МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФГБОУ ВО «Пермский государственный национальный исследовательский университет»

Физический факультет

Кафедра радиоэлектроники и защиты информации

**Создание дистрибутива Linux для vpn-сервера**

Курсовая работа

|  |  |
| --- | --- |
|  | Работу выполнил  студент 3 курса специальности  “Информационная безопасность автоматизированных систем”  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ А. И. Уланов |
| Оценка работы научным руководителем  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (оценка, выставленная руководителем)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (дата, подпись руководителя) | Научный руководитель:  к. ф.-м. н., старший преподаватель  кафедры радиоэлектроники и защиты  информации  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ И.Н. Черепанов |
| Оценка работы комиссией по защите курсовых работ  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (оценка, выставленная комиссией)  \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_  (дата, подпись председателя комиссии) |  |

Пермь, 2017

ОГЛАВЛЕНИЕ

[ВВЕДЕНИЕ 3](#_Toc480881150)

[О технологии VPN 4](#_Toc480881151)

[РАЗЛИЧНЫЕ РЕАЛИЗАЦИИ VPN-СЕРВЕРА В ОПЕРАЦИОННОЙ СИСТЕМЕ LINUX. 6](#_Toc480881152)

[PPTP 7](#_Toc480881153)

[STRONGSWAN 13](#_Toc480881154)

[СОЗДАНИЕ ДИСТРИБУТИВА НА ОСНОВЕ УСТАНОВЛЕННОЙ СИСТЕМЫ. 19](#_Toc480881155)

[ТЕСТИРОВАНИЕ 20](#_Toc480881164)

[ЗАКЛЮЧЕНИЕ 23](#_Toc480881165)

[СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ 24](#_Toc480881166)

[Приложение 1 25](#_Toc480881167)

[Приложение 2 30](#_Toc480881168)

[Приложение 3 37](#_Toc480881169)

[Приложение 4 44](#_Toc480881170)

# ВВЕДЕНИЕ

Сегодня работая в интернете нельзя быть уверенным в безопасности передаваемых дынных. Любая информация, передаваемая по открытым каналам, может быть попасть в руки злоумышленников, неприметный парень, сидящий за дальним столиком в кафе вполне может оказаться преступником, посягающим на ваши данные. Это могут быть ваши личные фотографии, реквизиты банковских карт если вы пользуйтесь сетью в личных целях, и информация, представляющая коммерческую тайну (или какую-либо другу. тайну), персональные данные ваших клиентов, если вы используйте интернет в рабочих целях. Технология VPN позволяет использовать открытые сети для передачи информации не опасаясь за ее безопасность.

Виртуальная частная сеть создает безопасный, зашифрованный туннель, через который проходят ваши данные. Любое приложение требующее подключение к интернету может использовать vpn-туннель, это может быть веб-браузер, клиент электронной почты, программы обмена мгновенными сообщениями, приложения для работы с сервисами, используемыми вашей компанией (1С-Бугалтерия). VPN позволяет скрыть все, что вы делали в интернете от посторонних глаз, маскировать ваше фактическое местоположение и дает возможность получить доступ к любому сервису (например корпоративному) независимо от того где вы живете или путешествуйте.

Цель работы: создание дистрибутива Linux для vpn-сервера.

Задачи:

1. Изучить различные варианты реализации vpn-серверов для операционной системы Linux.
2. Выбрать реализации, которые будут представлены в дистрибутиве.
3. Разбить настройки сервера на две категории. Те, что могут быть выполнены сразу, и те, выполнение которых ляжет на конечного пользователя.
4. Выполнить предварительные настройки.
5. Свернуть работающую систему в дистрибутив.
6. Протестировать получившийся дистрибутив.

# О технологии VPN

VPN представляет собой объединение отдельных машин или локальных сетей в виртуальной сети, которая обеспечивает целостность и безопасность передаваемых данных. Она обладает свойствами выделенной частной сети и позволяет передавать данные между двумя компьютерами через промежуточную сеть (internetwork), например Internet.

VPN отличается рядом экономических преимуществ по сравнению с другими методами удаленного доступа. Во-первых, пользователи могут обращаться к корпоративной сети, не устанавливая c ней коммутируемое соединение, таким образом, отпадает надобность в использовании модемов. Во-вторых, можно обойтись без выделенных линий.

Имея доступ в Интернет, любой пользователь может без проблем подключиться к сети офиса своей фирмы. Следует заметить, что общедоступность данных совсем не означает их незащищенность. Система безопасности VPN - это броня, которая защищает всю корпоративную информацию от несанкционированного доступа. Прежде всего, информация передается в зашифрованном виде. Прочитать полученные данные может лишь обладатель ключа к шифру.

Подтверждение подлинности включает в себя проверку целостности данных и идентификацию пользователей, задействованных в VPN. Первая гарантирует, что данные дошли до адресата именно в том виде, в каком были посланы. Далее система проверяет, не были ли изменены данные во время движения по сетям, по ошибке или злонамеренно. Таким образом, построение VPN предполагает создание защищенных от постороннего доступа туннелей между несколькими локальными сетями или удаленными пользователями.

Для построения VPN необходимо иметь на обоих концах линии связи программы шифрования исходящего и дешифрования входящего трафиков. Они могут работать как на специализированных аппаратных устройствах, так и на ПК с такими операционными системами как Windows или Linux.

Управление доступом, аутентификация и шифрование - важнейшие элементы защищенного соединения.

Туннелирование (tunneling), или инкапсуляция (encapsulation), - это способ передачи полезной информации через промежуточную сеть. Такой информацией могут быть кадры (или пакеты) другого протокола. При инкапсуляции кадр не передается в сгенерированном узлом-отправителем виде, а снабжается дополнительным заголовком, содержащим информацию о маршруте, позволяющую инкапсулированным пакетам проходить через промежуточную сеть (Internet). На конце туннеля кадры деинкапсулируются и передаются получателю.

# РАЗЛИЧНЫЕ РЕАЛИЗАЦИИ VPN-СЕРВЕРА В ОПЕРАЦИОННОЙ СИСТЕМЕ LINUX.

Для операционной системы Linux существует несколько различных вариантов реализации VPN серверов.

1. Использование IPsec (LibreSwan strongSwan). IPsec — это набор протоколов для обеспечения защиты данных, передаваемых по межсетевому протоколу IP. Позволяет осуществлять подтверждение подлинности (аутентификацию), проверку целостности и/или шифрование IP-пакетов. IPsec также включает в себя протоколы для защищённого обмена ключами в сети Интернет.
2. пакет pptpd. PPTPD – это стандартный пакет для операционных систем семейства Linux, предназначенный для развертывания PPTP-сервера.
3. OpenVPN— свободная реализация технологии виртуальной частной сети (VPN) с открытым исходным кодом для создания зашифрованных каналов типа точка-точка или сервер-клиенты между компьютерами. Она позволяет устанавливать соединения между компьютерами, находящимися за NAT и сетевым экраном, без необходимости изменения их настроек. OpenVPN была создана Джеймсом Йонаном и распространяется под лицензией GNU GPL.

Для реализации дистрибутива будем использовать пакет pptpd, так как он является наиболее простым в развертывании. И strongSwan, который позволяет обеспечить высокий уровень безопасности соединения. Так же ватажным фактором при выборе strongSwan стало наличие качественной документации на официальном сайте.

# PPTP

Point-to-Point Tunneling Protocol – это протокол туннелирования точка-точка, позволяющий компьютеру устанавливать защищённое соединение с сервером за счёт создания специального туннеля в стандартной сети. PPTP помещает (инкапсулирует) кадры PPP в IP-пакеты для передачи по глобальной IP-сети, например, Интернет. PPTP может также использоваться для организации туннеля между двумя локальными сетями. РРТР использует дополнительное TCP-соединение для обслуживания туннеля.

Протокол обрел свою популярность в силу того, что это первый протокол туннелирования, поддерживаемый корпорацией Microsoft. Все версии Windows (с Windows 95) поддерживают этот протокол.

За время своего существования безопасность протокола многократно анализировалась, за время жизни протокола в нем было обнаружено большое количество уязвимостей, несмотря на это протокол имеет право на жизнь, его применение позволит усложнить жизнь злоумышленникам, но следует учесть все уязвимые места протокола при настройке сервера.

На данный момент туннелирование по протоколу PPTP поддерживается всеми популярными платформами (Microsoft Windows, Android, Linux, Mac OS X) за исключением IOS, поддержка протокола в которой прекращена с IOS10.

**Настройка PPTP**

Все действия необходимо выполнять, войдя в систему как суперпользователь (root).

В первую очередь необходимо установить серверное приложение в систему. Для этого выполняем команду:

apt-get install ppp pptpd

Далее приступим к настройке сервера.

Основные настройки содержатся в файлах, для редактирования файлов можно использовать любой текстовый редактор, (nano, vim, jed и другие), я использовал jed

**Файл /etc/pptpd.conf**

Опции:

1. ppp [путь]–расположение сервиса pppd, являющегося управляющим для PPTP сервера, по умолчанию используется /usr/sbin/pppd

2. option [путь] – путь к файлу с остальными настройками (он будет описан ниже), по умолчанию используется /etc/ppp/options.pptpd.

3. debug – инструмент отладки, в случае использования в логи сервиса будет записана более подробная информация.

4. stimeout [время в секундах] – время ожидания PPTP-пакетом, прежде, чем он будет передан в управляющую программу pptpctrl и попадет к клиенту (защита от DoS-атак), по умолчанию 10 секунд.

5. noipparam – pppd не будет знать реальный ip клиента, по умолчанию отключен.

6. bcrelay [Название интерфейса] – позволяет клиентам рассылать широковещательные запросы, которые сервер получает через указанный интерфейс.

7. listen [ ip-адрес ] – адрес интерфейса на котором сервер будет слушать порт №1723, по умолчанию сервер работает на всех сетевых интерфейсах операционной системы.

8. localip [ip-адрес] – ip-адрес (диапазон адресов), который будет использоваться в качестве адреса сервера в виртуальной сети.

9. remoteip [диапазон ip-адресов ] –диапазон ip-адресов, из которого клиенты будут получать свои адреса.

10. speed [ значение скорости байт/с ] – максимальная скорость соединения, по умолчанию 115 200 байт/с.

11. connections n – максимальное количество соединений с pptp-сервером, по умолчанию используется значение 100. Эту опцию необходимо отредактировать в соответствии с опцией remoreip.

12. delegate - отключает опцию remoteip, ip адреса будут раздаваться в соответствии с инструкциями в файле chap-secrets, так же может использоваться radius-сервер.

13. logwtmp – позволяет серверу передавать данные аутентификации пользователей через функцию logwtmp.

Этот файл я оставил без изменений. Для работы сервера конечному пользователю необходимо задать адрес сервера, диапазон IP адресов, которые будут выдаваться клиентам и откорректировать максимальное количество соединений в соответствии с этим диапазоном.

#ip-адрес сервера в виртуальной сети

localip 192.168.2.1

# Диапазон адресов для клиентов сервера

remoteip 192.168.2.100-220

#Максимальное количество соединений

connections 120

**Файл /etc/ppp/pptpd-options**

Опции:

1. name [название сервера] – название сервиса, которое необходимо указать в файле /etc/ppp/chap-secrets

2. refuse-[ название протокола аутентификации или метода шифрования] – запрет на использование указанного протокола или метода шифрования.

3. require-[ название протокола аутентификации или метода шифрования] – требование использовать указанный протокол или метод шифрования.

4. ms-dтs [ip-адресс] – первичный и вторичный адрес dns-сервера, запись распложенная в файле выше будет трактоваться как адрес первичного dns-сервера, ниже вторичного. Всего в файле может быть две таких записи.

5. proxyarp – опция позволяет клиентам работать с локальной сетью без использования NAT, использует ARP.

6. debug , dump – запись дополнительной информации о ходе соединения в логи.

7. logfile [путь] – расположение файла, в который будет писаться лог

8. nologfd – блокирует запись в системный журнал через функцию stderr, по умолчанию активна.

9. novj – отключение зажатия Вана Якобсона для заголовков

10. nodefaultroute – отключение автоматического внесения изменений в таблицу маршрутизации. устанавливающие подключившегося клиента маршрутом по умолчанию

11. idle – количество секунд простоя, после которого соединение будет сброшено

12. lock – опция блокирует интерфейс, делая невозможным его использование другими соединениями, пока интерфейс используется

файл содержит и другие параметры, указанные выше является наиболее важными и часто используемыми.

Редактирование файла.

• Изменим название сервера:

name vpn-pptpd

• По умолчанию на сервере запрещены все виды шифрования, кроме MS-CHAPv2 с использованием 128-битного ключа, что обеспечивает максимально возможную безопасность при работе с PPTP. Оставим эти параметры без изменений.

• debug, dump – не будем использовать эти опции, их использование ведет к непомерному разрастанию файла логирования. Опцию nologfd оставим включенной по той же причине.

• novj – оставим опцию активной, системы использующие данный метод сжатия сейчас применяться редко, а использование метода ведет к снижению уровня безопасности соединения.

• nodefaultroute – поскольку сервер рассчитан на использование по схеме сервер – клиенты, оставим эту опцию активной.

• lock – во избежание конфликтов в системе оставим опцию активной.

• Укажем путь к файлу, в который будет писаться лог:

logfile [/var/log/pptpd.log]

• Укажем время простоя до сброса соединения

idle 1000

**Файл /etc/ppp/chap-secrets**

В этом файле конечному пользователю необходимо указать учетные данные, при помощи которых клиенты будут подключаться к серверу.

Файл содержит строки следующего вида:

[Имя клиента] [Имя сервера] [пароль] [ip-адрес]

Вместо значения ip-адреса можно указа “\*”, в этом случае ip будет выдан из пула, который указан в файле /etc/ppp/pptpd-options, ip-адреса указанные здесь явно могут не принадлежать пулу, однако их следует учитывать при задании максимального количества соединений.

например:

user1 vpn-pptpd password \*

user2 vpn-pptpd password 192.168.2.20

Для добавления пользователей можно воспользоваться командой echo, команда дописывает строку в конец файла

echo “[строка]” >> [путь к файлу]

echo "test1 pptpd 11111 \* " >> /etc/ppp/chap-secrets

Для работы сервера необходимо настроить межсетевой экран iptables

# Пропускать все пакеты с интерфейсов ppp\*, например ppp0

iptables -A INPUT -i ppp+ -j ACCEPT

iptables -A OUTPUT -o ppp+ -j ACCEPT

# Пропускать входящие соединения на порт 1723 (PPTP)

iptables -A INPUT -p tcp --dport 1723 -j ACCEPT

# Пропускать все пакеты GRE

iptables -A INPUT -p 47 -j ACCEPT

iptables -A OUTPUT -p 47 -j ACCEPT

# Включить форвардинг IP

iptables -F FORWARD

iptables -A FORWARD -j ACCEPT

# Включить NAT для интерфейсов eth0 и ppp\*

iptables -A POSTROUTING -t nat -o eth0 -j MASQUERADE

iptables -A POSTROUTING -t nat -o ppp+ -j MASQUERADE

Далее сохраняем настройки в файл

iptables-save > /etc/iptables.rules

Чтобы правила подхватывались при загрузке системы, создаем новый файл /etc/network/if-pre-up.d/iptables в который записываем следующий скрипт:

#!/bin/sh

iptables-restore < /etc/iptables.rules

exit 0

Далее выполняем команду, включающую скрипт в автозагрузку.

chmod +x /etc/network/if-pre-up.d/iptables

При настройке сервера конечный пользователь должен:

1. В файле /etc/pptpd.conf указать адрес сервера, пул ip адресов и максимальное количество подключений.
2. В файле /etc/ppp/chap-secrets завести учетные записи пользователей
3. перезагрузит сервер командой /etc/init.d/pptpd restart
4. update-rc.d pptpd enable добавить сервер в автозагрузку

Сервер за NAT.

Когда сервер находится за NAT, что это обычно бывает в случае, когда сервер размещен за домашним роутером, необходимо выполнить дополнительные действия в виде проброса портов UDP port 500 и UDP port 4500 на сервер. Также в разрешении нуждаться протоколы 50 (ESP) и 51 (AH).

Инструкция по настройке клиентов приведена в приложении 1.

# STRONGSWAN

IPsec – это стандарт создания защищенных соединений в IP-сетях. В системах Linux, протоколы входящие в стандарт работают на уровне ядра.

В дистрибутивах Debian до версии 8 для реализации IPsec по умолчанию находились демоны rocoon и openswan. Начиная с 8 openswan был заменен на strongswan. strongSwan представляет собой комплексное IPsec-решение для обеспечения шифрования и аутентификации между серверами и клиентами

В данной реализации VPN для подключения к шлюзу клиент используется протокол EAP (RFC 3748) совместно с Microsoft CHAP version 2 для аутентификации.

Этот протокол используется в VPN – клиенте Windows. В дополнении к идентификации IKEv2 по IP-адресу клиент использует EAP идентификацию по определенному на шлюзе имени и паролю.

Шлюз проходит аутентификацию на клиенте с помощью сертификата RSA.

Основное преимущество данного решения по отношению к VPN на сертификатах в том, что не нужно импортировать каждый сертификат на клиента, достаточно знать только логин и пароль. Дополнительным преимуществом является использование для установки IPSEC соединения протокола IKEv2, который обладает рядом преимуществ по отношению к устаревшему IKEv1.

**Настройка Strongswan.**

Файлы конфигурирования по умолчанию хранятся в директории /etc/ и имеют следующие названия:

* ipsec.conf – определяет параметры IPSEC-соединений и параметры подключений в целом;
* ipsec.secrets – служит для хранения ссылок на сертификаты и ключи аутентификации;
* strongswan.conf – для подключения криптографических алгоритмов и дополнительных функций.

Помимо этого во время установки программного обеспечения для хранения сертификатов и CRL-файлов используемых демонами pluto и charon создается директория /etc/ipsec.d, в которой находятся следующие каталоги:

* private – содержит закрытые ключи RSA и ECDSA;
* certs – содержит сертификаты X.509 и PGP;
* crls – хранит список отозванных сертиифкатов;
* cacerts – хранит доверенные сертификаты CA;
* ocspcerts – содержит подписанные OCSP сертификаты;
* reqs – содержит запросы на сертификаты в формате PKCS#10.

Файл /etc/ipsec.secrets содержит неограниченное количество следующих типов ключей (паролей):

* RSA для определения пароля к сертификату открытого ключа;
* ECDS для определения пароля к сертификату открытого ключа;
* PSK для определения Pre-shared ключа;
* EAP для учетных записей EAP;
* NTLM для учетных записей NTLM;
* XAUTH для учетных записей XAUTH;
* PIN для пин-кода смарт-карт.

Устанавливаем демон

apt-get install strongswan

В первую очередь настроим OPENSSL,

[ CA\_default ]

dir = /etc/ipsec.d # Основная директория, в ней будут храниться все сертификаты

certificate = $dir/cacerts/strongswanCert.pem # Здесь у нас будет лежать CA сертификат

private\_key = $dir/private/strongswanKey.pem # А здесь закрытый ключ CA сертификата

В настройках OPENSSL пользователю будет необходимо указать внешний IP сервера при генерации сертификата

subjectAltName=IP:<внешний IP сервера >

Далее непосредственно настройки демона

* Файл strongswan.conf

charon {

load = curl test-vectors aes des sha1 sha2 md5 pem pkcs1 gmp random x509 revocation hmac xcbc cmac ctr ccm gcm stroke kernel-netlink socket-default updown eap-identity

}

* Файл ipsec.conf

Секция config setup, которая определяет базовые параметры:

config setup

strictpolicy=no

charonstart=yes

plutostart=no

charondebug="ike 2, knl 3, cfg 0"

Секция conn, в которой настраиваются соединения

conn %default / определяет базовые параметры всех IPSEC-соединений

ikelifetime=60m

keylife=20m

rekeymargin=3m

keyingtries=1

dpdaction=restart

dpdelay=30s

dpdtimeout=180s

conn rw / название IPSEC-соединения

left=<external\_IP> / адрес используемого внешнего интерфейса

leftsubnet=<subnet/prefix> / подсеть, к которой будем предоставлять доступ

leftid=<external\_IP>

leftcert=/etc/ipsec.d/certs/servercert.pem / сертификат, используемый для установки IKE SA

leftauth=pubkey / задаем авторизацию у клиента при помощи сертификата RSA

right=%any / разрешаем подключение с любого IP адреса

rightauth=eap-mschapv2 //Задаем метод авторизации MSCHAPv2-EAP

rightsendcert=never

rightsourceip=<subnet/prefix> /пул адресов, из которого будут выдаваться IP

auto=add / подключение будет инициироваться клиентом

keyexhcnage=ikev2

type=tunnel //тип подключения – туннель

Так же нам нужно указать в файле /etc/ipsec.secrets файл, в котором храниться открытый ключ, при конечной настройке сервера необходимо будет указать пароль, указанный при создании сертификата и учетные записи пользователей

: RSA /etc/ipsec.d/private/serverkey.pem "<пароль>" //указываем RSA сертификат

Для работы этого сервера так же, как и для PPTP, требуется включить маршрутизацию, этот процесс был описан выше.

Для работы сервера необходимо настроить межсетевой экран iptables. При настройке PPTP сервера мы создали файл с сохранёнными правилами. Допишем в него правила, необходимые для работы Strong swan и закомментируем их.

iptables -t nat -A POSTROUTING -j SNAT --to-source %IP сервера% -o eth+

iptables -A INPUT -p udp --dport 500 --j ACCEPT

iptables -A INPUT -p udp --dport 4500 --j ACCEPT

iptables -A INPUT -p esp -j ACCEP

Действия по настройке сервера, требуемые от пользователя.

Пользователь должен сгенерировать сертификаты, с которыми будет работать сервер.

В первую очередь необходимо сгенерировать CA сертификат.

openssl req -x509 -days 3650 -newkey rsa:2048 -keyout private/strongswanKey.pem -out cacerts/strongswanCert.pem

Далее необходимо конвертировать CA сертификат в формат p.12. Этот формат поддерживается в операционных системах семейства Windows и большинством других операционных систем.

openssl pkcs12 -export -inkey private/strongswanKey.pem -in cacerts/strongswanCert.pem -name "host" -certfile cacerts/strongswanCert.pem -caname "strongSwan Root CA" -out CAcert.p12

Далее нужно сгенерировать сертификат для сервера. Пред генерацией сертификата пользователю необходимо задать параметр subjectAltName в openssl.cnf

subjectAltName=IP:<внешний IP сервера >

Создаем запрос на сертификат.  
openssl req -newkey rsa:1024 -keyout private/serverkey.pem -out reqs/serverreq.pem

Запрашиваем сертификат у CA, с помощью ранее созданного запроса.  
openssl ca -in reqs/serverreq.pem -days 730 -out certs/servercert.pem -notext

Далее пользователю нужно завести учетные записи для пользователей и указать пароль для RSA, отредактировав файл /etc/ipsec.secrets

Учетные записи вноситься в формате

<Имя пользователя> : EAP "<Пароль>"

Следует отметить, что перед и после двоеточия необходим ровно один пробел.

: RSA /etc/ipsec.d/private/serverkey.pem "<пароль>" //необходимо указать пароль, заданный при создании сертификата

User1 : EAP "password" //пример регистрационной записи пользователя.

Для правильной настройки межсетевого экрана пользователю необходимо отредактировать файл etc/iptables.rules, пользователю нужно закомментировать блок правил для работы PPTP сервера, раскомментировать блок для работы strongswan и указать IP адрес сервера.

Далее нужно выполнить команду для применения настроек.

iptables-restore < /etc/iptables.rules

Включить автозагрузку демона и запустить его.

systemctl enable strongswan

systemctl start strongswan

Для подключения клиентов необходимо импортировать CA сертификат в формат p.12. на клиентские машины. Инструкция представлена в Приложении 3.

Инструкция по настройке клиентов приведена в Приложении 2.

Сервер за NAT

Когда сервер находится за NAT, что это обычно бывает в случае, когда сервер размещен за домашним роутером, необходимо выполнить дополнительные действия в виде проброса портов port 500 и port 4500 на сервер. Также в разрешении нуждаться протоколы 50 (ESP) и 51 (AH). Пример проброса порта на домашнем роутере Zyxel Keenetic Start приведен в Приложении 4.

СОЗДАНИЕ ДИСТРИБУТИВА НА ОСНОВЕ УСТАНОВЛЕННОЙ СИСТЕМЫ.

Помимо серверов в сборку добавлены программы, которые могут быть полезны бедующему пользователю дистрибутива.

* Midnight Commander – файловый менеджер
* Jed – текстовый редактор
* Sudo – программа для делегирования привилегированных ресурсов

Для создания дистрибутива будем использовать приложение Remastersys.

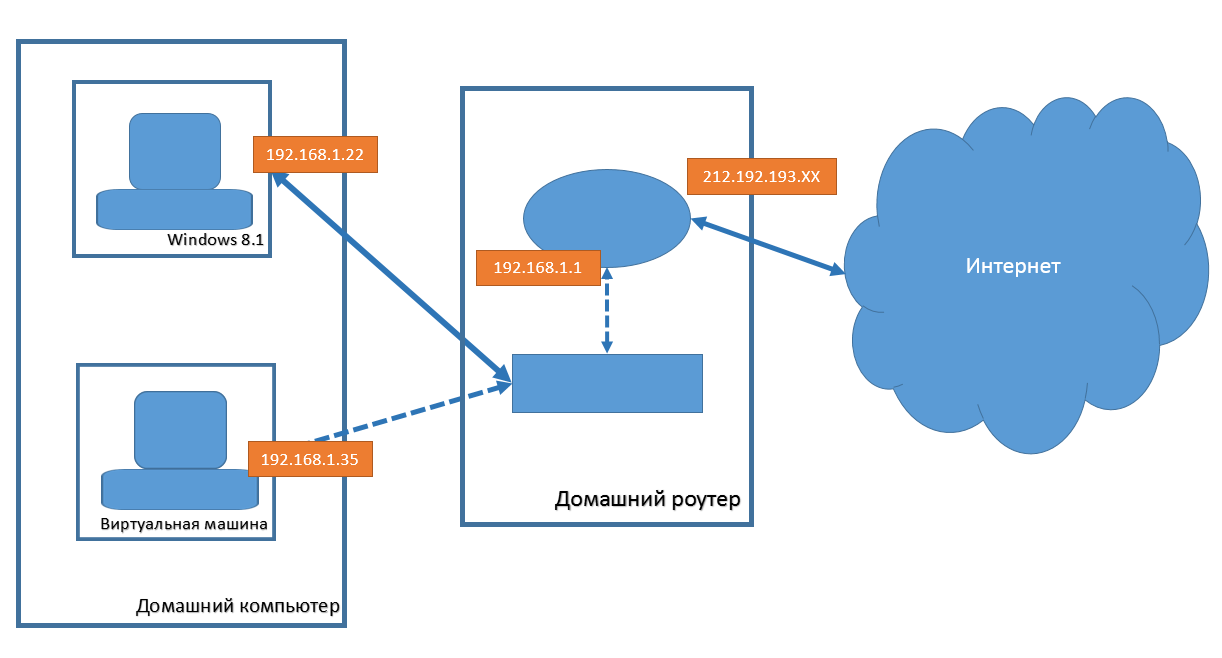
1. Выполняем команду «remastersys dist cdfs». Эта команда создает в домашней директории папку с файлами для будущей системы. Процесс создания этих файлов времязатратный и ресурсоемкий.
2. Выполняем команду «remastersys dist iso custom.iso». Эта команда создает файл образа диска, т. е. сам дистрибутив.

Далее созданный дистрибутив можно использовать.

# ТЕСТИРОВАНИЕ

Для тестирования работоспособности дистрибутива я выбрал бытовые задачи, вставшие передо мной.

Первое – это безопасный доступ к удаленному рабочему столу. Для доступа к удаленному рабочему столу необходимо держать открытым tcp порт 3389, этот факт может быть обнаружен злоумышленником при сканировании сети. После чего он может предпринять попытку получения доступа к компьютеру. Ниже представлена топология сети, в которой проводилось тестирование.



На домашнем компьютере установлена операционная система Windows 8.1 и запущена виртуальная машина с сервером. На домашнем роутере работает NAT, поэтому я включил проброс портов необходимых для работы сервера.

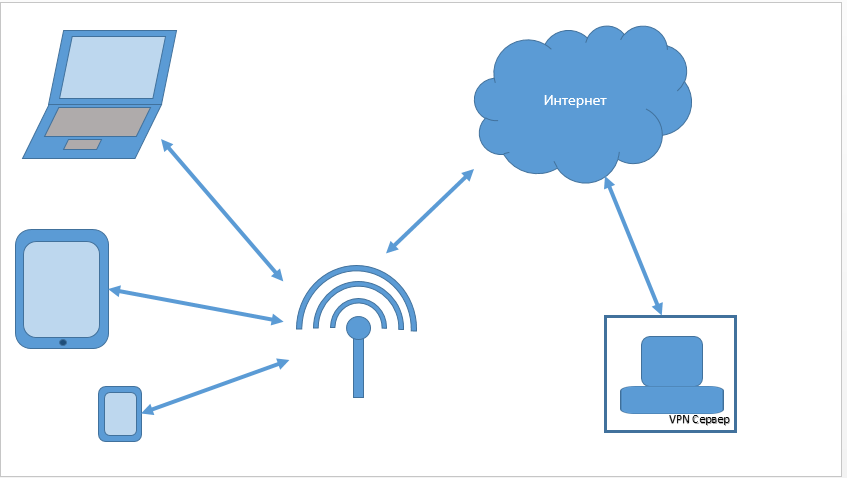
Следуя инструкциям, удалось настроить работу серверов. При работе и с pptp и с strongswan IP адреса подключившимся клиентам выдавались из подсети 192.168.2.0/24. Тестовые подключения выполнялись из домашней сети (192.168.1.0/24) и из внешней сети. В качестве внешней сети использовалась сеть оператора сотовой связи «Мегафон». Подключение к удаленному рабочему столу выполнялось на IP адрес компьютера в домашней сети (192.168.1.22). В качестве устройств, с которых выполнялись подключения выступали

* Компьютер с операционной системой Windows 8.1, в качестве клиента удаленного рабочего стола Remote Desktop Connection
* Компьютер с операционной системой linux ubuntu 16.04 в качестве клиента удаленного рабочего стола Remmina
* Планшетный компьютер Apple IPad с операционной системой IOS 9.3.5, в качестве клиента удаленного рабочего стола Remote Desktop Connection

Оба сервера работали стабильно с всеми устройствами.

Второе – безопасность при работе в недоваренных wi-fi сетях.

Для решения этой задачи возможно использовать сервер на домашнем компьютере, но для этого потребуется не выключать компьютер. Этот факт ставит под сомнение рациональность применения донного метода. Помимо того, домашний компьютер – это недостаточно надежная система, перебои с электроснабжением, человеческий фактор и многое другое могут вывести из строя сервер. Банальная перезагрузка потребует вмешательства для восстановления работы.

Учитывая все выше сказанное можно утверждать, что домашний компьютер для этой задачи не подходит. В данном случае целесообразно использовать удаленный сервер. Ниже представлена схема решения.

Мобильные устройства получая доступ к сети через публичный wi-fi, устанавливают шифрованное соединение с удаленным vpn сервером и используют его как шлюз для выхода в интернет.

Для тестирования дистрибутива я на короткий срок арендовал сервер у компании FirstByte и разместил на нем свой дистрибутив.

Тестирование работоспособности сервера производилось из wi-fi сети университета.

Подключения выполнялись с тех же устройств, что и в первом случае.

Ширина канала при подключении была равна ширине канала предоставляемой сетью университета. Время ответа при использовании команды ping ожидаемо выросло. В целом оба сервера работали стабильно. Однако стоит отметить, что при работе с PPTP сервером служба доставки контента vk.com работала некорректно.

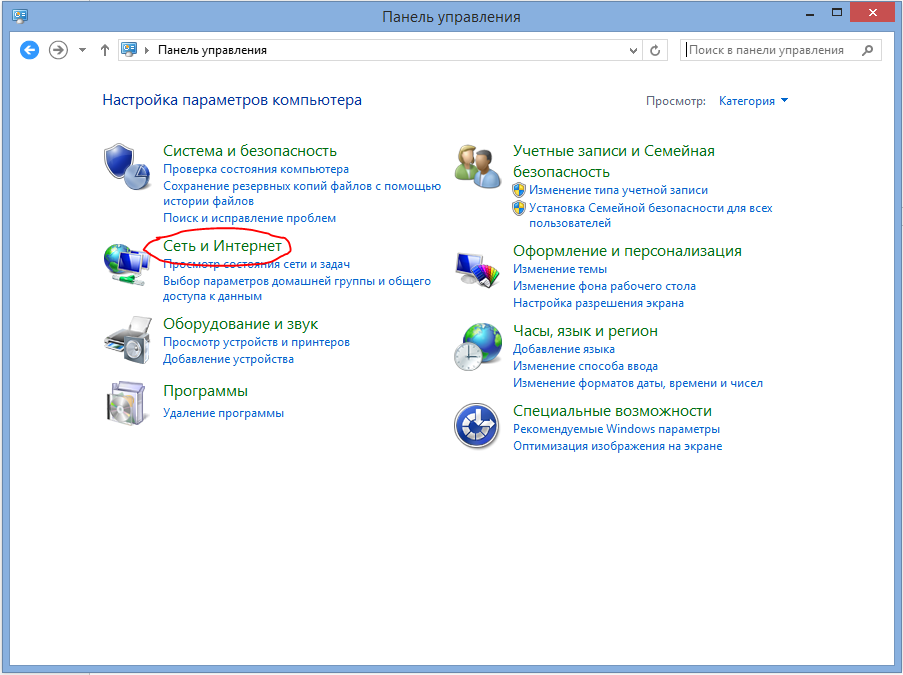
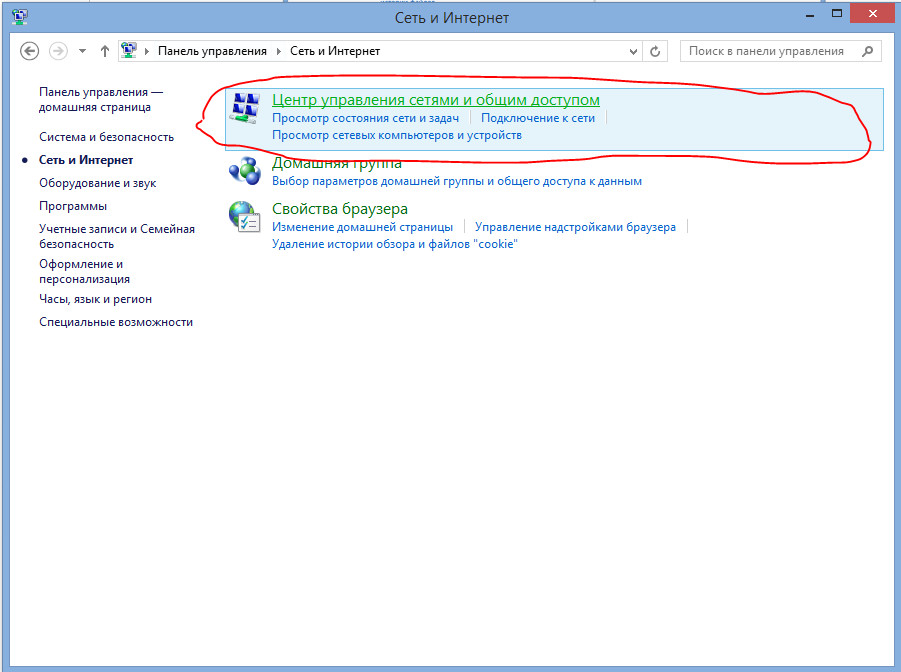
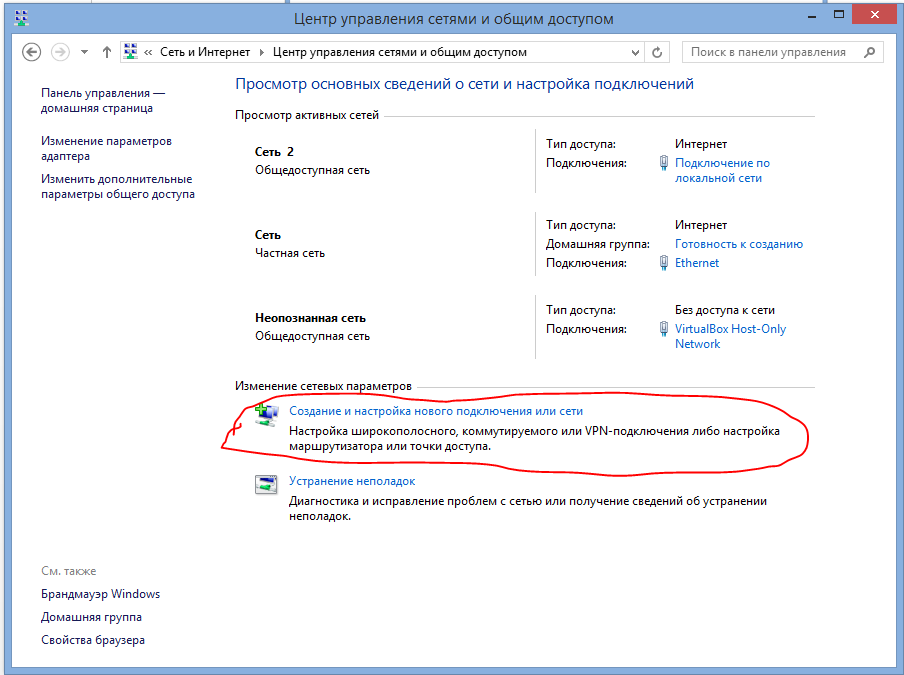
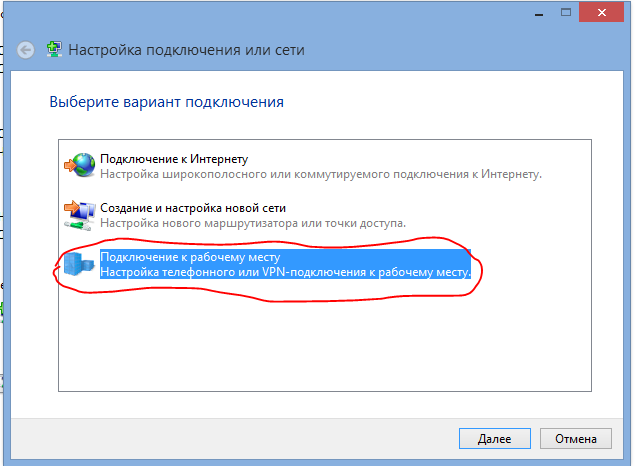
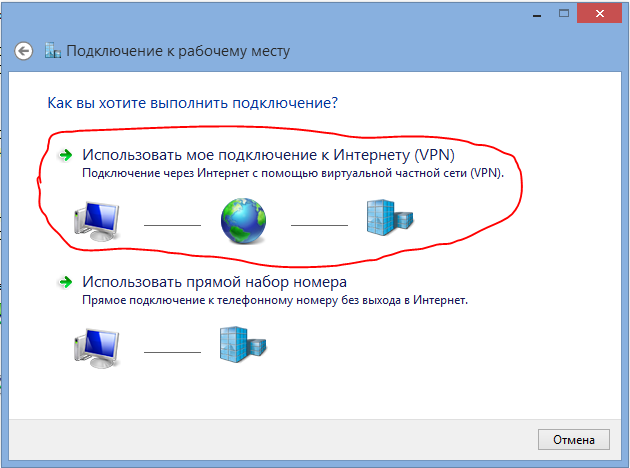
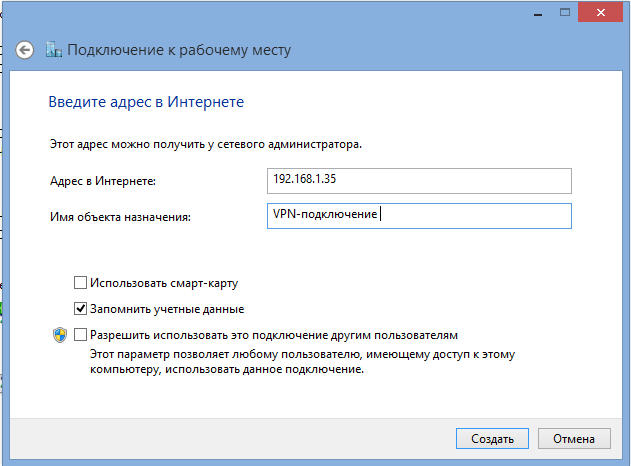
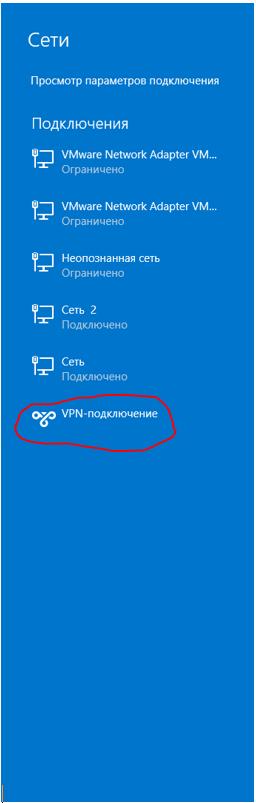
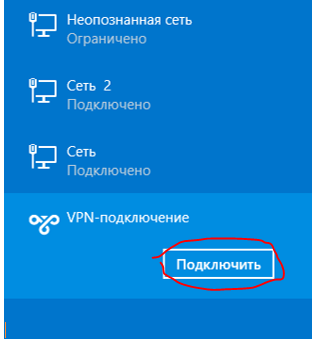
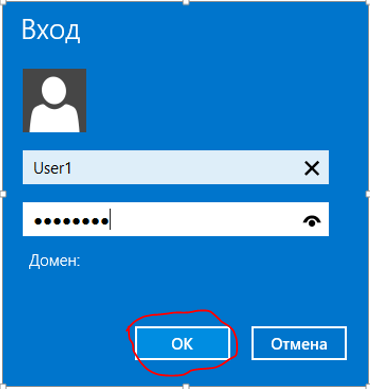
# ЗАКЛЮЧЕНИЕ

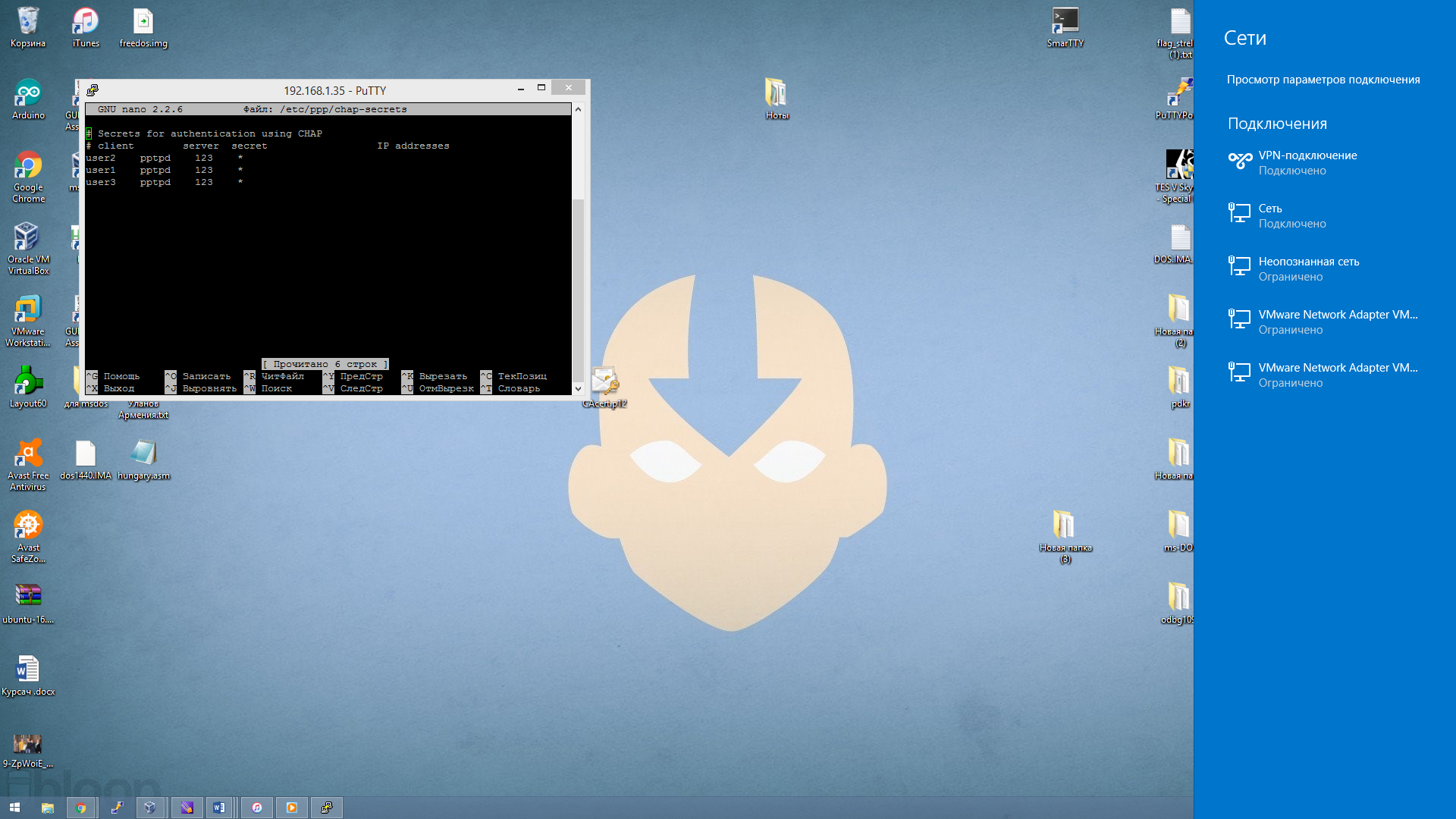
Разработанный дистрибутив поможет системному администратору с минимальными временными затратами развернуть собственный vpn-сервер и повысить уровень безопасности. Так же дистрибутив может помочь рядовым пользователям в решении повседневных задач, как было показано в тестовых примерах. В перспективе в данный дистрибутив планируется добавить web-интерфейс для облегчения процесса настройки сервера, а также добавить в дистрибутив другие реализации vpn-серверов для Linux.

# СПИСОК ИСПОЛЬЗУЕМОЙ ЛИТЕРАТУРЫ

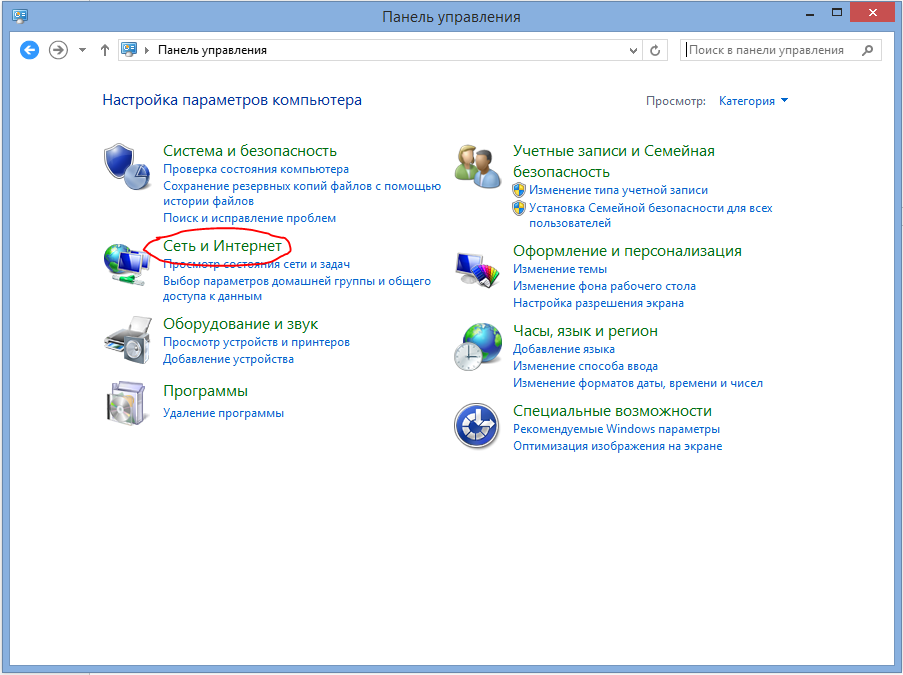
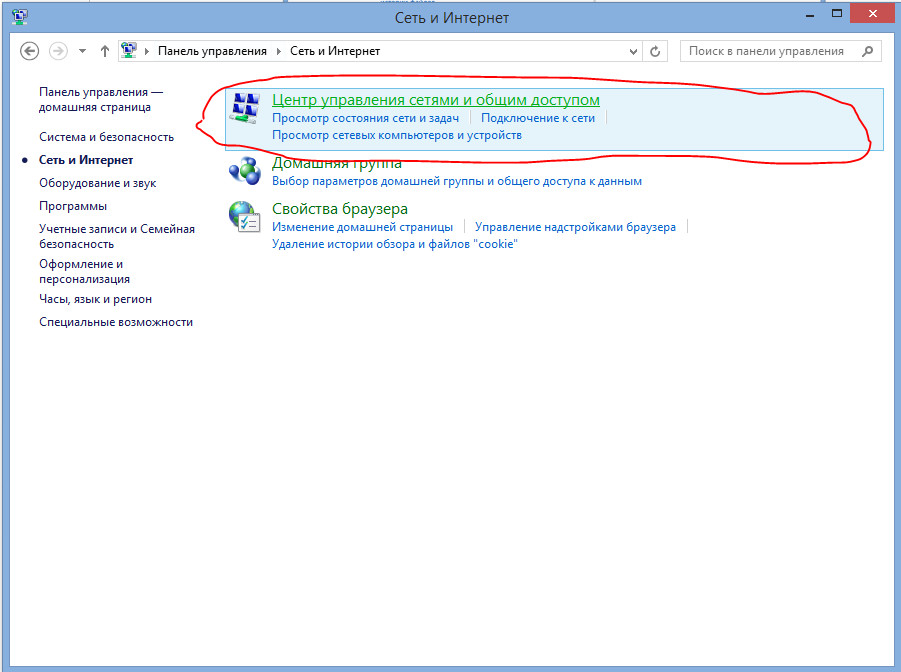
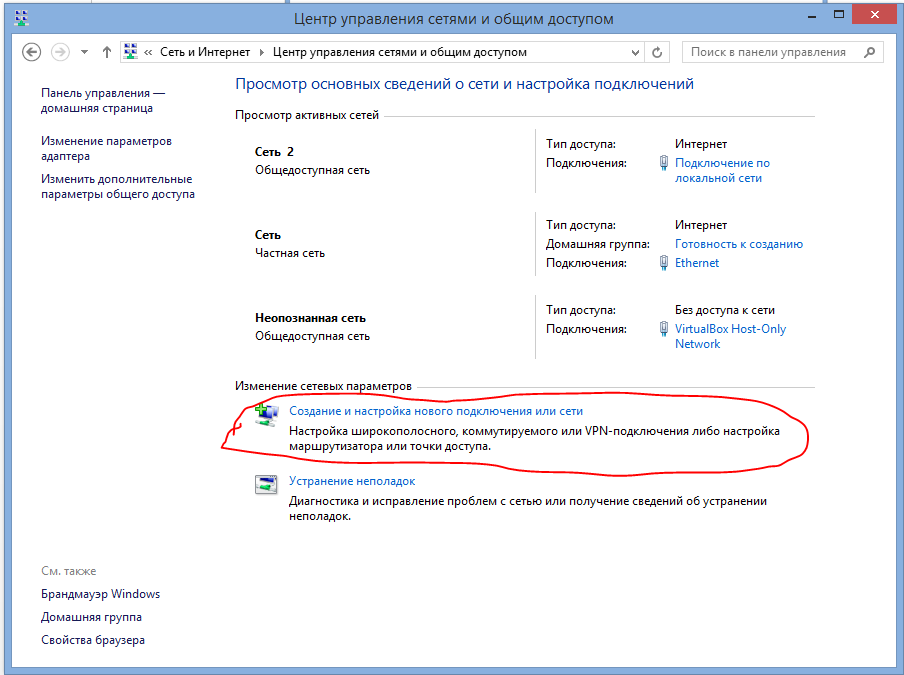
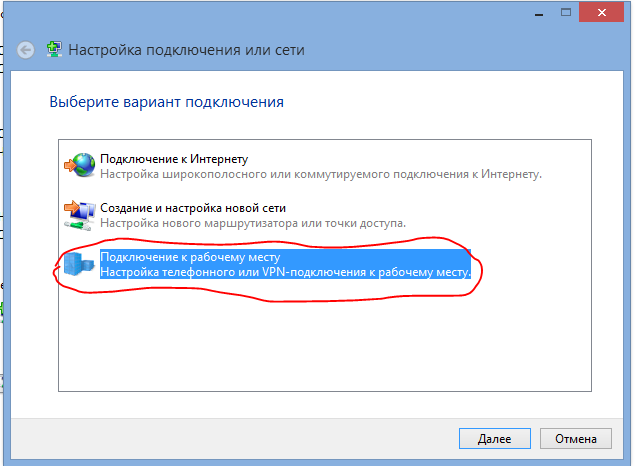
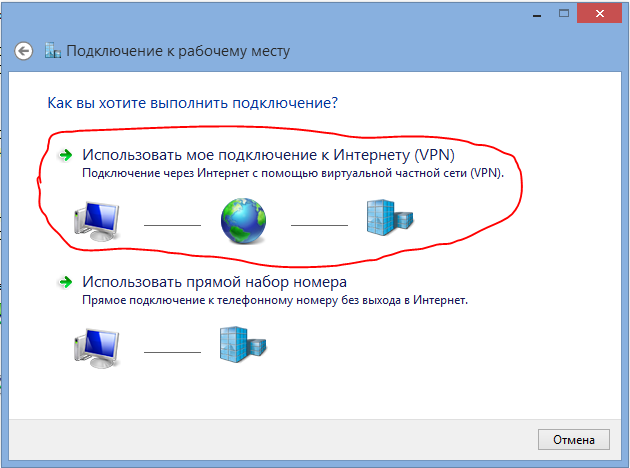
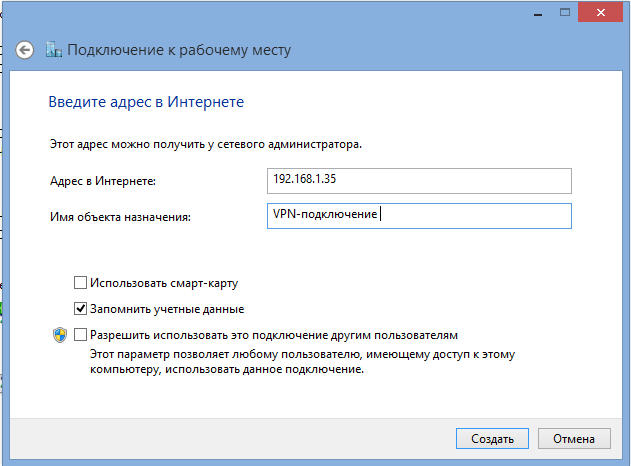
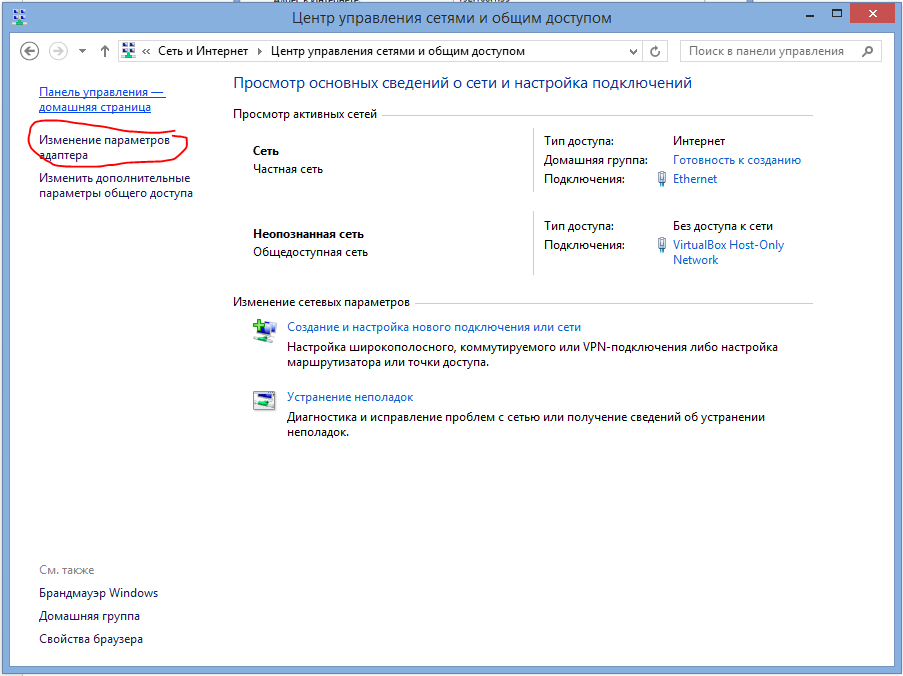
1. В. Браун «Виртуальные частные сети», изд. «Лори», 2016, стр. 16-37
2. В. Шаньгин «Информационная безопасность», изд. «ДМК Пресс», 2014, стр. 355-419
3. StrongSwan Documentation [Online] ttps://www.strongswan.org/documentation.html
4. IPSEC VPN on Ubuntu 15.04 with StrongSwan [Online] https://raymii.org/s/tutoriall s/IPS EC\_vpn\_with\_Ubuntu\_15.04.htmlhttp://pysnmp.sourceforge.net/quick-start.html

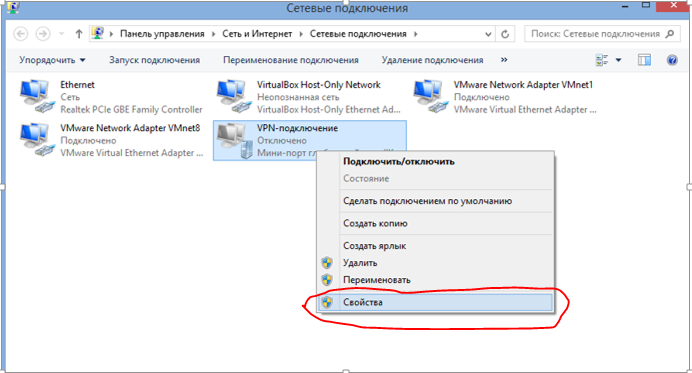
# Приложение 1 «Настройка PPTP-vpn на операционной системе Windows 8»

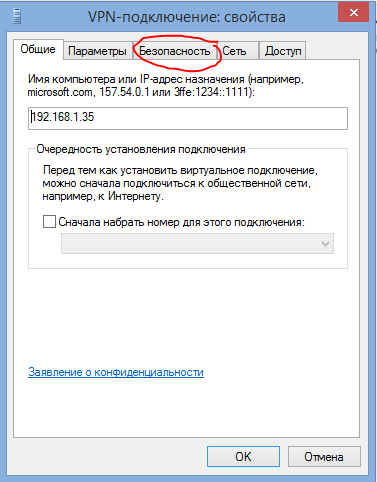
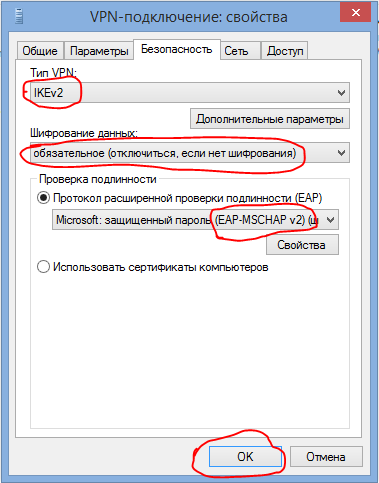
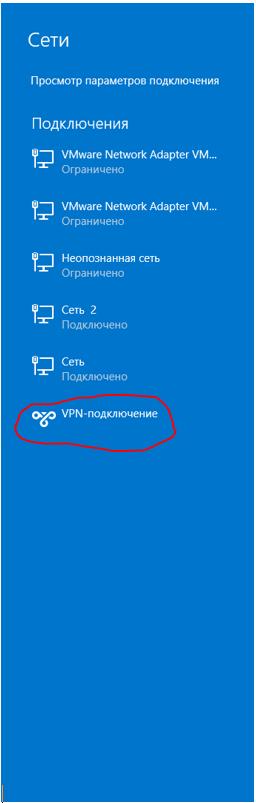
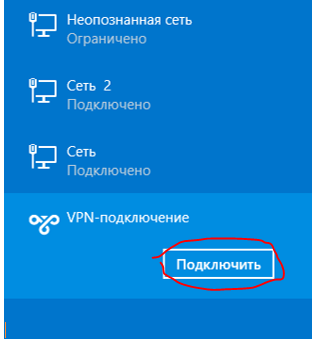
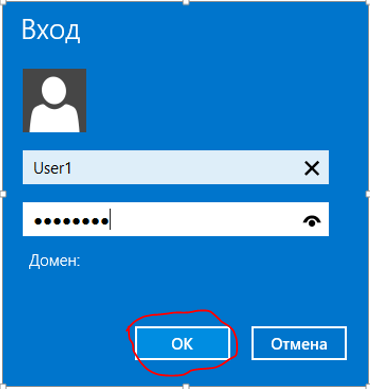
1. Заходим в панель управления и выбираем пункт «Сеть и Интерент»
2. Выбираем пункт «Центр управления сетями и общим доступом» 
3. В вкладке «Изменение сетевых параметров» выбираем пункт «Создание и настройка нового подключения или сети».
4. Выбираем пункт «Подключение к рабочему месту» и жмем «Далее»
5. Выбираем пункт «Использовать мое подключение к Интернету (VPN)»
6. В поле «Адрес в Интернете» вводим IP адрес сервера и нажимаем «Создать»
7. В появившейся сбоку панели кликаем на только что созданное подключение. 
8. Далее кликаем на пункт «Подключить». 
9. Вводим свои учетные данные (логин/пароль) и кликаем на «ОК»
10. Подключение настроено.

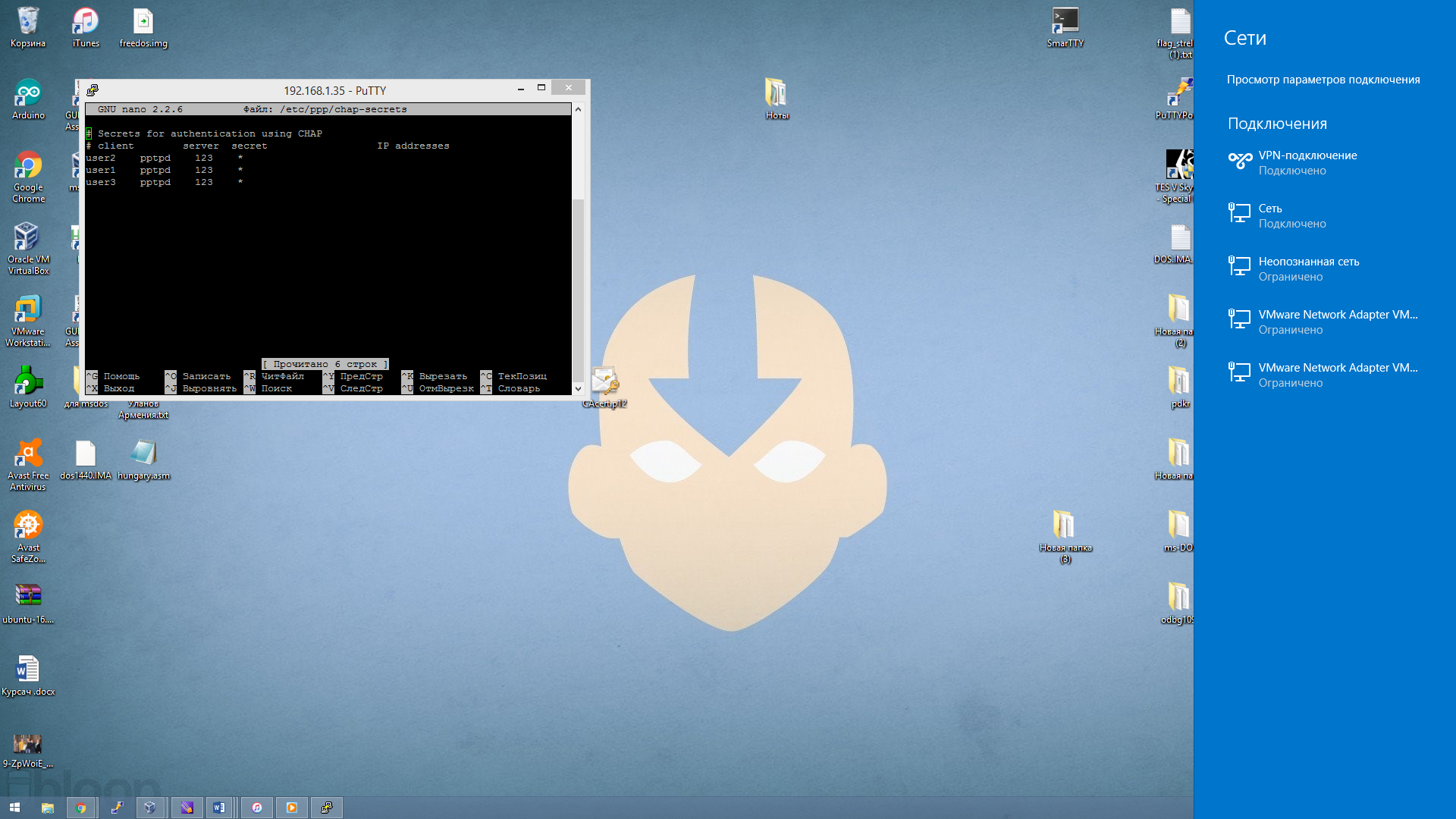


# Приложение 2 «Настройка подключения к серверу strongswan в Windows»

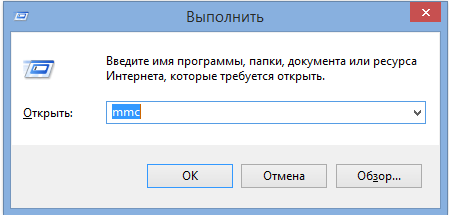
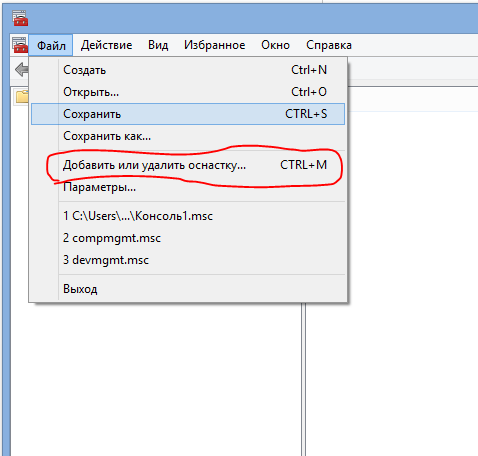
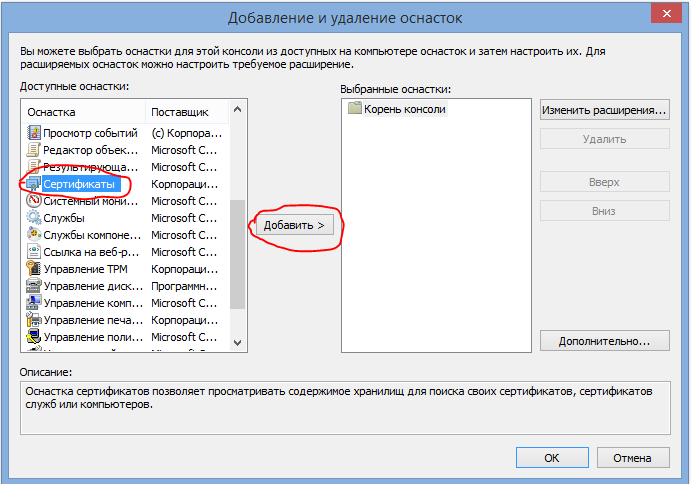
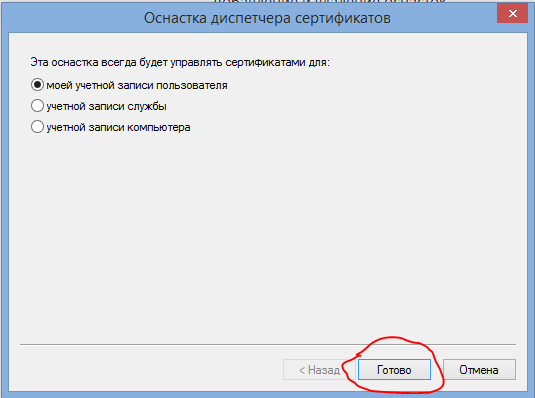
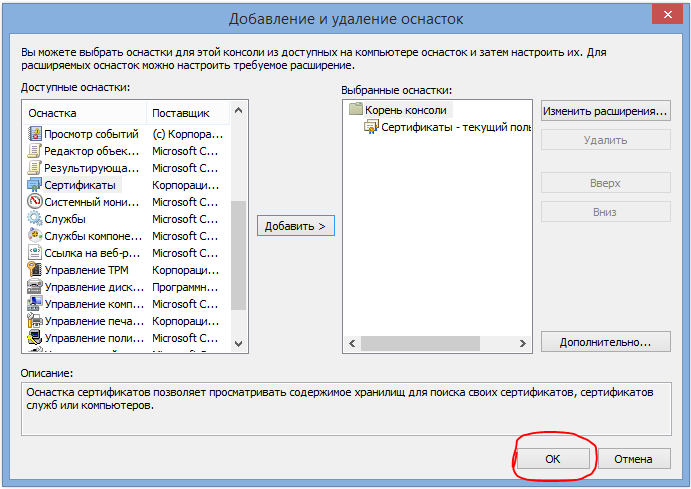
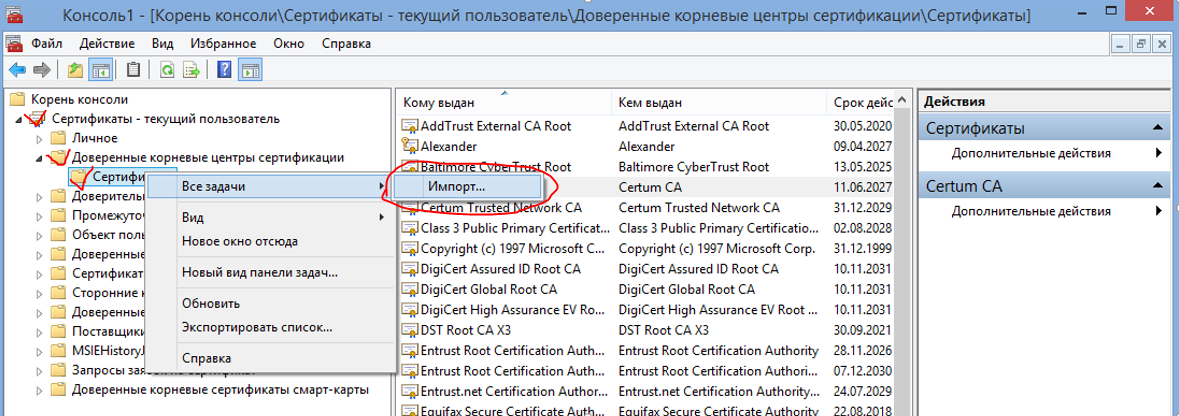
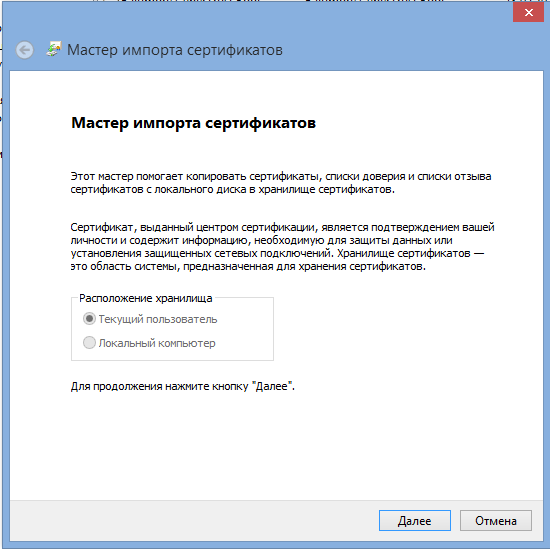
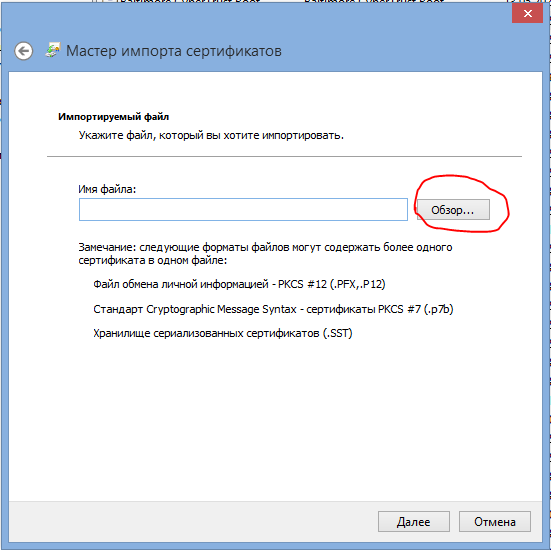
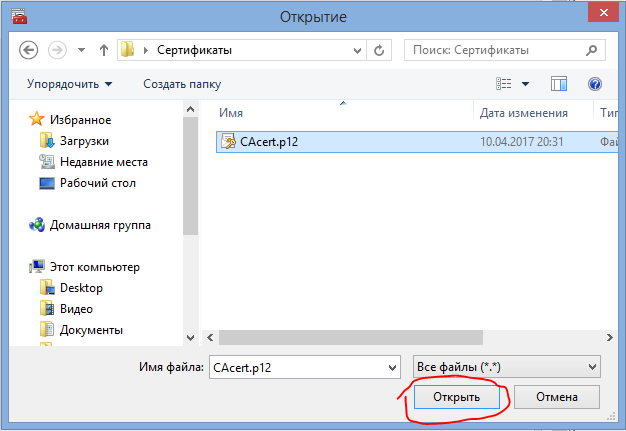
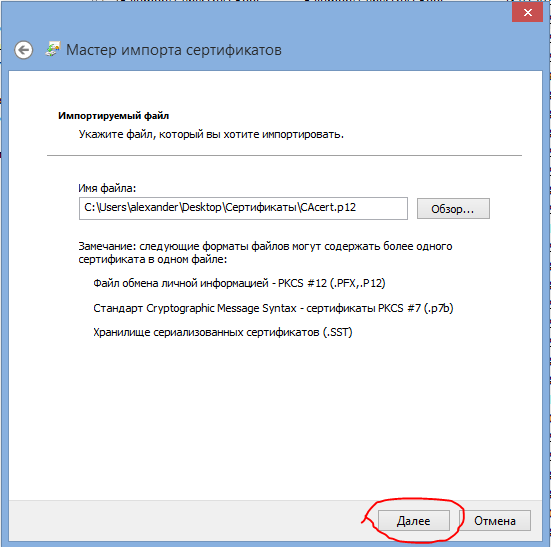
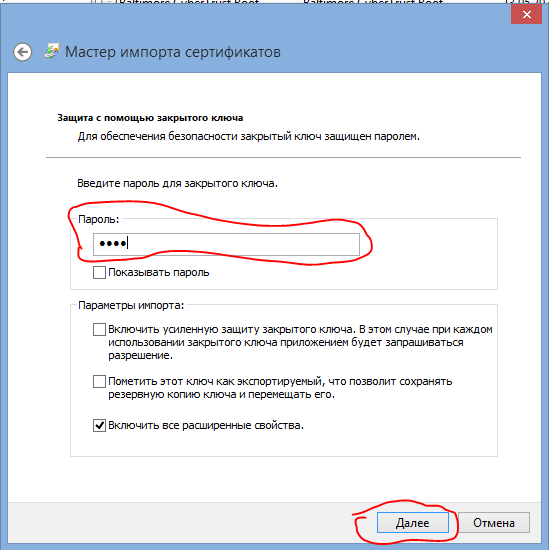
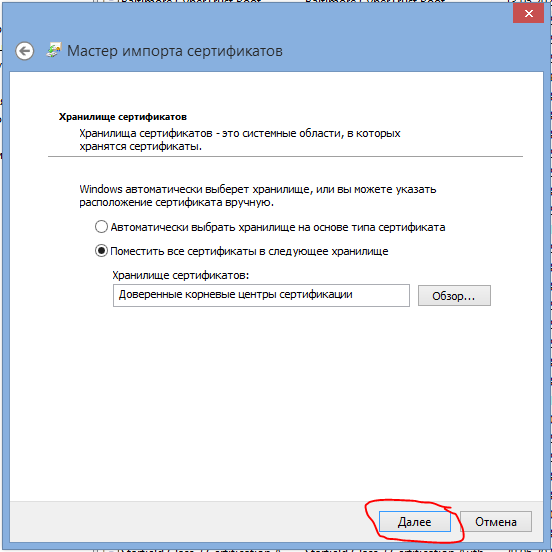
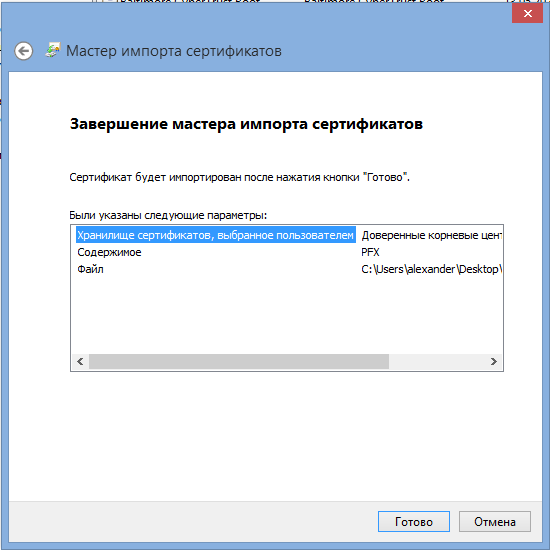
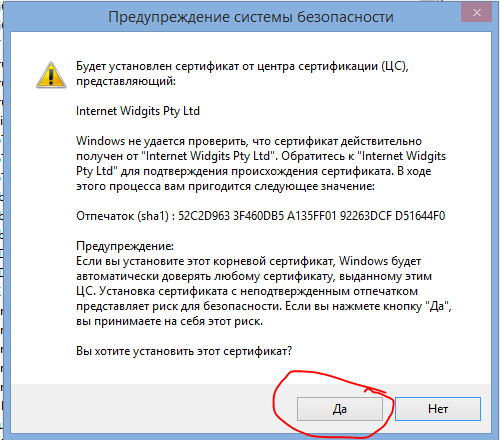
1. Заходим в панель управления и выбираем пункт «Сеть и Интерент»
2. Выбираем пункт «Центр управления сетями и общим доступом» 
3. В вкладке «Изменение сетевых параметров» выбираем пункт «Создание и настройка нового подключения или сети».
4. Выбираем пункт «Подключение к рабочему месту» и жмем «Далее»
5. Выбираем пункт «Использовать мое подключение к Интернету (VPN)»
6. В поле «Адрес в Интернете» вводим IP адрес сервера и нажимаем «Создать»
7. В «Центре управления сетями и общим доступом» вибираем пункт «Изменение параметров адаптера 
8. Щелчком правой кнопки мыши вызываем контекстное меню только что созданного подключения и выбираем пункт «Свойства»



1. В появившемся окне выбираем вкладку безопасность 
2. Тип VPN необходимо указать IKEv2, шифрование обязательное, протокол проверки подлинности EAP-MSCHAPv2, после чего кликнуть «ОК». 
3. На боковой панели управления подключениями находим только что созданное подключение. 
4. Далее кликаем на пункт «Подключить». 
5. Вводим свои учетные данные (логин/пароль) и кликаем на «ОК»
6. Подключение настроено.



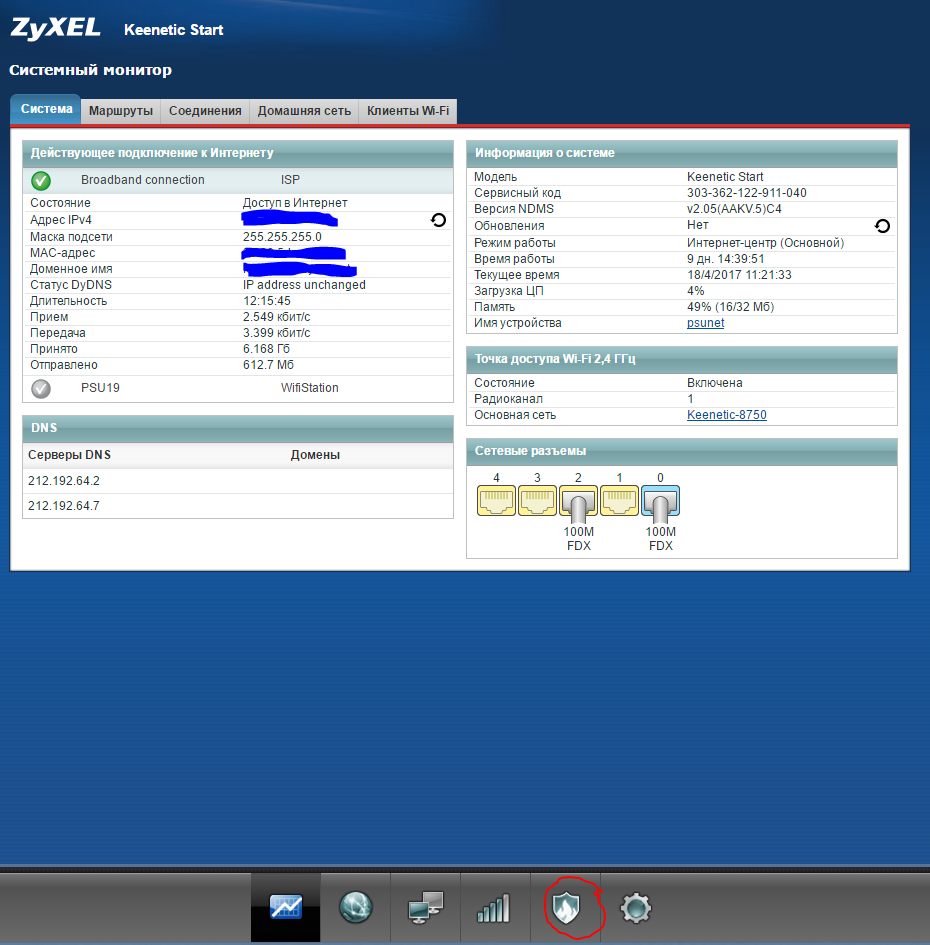
# Приложение 3 «Экспорт CA сертификата в Windows»

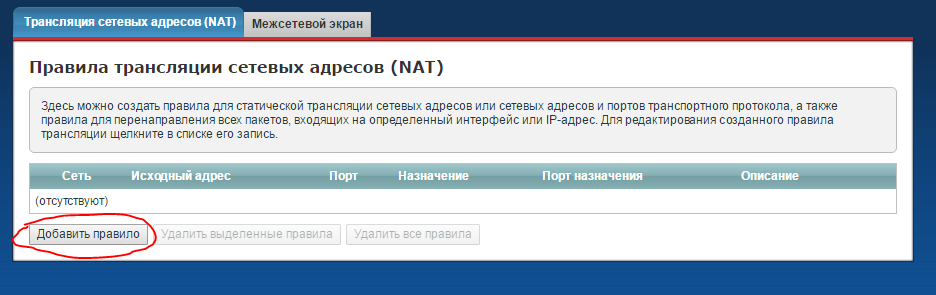
1. Запускаем консоль mcc 
2. В вкладке «Файл» выбираем пункт «Добавить или удалить оснастку» 
3. Выбираем оснастку «Сертификаты» и кликаем «Добавить» 
4. В появившемся окне кликаем на «Готово»
5. Оснастка «Сертификаты» отобразиться на панели выбранных оснасток, кликаем «ОК» 
6. К корне консоли кликаем на вкладку «Сертификаты – текущий пользователь», затем на вкладку «Доверенные корневые центры сертификации», далее щелчком правой кнопки мыши вызываем контекстное меню пункта «Сертификаты», выбираем пункт импорт в списке «Все задачи»
7. Кликаем «Далее». 
8. Кликаем на кнопку «Обзор» 
9. Находим сертификат в проводнике и кликаем «Открыть»
10. Кликаем «Далее» 
11. Вводим пароль от сертификата и кликаем «Далее» 
12. Кликаем «Далее» 
13. Кликаем «Готово» 
14. Соглашаемся С предупреждением системы безопасности 
15. Процесс завершен

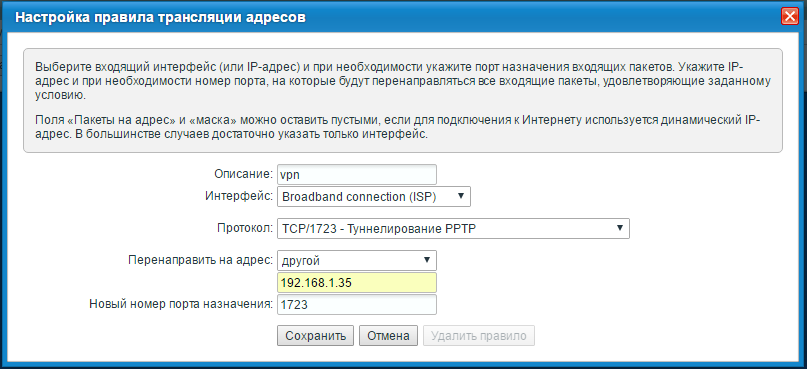
# Приложение 4 «Проброс портов на домашнем роутере»

В качестве устройства используется домашний роутер Zyxel Keenetic Start.

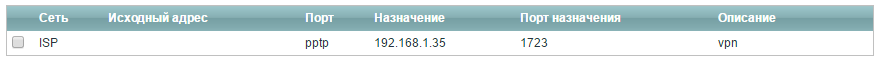
1. Выбираем вкладку безопасность



1. Жмем кнопку добавить правило
2. В поле Описание заносим название для правил, выбираем интерфейс, который смотрит во внешнюю сеть и порт который необходимо пробросить, задаем ip на который необходимо пробросить порт и новый порт назначения. Жмем «Сохранить».



В результате получаем правило



Исходный адрес не задан, так как IP адреса в сети, к которой подключен роутер, назначается динамически. Т.е. необходимо перенаправление с любого адреса.