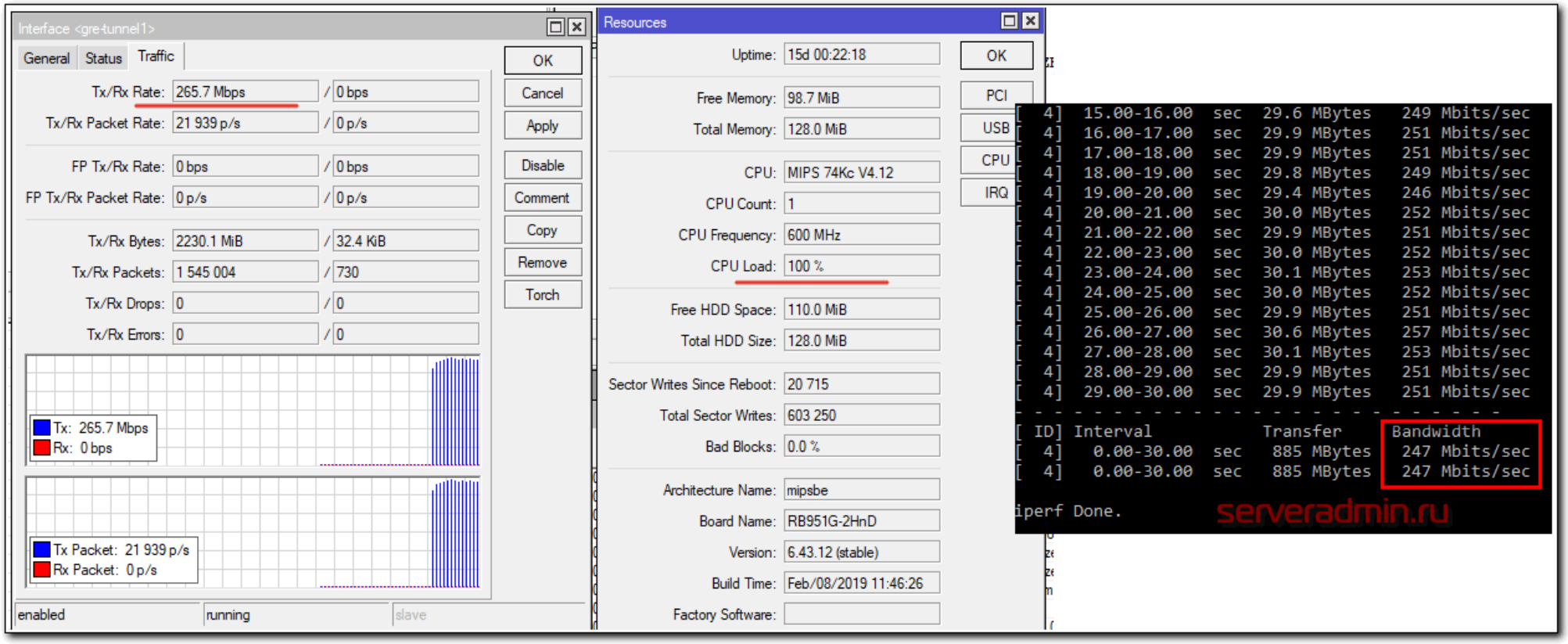


Добавляем маршрут в удаленную локальную сеть.



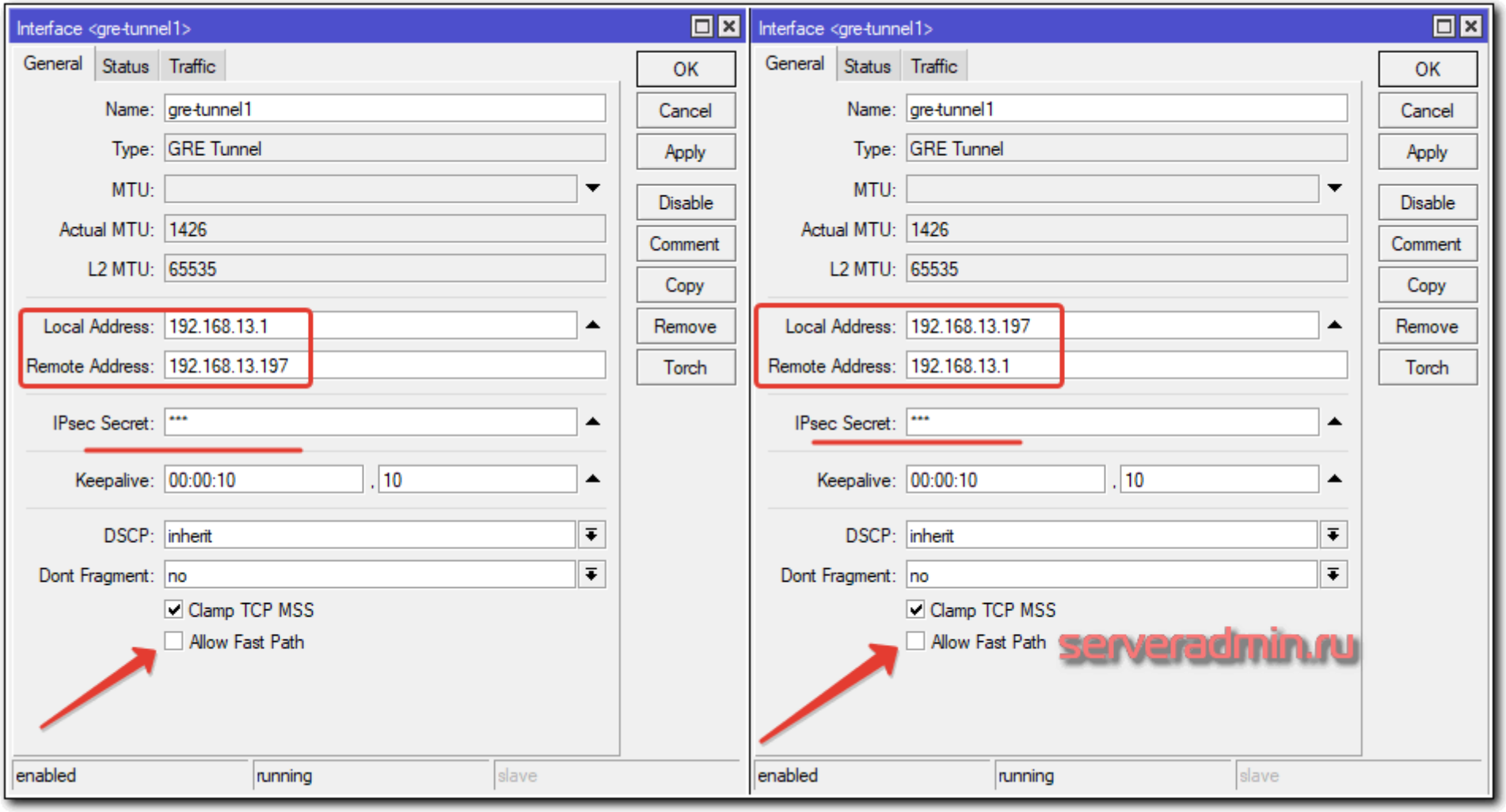
После этого маршрутизация трафика между локальными сетями должна заработать. Не забудьте на firewall разрешить gre протокол.

Проверим теперь скорость соединения по GRE туннелю.

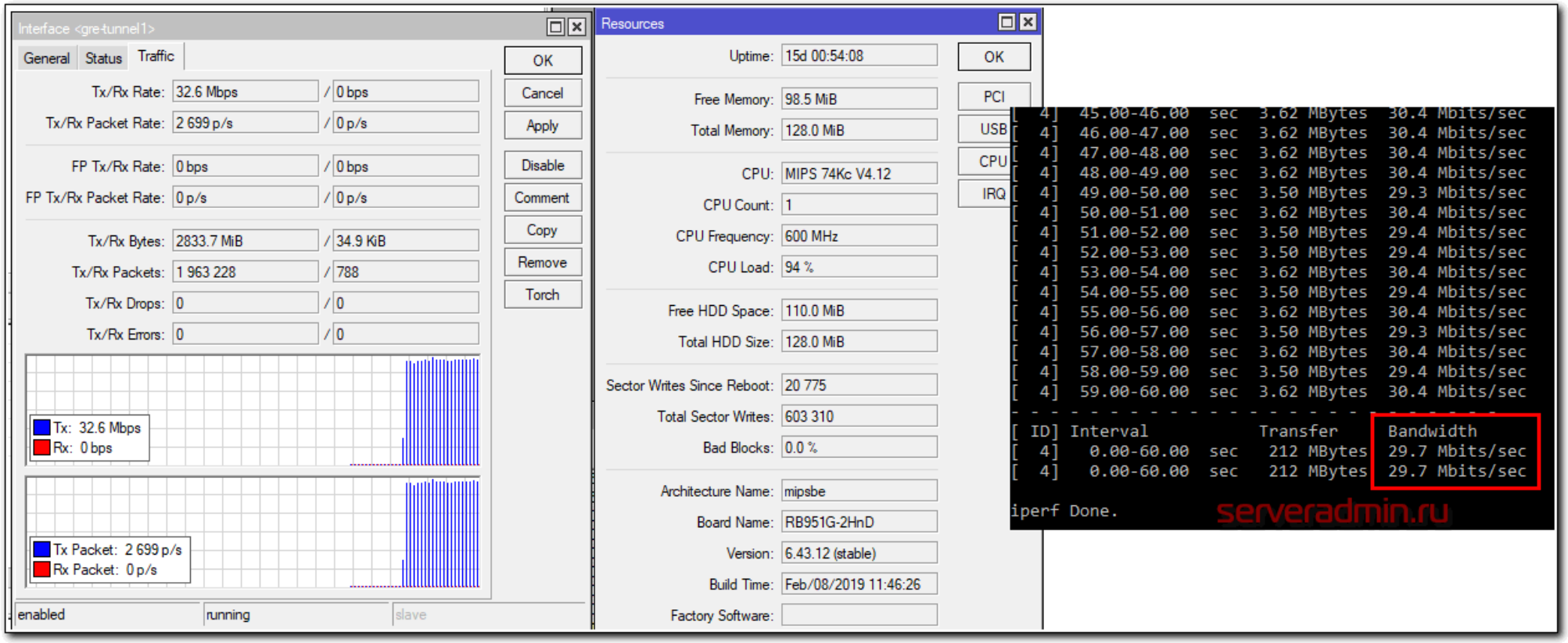


У меня получилось **247 мбит/сек**. Напомню, что это нешифрованный маршрутизируемый vpn туннель. Отличие от l2 туннеля EoIP примерно в 3 раза по скорости в меньшую сторону. Выводы делайте сами какие туннели использовать. Если не нужна маршрутизация, то однозначно EoIP.

Теперь проверим то же самое, только настроив в GRE шифрование Ipsec. Добавляем соответствующие настройки в GRE туннели на обоих микротиках.



Измеряю скорость GRE + Ipsec, алгоритм шифрования aes-128 cbc.



Интересные записи:
Шутки для сисадминов
Мои программы для системного администрирования
Монетизация ИТ блога, сколько можно заработать на информационном сайте

Получилось в среднем **29,7 мбит/сек**, что примерно соответствует всем результатам с ipsec. Не удивительно, ведь алгоритм шифрования во всех случаях один и тот же. Но тем не менее, в GRE Tunnel скорость немного выше всех остальных участников. Из этого можно сделать вывод, что исключительно для l3 site-to-site подключений GRE Tunnel подходит в плане быстродействия лучше всего.

Сравнение скорости L2tp, Pptp, EoIP, GRE и OpenVPN туннелей

Сведу все данные измерений в единую таблицу для наглядного и удобного анализа и сравнения скоростей всех упомянутых vpn соединений в Mikrotik.

Сравнение скорости vpn каналов в mikrotik

VPN Туннель	Шифрование	Скорость (Мбит/с)
l2tp	нет	194
l2tp	IPsec AES-128 CBC	26
pptp	нет	194
pptp	MPPE128	71
openvpn	BF-128-CBC	24
eoip	нет	836
eoip	IPsec AES-128 CBC	27
gre	нет	247
gre	IPsec AES-128 CBC	29,7

Приведенная таблица наглядно показывает разницу в различных методах шифрования. С помощью нее можно быстро оценить, к каким потерям производительности может привести шифрование. Сейчас все по-умолчанию все шифруют, но если разобраться, очень часто это не требуется. Можно пойти на некий компромис и использовать pptp сервер, который хоть и не обеспечивает 100% безопасное шифрование, но тем не менее скрывает трафик от просто любопытных глаз и имеет неплохое быстродействие. В любом случае трафик просто так не прочитать, надо целенаправленно приложить усилия для дешифровки. В некоторых случаях такой защиты будет достаточно.

Заключение

Не понравилась статья и хочешь научить меня администрировать? Пожалуйста, я люблю учиться. Комментарии в твоем распоряжении.

Расскажи, как сделать правильно!

Изначально не планировал писать такую большую и подробную статью. Аппетит приходит во время еды. По мере того, как стал углубляться в тему, становилось все интереснее и интереснее попробовать разные варианты и сравнить их. В итоге я перебрал все известные vpn подключения в mikrotik. Не дошли руки только до SSTP, но я точно знаю, что он будет очень медленно работать на RB951G-2hnD и в целом на микротиках медленнее всех остальных решений. Не думаю, что его использование будет оправданно.

Статью писал несколько дней, мог что-то напутать, опечататься или ошибиться. Все замечания принимаю в комментариях. Надеюсь, мой материал исследование на тему настройки vpn соединений в микротиках был вам интересен и полезен. Единственное, о чем жалею, что не затронул тему настройки pptp, l2tp и openvpn подключений на клиентских устройствах сотрудников. Без них материал на тему настройки vpn получился не полноценным, ведь это важная часть работы vpn тоннелей. Их используют не только для объединения офисов, но и для подключения удаленных сотрудников.