Автор: Administrator
20.10.10 16:12 - Последнее обновление 20.10.10 16:21

Настройка маршрутизатора на основе Debian/GNU Linux, OpenVPN и NetAMS 3.3.5
(debian linux vpn route netams billing traffic)

Настройка роутера Debian/GNU Linux с учетом трафика на основе NetAMS 3.3.5 и iptables,а также создание ВПН канала для связи двух удаленных друг от друга офисов на базе OpenVPN. Програмное обеспечение. Дистрибутив Debian/GNU Linux 4.0.r3(3 DVD), NetAMS 3.3.5, Webmin 1.410, OpenVPN 2.0_rc16, OpenVPN-2.5.wbm. **Задача.** Есть два офиса(контроллер домена на базе Windows 2003 Server, рабочие станции на базе Windows XP SP2), расположенные на большом расстоянии друг от друга. Необходимо в реальном времени обеспечить доступ удаленному офису к ресурсам главного офиса и наоборот, а именно: 1. Доступ к файлам и принтерам 2. Доступ удаленного офиса к серверу терминалов, т.е. к справочной системе Консультант+ и к программе 1С Бухгалтерия 3. Обеспечить прозрачность удаленной сети для пользователей. Главный офис. В главном офисе находятся: "основной" контроллер домена, все файловые сервера, сервер терминалов и будущий основной роутер Debian/GNU Linux. Т.е. будет только один компьютер на базе Linux и он должен уметь следующее: 1. "Раздавать интернет" пользователям, используя авторизацию и квотирование трафика. 2. Делать возможным взаимодействие удаленных КПК(подключенных к интернету) с программой 1С бухгалтерия. 3. Принимать внешние запросы на ВПН соединение, т.е. - ВПН сервер. Удаленный офис. В удаленном офисе расположены: рабочие станции и второй роутер на базе Debian/GNU Linux. Функции которого: Установка и автоматическая поддержка ВПН соединения с роутером главного офиса.

Установка и настройка роутера в главном офисе.

Рассмотрим более детально, что должен делать главный интернет сервер: 1. Конфигурация NetAMS: а) Авторизация пользователей для доступа в интернет через WEB по паролю без привязки к IP адресу. б) Квотирование трафика. г) Некоторым пользователям разрешено пользоваться только ICQ. д) Учет локального-бесплатного трафика, который не входит в квоты. е) Учет ICQ-трафика(порт 5190). 2. Конфигурация OpenVPN: а) Маршрутизация пакетов от клиента OpenVPN в локальную сеть обеспечивается подключением типа мост TAP интерфеса сервера OpenVPN к сетевому интерфейсу локальной сети. б) Автоматическая установка ВПН клиенту адреса DNS сервера локальной сети. 3. Конфигурация файрвола(политика: то что явно не разрешено - запрещено): а) Запрещена любая установка соединений из вне непосредственно с сервером, кроме ВПН(UDP:1194) и Webmin(TCP:10000) б) Из локальной сети разрешено обращаться напрямую к серверу только(+ пункт а): ssh(порт 22) и 80 порт(авторизация через WEB в NetAMS для получения пользователями локальной сети доступа в интернет). в) Перенаправление(порт форвардинг) запросов из вне на сервер терминалов по порту 5555(взаимодействие с КПК) протокол TCP. г) ДНС запросы(TCP, UDP 53 порт) идут напрямую, минуя NetAMS. д) Разрешена почта только "провайдерная"(бесплатная) и электронная отчетность, и она(почта) так же идет напрямую, минуя NetAMS. Для обеспечения надежности работы организации. В

случае падения NetAMS главная артерия документооборота будет продолжать функционировать в штатном режиме. ж) Пользователям локальной сети после WEB авторизации в NetAMS разрешено обращение в интернет по следующим портам протокола TCP: 80:82,443,1024:65535. з) Разрешено обращение VPN клиентов, пользователей как главного, так и удаленного офисов друг к другу по любым протоколам и портам. Установка системы: ADM Athlon 64 X2 3800, 1024 MB, SATA 160GB HDD, 2 сетевых интерфейса. Устанавливаем только базовую систему. Нам потребуются большие разделы для логов(/var) и для устанавливаемого ПО(/usr), для раздела /home не нужно много места, т.к. не будет пользователей, использующих роутер как файловый сервер. В первую очередь необходимо сконфигурировать сетевой интерфейс, подключенный к интернету, т.к. сразу же будут установлены и настроены обновления безопасности системы. После инсталляции установим пакет mc (Midnight Commander). Сначала добавляем все 3 DVD дистрибутива "в базу" системы установки apt. Добавление DVD дистрибутива в базу:

apt-cdrom add

После вставляем наш DVD в DVD-дисковод и нажимаем ввод. Повторяем для всех 3-х DVD. Установка пакета mc:

apt-get install mc

Далее для удобства будущего удаленного администрирования устанавливаем пакет ssh.

apt-get install ssh

Для того чтобы подключиться к нашему серверу из MS Windows необходима программа putty. В putty: вводим адрес сервера(небходимо сконфигурировать 2й сетевой интерфейс), а так же меняем кодировку(Window-Trancelation) на UTF-8(опционально). Конфигурируем 2-й сетевой интерфейс(/etc/network/interfaces), например:

This file describes the network interfaces available on your system # and how to activate them. For more information, see interfaces(5). # The loopback network interface auto lo iface lo inet loopback # The primary network interface allow-hotplug eth1 iface eth1 inet static address IP адресс провайдера netmask 255.255.255.X network X.X.X.0 broadcast X.X.X.255 gateway X.X.X.X dns-nameservers X.X.X.X X.X.X.X auto eth1 iface eth0 inet static address 192.168.X.X netmask 255.255.255.0 auto eth0

После необходимо перезапустить "сеть", чтобы ОС установила новые IP адреса: /etc/init.d/networking restart Вводим команду ifconfig и проверяем установленную конфигурацию сетевых интерфейсов. И команду route -N, в маршрутах должен быть адрес шлюза провайдера, который указан в файле /etc/network/interfaces. Адреса ДНС серверов находятся в файле /etc/resolf.conf:

nameserver X.X.X.X nameserver X.X.X.X

Теперь необходимо проверить, как все работает, вводим команду ping с параметрами: адрес компьютера в локальной сети и адрес шлюза провайдера, например:

ping 192.168.1.3 ping 82.67.176.1

Пишем скрипт, чтобы обесречить доступ в интернет пользователям локальной сети. Назовем скрип goinet и установим права на запуск, запускаем:

#!/bin/sh INET="eth1" INETIP="X.X.X.X" iptables -F INPUT iptables -F FORWARD iptables -F OUTPUT iptables -P INPUT ACCEPT iptables -P OUTPUT ACCEPT #Для тэстирования NetAMS закомментировать iptables -P FORWARD ACCEPT #Для

тэстирования NetAMS раскомментировать #iptables -P FORWARD DROP #iptables -A FORWARD -j QUEUE iptables -t nat -A POSTROUTING -o $INET -j SNAT --to-source $INETIP echo "1" > /proc/sys/net/ipv4/ip_forward

После запуска этого скрипта на одной из рабочих станций под управлением MS Windows устанавливаем адрес шлюза = адресу нашего Линукс роутера в локальной сети, так же адреса ДНС серверов провайдера, пингуем шлюз провайдера и заходим в интернет. Если все работает, значит можно приступать к установке NetAMS. Настройку файрвола нужно проводить после установки и конфигурирования системы учета трафика и ВПН, чтобы свести к минимуму возможные конфиликты ПО с файрволом. *В случае дублирования IP адреса в локальной сети нашего интернет сервера *с одной из рабочих станций будут непредсказуемые ошибки во всей сети. *Если при пинговании одного из IP адресов локальной сети с роутера возникают *потери пакетов, то одна из возможных причин - это дублирование IP адреса.

Установка и настройка NetAMS 3.3.5 (
http://www.netams.com
).

1.Устанавливаем пакеты необходимые для NetAMS и модули Perl, которые могут понадобиться в случае использования дополнительного ПО для администрирования NetAMS(модуль Perl, Crypt::GeneratePassword, может потребоваться при использовании билинга в NetAMS, http://search.cpan.org/CPAN/authors/id/J/JW/JWALT/Crypt-GeneratePassword-0.03.tar.gz):

apt-get install mysql-server mysql-client apache2 apache2-mpm-prefork libc6-dev libmysqlclient15-dev libapache2-mod-perl2 php5 php5-mysql binutils cpp gcc iptables-dev g++ make libssl-dev libpcap0.8-dev libnet-telnet-perl libdbd-mysql-perl libdbi-perl libcgi-perl libdate-calc-perl libgd-graph-perl

2.Собираем и устанавливаем NetAMS: Дистрибутивный архив имеет имя вида: netams-3.2.10.tar.gz где 3 - номер версии, 2 - номер подверсии и 10 - номер билда. Распаковываем:

tar zxvf netams-3.2.XXXX.tar.gz cd netams-3.2.XXXX

Запускаем сборку:

make

В конечном итоге в каталоге src/ вы должны получить исполняемые файлы: netams, netamsctl, flowprobe, ulog2netflow и ipfw2netflow. Запускаем их установку на место:

make install

3. Создаем конфигурационный файл NetAMS(/etc/netams.cfg):

debug none language ru user name admin real-name Admin password 123 permit all user name netams password 123 permit all user name nawt password 123 permit all

Этими командами настраивается сервис main, причем явно писать "service main" не нужно. Вначале отключается вывод всей отладочной информации - это нужно для уменьшения размера лог-файла. Далее заводятся пользователи системы, имеющие в ней права администратора (permit all). Указанный пароль "123" потом будет храниться в зашифрованном виде.

service server 0 login local listen 20001 max-conn 6

Этими командами настраивается сервис server, который обеспечивает подключение

администратора и скриптов к работающему экземпляру NeTAMS по протоколу telnet. Входящие соединения принимаются только на локальный адрес 127.0.0.1, порт 20001, и возможно не более шести одновременных соединений. Согласно предыдущим строкам подключиться смогут только три пользователя.

service processor lookup-delay 20 flow-lifetime 120 policy name urls target layer7-detect policy name localtrafik target file /etc/localtraf.txt policy name allip target proto ip policy name trafik target proto tcp port 80:82 443 1024:65535 policy name aska target proto tcp port 5190 restrict all drop local pass

Настраивается главный сервис - processor. lookup-delay flow-lifetime задаются параметры, как часто будут проверяться списки юнитов и откладываться записи в базу данных. Следующие параметры задают политики, по которым будет идти учет трафика. Политика urls необходмима для протоколирования запрошенных ссылок(URL)(опционально). localtrafik учитывает бесплатный-локальный трафик, подсети локального трафика берутся из файла /etc/localtraf.txt, например: X.X.X.X /18 X.X.X.X /19 Политика "allip" задает весь IP-трафик, "trafik" - только тот, который идет по портам TCP 80:82 443 1024:65535, "aska" - тот, который использует программа ICQ. Последняя строка определяет, как поступать с пакетами, которые прошли через учет по списку юнитов и совпали (или не совпали) с каким-либо юнитом. Указанная конфигурация пропускает пакеты, которые принадлежат имеющимся в конфигурационном файле юнитам, и не пропускает остальные. Полезно использовать именно указанное сочетание, т.к. это поможет не пускать в сеть "незаконные" компьютеры.

unit group name LAN acct-policy allip trafik aska unit net name local ip 192.168.1.0/24 no-local-pass acct-policy allip aska unit host name server ip 192.168.1.222 no-local-pass acct-policy allip unit net name all ip 0.0.0.0/0 no-local-pass acct-policy all-ip unit user name user1 sys-allow-login description "Скотт" password 123 parent LAN acct-policy allip %localtrafik trafik aska unit user name user2 sys-allow-login password 123 parent LAN acct-policy allip allip %localtrafik trafik aska aska fw-policy aska

Здесь определяются юниты, или учётные объекты. В начале создается группа, которая будет родительской по отношению к включенным в нее юнитам. Затем следует юнит, обозначающий всю подсеть. Далее идут юниты, представляющие отдельные компьютеры. Для юнита local указан также параметр no-local-pass, который заставляет считать нелокальными все пакеты, принадлежащие сети и не описанные для других юнитов - этим мы отсекаем "неизвестные подключения". Если вы задаете последовательность из нескольких политик подсчета трафика подряд, то по умолчанию подсчет ведется для каждой политики и трафик суммируется. Чтобы при совпадении политики дальнейший подсчет прекратился используется break flag [%]. В данном случае будет учитываться общий трафик, в который входит локальный трафик и платный трафик, отдельно будет видно ICQ трафик, который уже входит в платный. Для юнита user2 введен параметр fw-policy aska, то есть весь трафик кроме этой политики будет блокирован для данного юнита. Чтобы разрешить прохождение трафика согласно политикам учета, необходимо убрать параметр fw-policy aska. Указан также пароль, который может быть использован для доступа к индивидуальной статистике в виде HTML-страниц.

storage 1 all

Указывает сервису processor на необходимость сохранять статистику в хранилище,

описанном сервисом storage за номером 1. При этом запить будет идти в обе таблицы одновременно - raw и summary.

service storage 1 type mysql accept all

Определяет хранилище для статистики. Тип хранилища - MySQL, для доступа к базе будут использованы стандартные настройки: имя пользователя root, пустой пароль, работающий на той же машине SQL-сервер (подключение через unix socket). Имя базы данных - netams.

service data-source 1 type ip-traffic source ipq layer7-detect urls

Определяет, каким образом данные о трафике будут попадать в NeTAMS. ip-traffic - данные берутся путем перехвата ip-пакетов из ядра через netfilter. Параметр layer7-detect urls необходим, чтобы можно было протоколировать посещаемые url (Указать политику учета для тех юнитов, которые надо отслеживать).

service monitor 0 monitor to storage 1

Сервис monitor позволяет осуществлять запись данных из заголовков пакетов, относящихся к указанным юнитам. При этом в базе данных сохраняется не только информация о локальном источнике-получателе пакета, размере и времени, но и об удаленной стороне. Для этого необходимо добавить, например, monitor unit user1.

service login 0 storage 1 max-inact 3000 max-abs 3600 min-abs 1800 set-user-ip yes set name user1 password 123 inact 3000 abs 0 set name user2 password 123 inact 3000 abs 0

Сервис login необходим для управления процедурами доступа пользователей через веб-интерфейс. Для работы сервиса необходимо будет указать номер сервиса-хранилища данных. max_inact - максимально допустимая величина времени таймаута неактивности. max-abs - максимально допустимая величина времени абсолютного таймаута. min_abs - минимально допустимая величина времени абсолютного таймаута. Задается в секундах. set-user-ip - Указывает на необходимость в случае успешной авторизации перезаписать IP-адрес юнита (если он имеет тип user) на текущий; при наступлении таймаута или останове доступа адрес сбрасывается в 0.0.0.0. set - раписывает в структуру данных юнита в памяти и одновременно в SQL-базу параметры юнита (определяется по имени или номеру OID). Для настройки параметров сервиса login необходимо подключиться к программе через telnet-интерфейс, перейти в режим настройки сервиса командой service login 0 и ввести команду set name user1 password 123 inact 3000 abs 0. После набираем команды: exit(выйти из упраления сервисом), вводим команду show login.

service quota 0 storage 1 policy trafik set name user2 active month 70M in

policy - задает политику учета (acct-policy), которая будет использоваться при проверке квот. Это политика по умолчанию для всех, существует возможность переопределить ее для конкретного юнита. Если не указано, используется первая политика из определенных policy XXX сервиса processor. Команда set используется аналогично как в сервисе login.

service html 0 path /var/www/netams/stat run 2min client-pages all htaccess yes display-health yes display-top 5

Сервис html позволяет автоматически генерировать HTML-страницы с отчетами. Процесс netams будет автоматически создавать эти страницы раз в 2 минуты(далее нужно увеличить этот параметр) и складывать их в каталог /var/www/traffic. Будет

создаваться как администраторская часть дерева страниц, так и клиентская. Доступ к статистике будет защищен паролем (на администраторскую " admin:123, клиентам " их логины-пароли).

htaccess { yes | no }

Включает и выключает механизм автоматической защиты каталогов с помощью файлов .htaccess и .htpassword. При этом используются пароли администратора NeTAMS (те, которые задаются в "user... crypted..." в начале конфигурационного файла и собственно пароли на юниты ("unit ... password ..."). При этом администратору доступны любые подкаталоги веб-дерева, а пользователям - только их собственные.

client-pages { all | groups | none | group GG1 GG2 ... }

Показывает, будут ли создаваться клиентские страницы для веб-представления статистики: all - будет создаваться все groups - только общая статистика и статистика подкаталоги для юнитов типа "группа" none - только общая статистика group GG1 GG2 ... - клиентские статистики только для перечисленных групп и содержащихся в них юнитах (не рекурсивно). Чтобы добавить или удалить группу в списке, необходимо дать команду с новым списком полностью. display-top N - Включает механизм генерации статических страниц, показывающих TOP N (N - число, желательно порядка 10) потребителей трафика (юниты типа USER и HOST) для периодов времени с начала часа, дня, недели и месяца. display-health { yes | no } - Включает и выключает механизм автоматического отображения "здоровья" системы (аналогично show health), т.е. свободного места на жестком диске и загрузки процессора. По умолчанию-выключено. 4.Запускаем нетамс /etc/init.d/netams.sh start, сам файл:

```
#!/bin/sh daemondir=/usr/local/sbin path_to_etc=/etc case "$1" in start) sleep 3;
/bin/mkdir -p /var/run/netams if [ -x $daemondir/netams ]; then if [ -f
$path_to_etc/netams.cfg ]; then $daemondir/netams -lf $path_to_etc/netams.cfg
> /dev/null && echo -e "33[40;36;1mStart NetAMS daemon...ok33[0m" fi fi
;; stop) killall netams rmdir /var/run/netams && echo -e "33[40;36;1mStop
NetAMS daemon...ok33[0m" ;; *) echo "$0 start | stop" ;; esac
```

4. Подключаемся к NetAMS с помощью telnet:

telnet localhost 20001

Вводим логин admin и пароль прописанный в конфиге /etc/netams.cfg далее вводим команду html, show config, show version, save. Выходим из telnet и перезапускаем NetAMS. (чтобы произошла корректная привязка сгенерированных OID юнитов к базе данных). Снова Подключаемся к NetAMS с помощью telnet. Устанавливаем квоты и логинЫ, из конфига эти параметры в базу не вносятся и если не установить в telnet, то после рестарта NetAMSa они пропадут. Так же после установки всех параметров и начального формирования конфига(можно и позже), необходимо его сохранить командой save в telnet и перезапустить демон NetAMS, иначе возможна неработоспособность сервиса генерирования html страниц. Или прописать в конфигурационном файле:

service scheduler 0 time 2min action "html"

5. Изменяем скрипт, чтобы трафик проходил через NetAMS:

```
#!/bin/sh INET="eth1" INETIP="X.X.X.X" iptables -F INPUT iptables -F FORWARD
iptables -F OUTPUT iptables -P INPUT ACCEPT iptables -P OUTPUT ACCEPT #Для
```

тэстирования NetAMS закомментировать #iptables -P FORWARD ACCEPT #Для тэстирования NetAMS раскомментировать iptables -P FORWARD DROP iptables -A FORWARD -j QUEUE iptables -t nat -A POSTROUTING -o $INET -j SNAT --to-source $INETIP echo "1" > /proc/sys/net/ipv4/ip_forward

6. Добавляем в конфиг апача /etc/apache2/httpd.conf:
<Location /netams> Options Indexes FollowSymLinks MultiViews AllowOverride all Order allow,deny allow from all DefaultLanguage ru AddDefaultCharset WINDOWS-1251 </Location> <Location /netams/cgi-bin> Options Indexes FollowSymLinks MultiViews ExecCGI AllowOverride none Order allow,deny allow from all DefaultLanguage ru AddDefaultCharset WINDOWS-1251 </Location> <Location /netams/stat> Options Indexes FollowSymLinks MultiViews ExecCGI AllowOverride none Order allow,deny allow from all DefaultLanguage ru AddDefaultCharset WINDOWS-1251 </Location> AddHandler cgi-script .cgi ServerName lol <Directory /var/www/netams> Options FollowSymLinks ExecCGI Indexes AllowOverride All </Directory> ServerName rofl <Directory /var/www/login> Options FollowSymLinks ExecCGI Indexes AllowOverride All </Directory>

Для того чтобы можно было через web интерфейс получить доступ к административному и пользовательскому интерфейсу NetAMS, например, по адресу:
http://192.168.1.1/netams
И для получения доступа к сервису login через web:
http://192.168.1.1/login/login.cgi
7. Создаем каталог /var/www/login и в него копируем файлы config.cgi login.cgi netams_api.pl из каталога /var/www/netams/cgi-bin/ 1.создаем каталог /var/www/netams/cgi-bin/ и переписываем туда файлы из каталога cgi-bin дистрибутива NetAMS 2.создаем каталог /var/www/netams/images и /var/www/stat/images 3.переписываем туда файлы из каталога /var/www/netams/cgi-bin/images 4.переписываем файл admintool.cgi из /var/www/netams/cgi-bin/ в /var/www/netams/stat и правим его только в этом каталоге так:
#!/usr/bin/perl # # $Id: admintool.cgi,v 1.29 2004/05/06 10:49:54 jura Exp $ use CGI qw/:standard/; $cgi=new CGI; #$uri=$ENV{"REQUEST_URI"}; $host=$ENV{"SERVER_NAME"}; #$uri=/admintool.cgi/admin/index.cgi/; $uri="/netams/cgi-bin/admin/index.cgi"; $url="http://$host$uri"; #$url="$uri"; print $cgi->redirect($url);
В файле /login/login.cgi прописываем пути, например:
$url_to_stat="/netams/stat/clients/" <img src ="images/logo_small.gif"
В файлах config.cgi(/var/www/netams/cgi-bin/,/var/www/netams/cgi-bin/admin и в /var/www/login) изменяем параметры логина к нетамсу(логин и пароль, прописанные в конфиге netams.conf, т.е. netams 123). А так же(если нужно) устанавливаем:
$statistic_url="/netams/stat/"
8.Создаем каталог /var/www/login/images и переписыаем в него файл logo_small.gif. 9.Копируем файл .htaccess из /var/www/netams/stat/ в /var/www/netams/ 10.Удаляем(если они есть) файлы .htaccess из каталога cgi-bin и cgi-bin/admin. 11.Создаем файл /var/www/netams/index.html:

<html><HEAD><!-- $Id: index.html,v 1.1 2008-01-02 13:31:38 anton Exp $ --> <META http-equiv="Pragma" content="no-cache"> <META http-equiv="Expires" content="-1"> <META http-equiv="Cache-Control" content="no-cache"> <META HTTP-EQUIV="refresh" CONTENT="5;URL=stat/"> </HEAD><title>NeTAMS</title> <body> <table width="770" height="100%" align="center" border=0 cellpadding=10 cellspacing=0> <tr align="center" valign="middle"><td align="center" valign="middle"> <table width=380 cellpadding=0 cellspacing=0 border=0> <tr align=center> <td align=center><img src="images/logo_sm.jpg" width=376 height=60 alt="" border="0">

</td></tr> <tr align=left> <td>
 <h3>You will be redirected soon...</h3>
  Common statistic Administration interface </td></tr> </table> <tr><td></td></tr> </table> </body></html>

12.Перезагружаем апач: /etc/init.d/apache2 restart 13.Пробуем снова с рабочей станции получить доступ в интернет. Не должно пустить. Далее заходим на http://адресроутера/login/login.cgi и вводим логин пароль: user 123 и снова пробуем. Заходим на
http://192.168.1.1/netams
, вводим логин пароль админа. 14.Скрипт загрузки NetAMS при старте системы /etc/init.d/netams.sh:

#!/bin/sh daemondir=/usr/local/sbin path_to_etc=/etc case "$1" in start) sleep 3; /bin/mkdir -p /var/run/netams if [-x $daemondir/netams]; then if [-f $path_to_etc/netams.cfg]; then $daemondir/netams -lf $path_to_etc/netams.cfg > /dev/null && echo -e "33[40;36;1mStart NetAMS daemon...ok33[0m" fi fi ;; stop) killall netams rmdir /var/run/netams && echo -e "33[40;36;1mStop NetAMS daemon...ok33[0m" ;; *) echo "$0 start | stop" ;; esac

Добавляем в автозагрузку:

update-rc.d netams.sh defaults

15. Перезагружаем систему(опционально). При загрузке должен запуститься NetAMS. Далее загружаем тэстовый скрипт для интернета и логинимся через веб, чтобы получить доспут в интернет. Если пользователь был "залогинен" до рестарта, то после перезагрузки у него все равно будет доступ, автоматическое отключение пользователя происходит через 3000 секунд его полной неактивности(задается в конфигурационном файле NetAMS).

Установка и настройка OpenVPN 2.0

Для быстрой настройки OpenVPN используется пакет Webmin 1.410 и модуль к нему OpenVPN-2.5.wbm (
http://www.webmin.com
), а так же пакет bridge-utils: Устанавливаем пакет bridge-utils(пакеты для возможности использования соединения типа мост):

apt-get install bridge-utils

Устанавливаем пакет WebMin:

dpkg -i webmin_1.410_all.deb

После выполнения этой команды, dpkg выдаст много ошибок о нарушенных зависимостях и не хватке многих пакетов в системе. Для завершения установки вводим команду:

apt-get install -f
Далее заходим на https://X.X.X.X:10000/(соглашаемся продолжить), вводим логин пароль (root-xxx) и заходим в Webmin-Change Language and Theme, меняем(опционально) язык на русский, обновляем страницу. Идем в Настройки Webmin-Модули Webmin и загружаем уже скаченный модуль из файла OpenVPN-2.5.wbm, либо в Third party module from, жмем select и выбираем и устанавливаем модуль OpenVPN-admin(будет загружен с сайта
http://www.webmin.com
). После его установки Webmin напишет, где модуль появится. Обновляем страницу, заходим в модуль(Службы-OpenVPN + CA). Жмем настройка модуля и проверяем пути. Необходимо исправить следующее: Server Hint for Clients (*) - адрес VPN сервера, который будет прописан в строке ввода: адрес VPN сервера при создании клиента к нашему серверу. То есть пишем адрес нашего VPN сервера(опционально). Command to start Bridge - /usr/share/webmin/openvpn/br_scripts/bridge_start Command to stop Bridge - /usr/share/webmin/openvpn/br_scripts/bridge_end Path to DOWN-ROOT-PLUGIN - /usr/lib/openvpn/openvpn-down-root.so Жмем сохранить. Далее идем в сертификаты, и генерируем сертификат для нашего сервера и клиентов: Key size (bit) - 1024(при увеличении возрастает нагрузка на систему) Expiration time of - 3650 дней по умолчанию (10 лет) Certification Authority key (days) Сохраняем и генерируем ключи для сервера и клиентов, для каждого клиента свой ключ. Идем в VPN List и создаем сервер: Device - TAP Bridge Device - br0 Network Device for Bridge - eth0(интрефейс локальной сети) IP config for bridge IP-Address/Gateway - IP адрес и маску подсети, которые устнавлены на eth0 IP-Range for Bridge-Clients - диапазон адресов для VPN клиентов management (Enable Management) - Enable порт, например, 2222. Дает возможность видеть информацию о VPN подключениях(опционально). Net IP assigns (option server) - ничего не пишем Additional Configurations - push "dhcp-option DNS 192.x.x.x" (автоматическая установка DNS сервера, возможен повторный вызов для добавления нескольких серверов) Сохраняем. Наш сервер готов к работе. Заходим в Client List и генерируем клиентов. Понадобится как минимум один - для обесепечения связи двух офисов. Экспортируем конфигурационные файлы клиента(ключ для клиента удаленного офиса необходимо сгенерировать без пароля). Получился следующий конфигурационный файл сервера:

port 1194 proto udp dev tap1 ca keys/xxcertificat/ca.crt cert keys/xxcertificat/nvserver.crt key keys/xxcertificat/nvserver.key dh keys/xxcertificat/dh1024.pem server-bridge 192.168.x.x 255.255.255.0 192.x.x.150 192.x.x.170 #@@ br0 eth0 crl-verify keys/xxcertificat/crl.pem ifconfig-pool-persist servers/xxserver/logs/ipp.txt tls-auth servers/xxserver/ta.key 0 cipher DES-CBC user nobody group nogroup status servers/xxserver/logs/openvpn-status.log log-append servers/xxserver/logs/openvpn.log verb 2 mute 20 max-clients 100 management 127.0.0.1 2222 keepalive 10 120 client-config-dir /etc/openvpn/servers/xxserver/ccd client-to-client comp-lzo persist-key persist-tun float ccd-exclusive up servers/nvserver/bin/nvserver.up plugin

/usr/lib/openvpn/openvpn-down-root.so "/etc/openvpn/servers/nvserver/bin/nvserver.down-root"
push "dhcp-option DNS 192.x.x.x"
Никакие маршруты прописывать не нужно, т.к. используется подключение типа мост.

Установка и настройка файрвола (iptables).

Немного теории: когда пакет попадает на наш роутер, сначала он попадает в таблицу mangle цепочка PREROUTING, далее он идет в таблицу nat-PREROUTING, далее, в зависимости от того куда направлен пакет, он идет либо: 1.mangle-INPUT..filter-INPUT..local_routing..mangle-OUTPUT..nat-OUTPUT..filter-OUTPUT..выход либо: 2.mangle-FORWARD..filter-FORWARD..выход потом на выход: mangle-POSTROUTING..nat-POSTROUTING. Интересуют только 2 цепочки таблицы filter: это INPUT - то есть все то, что идет непосредственно на наш сервер и цепочка FORWARD, то есть все то, что идет через роутер, как в локальную сеть, так и из нее. Еще раз конфигурация файрвола(политика: то что явно не разрешено - запрещено): а) Запрещена любая установка соединений из вне непосредственно с сервером, кроме ВПН(UDP:1194) и Webmin(TCP:10000) б) Из локальной сети разрешено обращаться напрямую к серверу только(+ пункт а): ssh(порт 22) и 80 порт(авторизация через WEB в NetAMS для получения пользователями локальной сети доступа в интернет). в) Перенаправление(порт форвардинг) запросов из вне на сервер терминалов по порту 5555(взаимодействие с КПК) протокол TCP. г) ДНС запросы(TCP, UDP 53 порт) идут напрямую минуя NetAMS. д) Разрешена почта только "провайдерная"(бесплатная) и электронная отчетность. Она(почта) так же идет напрямую, минуя NetAMS для обеспечения надежности работы организации. В случае падения NetAMS главная артерия документооборота будет продолжать функционировать в штатном режиме. ж) Пользователям локальной сети, после WEB авторизации в NetAMS, разрешено обращение в интернет по следующим портам,протокол TCP: 80:82,443,1024:65535. з) Разрешено обращение VPN клиентов, пользователей как главного, так и удаленного офисов друг к другу по любым протоколам и портам. файрвол(/etc/fw):

```
#!/bin/sh SERVER1C="192.168.170.5" INET="eth1" NET="eth0"
NET_NET="192.168.170.0/24" INETIP="84.53.199.198" LANIP="192.168.170.222"
ELCOM_SMTP="84.53.200.28" ELCOM_POP="84.53.200.5" IRIDAN="213.87.26.54"
UNPRIVPORTS="1024:65535" DNS_SERVER1="84.53.200.24"
DNS_SERVER2="84.53.199.254" SYSADMIN="192.168.170.3" #Сбрасываем все правила
iptables -F INPUT iptables -F FORWARD iptables -F OUTPUT iptables -t nat -F iptables -t
mangle -F #то что явно не разрешено - запрещено iptables -P FORWARD DROP iptables
-P INPUT DROP iptables -P OUTPUT DROP #Всякий кал бан iptables -A INPUT -p TCP !
--syn -m state --state NEW -j DROP #Явный сброс всех запросов на соединение из инета,
#кроме тех, какие нужны... iptables -A INPUT -d $INETIP -p TCP --syn --dport ! 10000 -j
DROP #Даем Webmin iptables -A INPUT -p TCP --dport 10000 -j ACCEPT #Даем
OpenVPN(и DHCP) iptables -A INPUT -p UDP --dport 1194 -j ACCEPT iptables -A INPUT -p
UDP --dport 67 -j ACCEPT iptables -A OUTPUT -p UDP --dport 68 -j ACCEPT #Порт
TCP:5555 форвардинг iptables -t nat -A PREROUTING -d $INETIP -p TCP --dport 5555 -j
DNAT --to-destination $SERVER1C:5555 #Открываем TCP порты, которые слушать
только из локалки iptables -A INPUT -s $NET_NET -p TCP -d $LANIP --dport 22 -j ACCEPT
#Даем инет серверу(например, для обновлений безопастности) iptables -A INPUT -d
```

$INETIP -p TCP --dport $UNPRIVPORTS -m state --state ESTABLISHED,RELATED -j ACCEPT #Разрешить DNS для сервера и для локалки(почта, аська) iptables -A INPUT -p UDP -s $DNS_SERVER1 -m state --state ESTABLISHED --sport 53 -j ACCEPT iptables -A INPUT -p TCP -s $DNS_SERVER1 --sport 53 --dport $UNPRIVPORTS -m state --state ESTABLISHED,RELATED -j ACCEPT iptables -A INPUT -p UDP -s $DNS_SERVER2 -m state --state ESTABLISHED --sport 53 -j ACCEPT iptables -A INPUT -p TCP -s $DNS_SERVER2 --sport 53 --dport $UNPRIVPORTS -m state --state ESTABLISHED,RELATED -j ACCEPT iptables -A FORWARD -p UDP -s $NET_NET -d $DNS_SERVER1 --dport 53 -j ACCEPT iptables -A FORWARD -p UDP -s $DNS_SERVER1 -d $NET_NET --sport 53 -m state --state ESTABLISHED -j ACCEPT iptables -A FORWARD -p TCP -s $NET_NET -d $DNS_SERVER1 --dport 53 -j ACCEPT iptables -A FORWARD -p TCP -s $DNS_SERVER1 -d $NET_NET --sport 53 -m state --state ESTABLISHED,RELATED -j ACCEPT iptables -A FORWARD -p UDP -s $NET_NET -d $DNS_SERVER2 --dport 53 -j ACCEPT iptables -A FORWARD -p UDP -s $DNS_SERVER2 -d $NET_NET --sport 53 -m state --state ESTABLISHED -j ACCEPT iptables -A FORWARD -p TCP -s $NET_NET -d $DNS_SERVER2 --dport 53 -j ACCEPT iptables -A FORWARD -p TCP -s $DNS_SERVER2 -d $NET_NET --sport 53 -m state --state ESTABLISHED,RELATED -j ACCEPT #Почта только провайдерная и электронная отчетность, iptables -A FORWARD -s $NET_NET -d $ELCOM_POP -p TCP --dport 110 -j ACCEPT iptables -A FORWARD -s $ELCOM_POP -d $NET_NET -p TCP --sport 110 -m state --state ESTABLISHED,RELATED -j ACCEPT iptables -A FORWARD -s $NET_NET -d $ELCOM_SMTP -p TCP --dport 25 -j ACCEPT iptables -A FORWARD -s $ELCOM_SMTP -d $NET_NET -p TCP --sport 25 -m state --state ESTABLISHED,RELATED -j ACCEPT iptables -A FORWARD -s $NET_NET -d $IRIDAN -p TCP --dport 110 -j ACCEPT iptables -A FORWARD -s $IRIDAN -d $NET_NET -p TCP --sport 110 -m state --state ESTABLISHED,RELATED -j ACCEPT iptables -A FORWARD -s $NET_NET -d $IRIDAN -p TCP --dport 25 -j ACCEPT iptables -A FORWARD -s $IRIDAN -d $NET_NET -p TCP --sport 25 -m state --state ESTABLISHED,RELATED -j ACCEPT #Даем возможность впн клиентам использовать файловые ресурсы #локальной сети iptables -A FORWARD -p ALL -s $NET_NET -d $NET_NET -j ACCEPT #Разрешаем КПКшкам обращаться к серверу AplusServer iptables -A FORWARD -p TCP -d $SERVER1C --dport 5555 -j ACCEPT iptables -A FORWARD -p TCP -s $SERVER1C --sport 5555 -m state --state ESTABLISHED,RELATED -j ACCEPT #Разрешаем авторизацию в NetAMS iptables -A INPUT -p TCP -s $NET_NET -d $LANIP --dport 80 -j ACCEPT #Разрешаем интеренет через NetAMS iptables -A FORWARD -p TCP -s $NET_NET --dport $UNPRIVPORTS -j QUEUE iptables -A FORWARD -p TCP -d $NET_NET --sport $UNPRIVPORTS -m state --state ESTABLISHED,RELATED -j QUEUE iptables -A FORWARD -p TCP -s $NET_NET -m multiport --dport 21,80,81,82,443 -j QUEUE iptables -A FORWARD -p TCP -d $NET_NET -m multiport --sport 21,80,81,82,443 -m state --state ESTABLISHED,RELATED -j QUEUE #Внутри демократия iptables -A INPUT -p ALL -s 127.0.0.1 -i lo -j ACCEPT iptables -A INPUT -p ALL -s $LANIP -i lo -j ACCEPT iptables -A INPUT -p ALL -s $INETIP -i lo -j ACCEPT iptables -A OUTPUT -p ALL -d 127.0.0.1 -o lo -j ACCEPT iptables -A OUTPUT -p ALL -s $INETIP -j ACCEPT iptables -A OUTPUT -p ALL -s $LANIP -j ACCEPT iptables -A OUTPUT -p ALL -s 127.0.0.1 -j ACCEPT #Даем torrent и т.п. сисадмину, минуя NetAMS(для локального трафика) iptables -A FORWARD -p TCP -d $SYSADMIN --dport $UNPRIVPORTS -j ACCEPT iptables -A FORWARD -p TCP -s $SYSADMIN --sport $UNPRIVPORTS -j ACCEPT iptables -A FORWARD -p UDP -d $SYSADMIN --dport

$UNPRIVPORTS -j ACCEPT iptables -A FORWARD -p UDP -s $SYSADMIN --sport $UNPRIVPORTS -j ACCEPT #Есть некоторые провайдеры, которые очень не любят, когда одним #подключением пользуется несколько компьютеров, если мы начинаем #устанавливать на все пакеты одно и то же значение TTL, то тем #самым мы лишаем провайдера одного из критериев определения того, #что подключение к Интернету разделяется несколькими компьютерами. #Для примера можно привести число TTL = 64, которое является стандартным #для ядра Linux. iptables -t mangle -A PREROUTING -i $INET -j TTL --ttl-set 64 #Включаем HAT iptables -t nat -A POSTROUTING -o $INET -j SNAT --to-source $INETIP echo "1" > /proc/sys/net/ipv4/ip_forward

Добавляем после отладки наш файрвол в автозагрузку(/etc/rc.local), Устанавливаем права на fw: chmod 711 /etc/fw (полный доступ для владельца и запуск для всех остальных).

Установка и настройка роутера в удаленном офисе.

Система: Celeron 2.4, 1024+256 MB, IDE 20GB HDD, 2 сетевых интерфейса. Выход в интернет с динамическим IP адресом. При объединении сетей двух офисов будет использовано соединение типа мост: интерфейса локальной сети удаленного офиса и TAP интерфейса VPN клиента удаленного офиса. Bridge интерфейсу br0 будут заданы IP адрес и маска подсети диапазона сети главного офиса, то есть получим одну сеть, состоящую из двух подсетей офисов с единым адресным пространством. Взаимосвязь двух удаленных друг от друга офисов может осуществляться по любым протоколам и портам. Понадобятся следующие пакеты: pppoe, pppoeconf, bridge-utils, openvpn, webmin, ssh.

Установка и настройка интернета (PPPoE).

Установка пакета:

apt-get install pppoe

Установка пакета pppoeconf:

apt-get install pppoeconf

Вводим в диалоге, где нужно: логин, пароль от провайдера и т.п, получаем конфигурационные файлы, например: /etc/ppp/chap-secrets:

Secrets for authentication using CHAP # client server secret IP addresses "providerLOGIN" * "providerPASSWORD"

/etc/ppp/resolf.conf:

nameserver X.X.X.X nameserver X.X.X.X

/etc/ppp/peers/dsl-provider:

Configuration file for PPP, using PPP over Ethernet # to connect to a DSL provider. # # See the manual page pppd(8) for information on all the options. ## # Section 1 # # Stuff to configure... # MUST CHANGE: Uncomment the following line, replacing the user@provider.net # by the DSL user name given to your by your DSL provider. # (There should be a matching entry in /etc/ppp/pap-secrets with the password.) #user myusername@myprovider.net # Use the pppoe program to send the ppp packets over the Ethernet link # This line should work fine if this computer is the only one accessing # the Internet through this DSL connection. This is the right line to use # for most people. #pty "/usr/sbin/pppoe -I eth0 -T 80 -m 1452" # An even more conservative version of the previous line, if things # don't work using -m 1452... #pty "/usr/sbin/pppoe -I eth0 -T 80 -m 1412"

If the computer connected to the Internet using pppoe is not being used # by other computers as a gateway to the Internet, you can try the following # line instead, for a small gain in speed: #pty "/usr/sbin/pppoe -I eth0 -T 80" # The following two options should work fine for most DSL users. # Assumes that your IP address is allocated dynamically # by your DSL provider... noipdefault # Try to get the name server addresses from the ISP. usepeerdns # Use this connection as the default route. # Comment out if you already have the correct default route installed. defaultroute ## # Section 2 # # Uncomment if your DSL provider charges by minute connected # and you want to use demand-dialing. # # Disconnect after 300 seconds (5 minutes) of idle time. #demand #idle 300 ## # Section 3 # # You shouldn't need to change these options... hide-password lcp-echo-interval 20 lcp-echo-failure 3 # Override any connect script that may have been set in /etc/ppp/options. connect /bin/true noauth persist mtu 1492 # RFC 2516, paragraph 7 mandates that the following options MUST NOT be # requested and MUST be rejected if requested by the peer: # Address-and-Control-Field-Compression (ACFC) noaccomp # Asynchronous-Control-Character-Map (ACCM) default-asyncmap plugin rp-pppoe.so eth1 user "providerLOGIN"

Файл /etc/network/interfaces:

This file describes the network interfaces available on your system # and how to activate them. For more information, see interfaces(5). # The loopback network interface auto lo iface lo inet loopback # The primary network interface allow-hotplug eth0 iface eth0 inet static address 192.168.1.1 netmask 255.255.255.0 network 192.168.1.0 broadcast 192.168.1.255 # dns-* options are implemented by the resolvconf package, if installed auto dsl-provider iface dsl-provider inet ppp pre-up /sbin/ifconfig eth1 up # line maintained by pppoeconf provider dsl-provider auto eth1 iface eth1 inet manual

Установка и настройка клиента OpenVPN и файрвола. Устанавливаем пакеты bridge-utils, openvpn, webmin, ssh:

apt-get install bridge-utils apt-get install openvpn dpkg -i webmin_1.410_all.deb apt-get install -f apt-get install ssh

В Webmin устанавливаем модуль OpenVPN-2.5.wbm, в данном случае очень вероятно, что роутер удаленного офиса в будущем может быть использован как VPN сервер, так же вместе с модулем будут установлены скрипты:

/usr/share/webmin/openvpn/br_scripts/bridge_start /usr/share/webmin/openvpn/br_scripts/bridge_end

Убираем из автозагрузки OpenVPN:

update-rc.d openvpn remove

Создаем скрипт /etc/ppp/ip-up.d/goinet:

#!/bin/sh cd /etc/openvpn /usr/sbin/openvpn --config nvclient9.conf --daemon --writepid /var/run/openvpn.pid

Теперь после установки соединения с провайдером, автоматически будет устанавливаться VPN соединение, используя конфигурационный файл clientname.conf(наш VPN клиент), например, следующего содержания:

client proto udp dev tap ca ca.crt dh dh1024.pem cert

nvclient7.crt key nvclient7.key remote x.x.x.x 1194 tls-auth ta.key 1 cipher DES-CBC user nobody group nogroup verb 2 mute 20 keepalive 10 120 comp-lzo persist-key persist-tun float resolv-retry infinite nobind #необходимо добавить в конфигурационный файл, #чтобы после установки соединения создавался мост и запускался файрвол up /etc/openvpn/start_bridge

Копируем конфигурационные файлы OpenVPN клиента в /etc/OpenVPN/. (ca.crt, dh1024.pem, clientname.crt, clientname.key, ta.key, clientname.conf и clientname.ovpn - конфигурационный файл для OpenVPN GUI). И если нужно добавляем строку up /etc/openvpn/start_bridge Создаем скрипты: файл /etc/openvpn/start_bridge:

#!/bin/bash #возможны ошибки при инициализации #tap интерфейса из-за этого скрипта и возможно нада будет прямо указать #интерфейс, например tap0 #Адрес TAP интерфейса VPN клиента может измениться, #определяем функцию получения IP адреса с интерфейса function get_addr() { IFCONFIG='/sbin/ifconfig'; HEAD='head -2'; TAIL='tail -1'; CUT='cut -d: -f2'; IP=`$IFCONFIG $1 | $HEAD | $TAIL | awk '{print $2}' | $CUT`; echo $IP; } #Получаем tap интерфейс, он не всегда может быть tap0 VPN=`/sbin/ifconfig | grep ^[t-t] | awk '{print $1}'` VPNIP=`get_addr $VPN` /usr/share/webmin/openvpn/br_scripts/bridge_start --setbr br=br0 eth=eth0 tap=tap0 ip=$VPNIP netmask=255.255.255.0 > /dev/null #запускаем файрвол после успешной или неуспешной установки VPN соединения /etc/openvpn/fw-server2 #end

файл /etc/ppp/ip-down.d/stop_bridge: #!/bin/bash #возможны ошибки при инициализации #tap интерфейса из-за этого скрипта и возможно нада будет прямо указать #интерфейс, например tap0 VPN=`/sbin/ifconfig | grep ^[t-t] | awk '{print $1}'` /usr/share/webmin/openvpn/br_scripts/bridge_end --killbridge --seteth br=br0 eth=eth0 tap=$VPN ip=192.168.1.1 netmask=255.255.255.0 > /dev/null Файл /etc/openvpn/fw-server2:

#!/bin/sh #IP адрес ppp0 интрефейса динамически назначается провайдером и #IP адрес TAP интерфейса VPN клиента может измениться, то #определяем функцию получения IP адреса с интерфейса function get_addr() { IFCONFIG='/sbin/ifconfig'; HEAD='head -2'; TAIL='tail -1'; CUT='cut -d: -f2'; IP=`$IFCONFIG $1 | $HEAD | $TAIL | awk '{print $2}' | $CUT`; echo $IP; } #Получаем PPP интерфейс, он не всегда может быть ppp0 INET=`/sbin/ifconfig | grep ^[p-p] | awk '{print $1}'` NET="br0" NET_NET="192.168.170.0/24" INETIP=`get_addr $INET` LANIP=`get_addr $NET` UNPRIVPORTS="1024:65535" DNS_SERVER1="84.53.200.24" DNS_SERVER2="84.53.199.254" #Сбрасываем все правила iptables -F INPUT iptables -F FORWARD iptables -F OUTPUT #Политика по умолчанию - все запрещено iptables -P FORWARD DROP iptables -P INPUT DROP iptables -P OUTPUT DROP #Явный бан всех запросов на соединение из инета #кроме Webmin iptables -A INPUT -d $INETIP -p TCP --syn --dport ! 10000 -j DROP #Даем Webmin iptables -A INPUT -p TCP --dport 10000 -j ACCEPT #Даем OpenVPN(и DHCP для OpenVPN клиента) iptables -A INPUT -p UDP --sport 1194 -j ACCEPT iptables -A INPUT -p UDP --dport 68 -j ACCEPT iptables -A OUTPUT -p UDP --dport 67 -j ACCEPT #Открываем TCP порты которые слушать только из локалки iptables -A INPUT -s $NET_NET -p TCP -d $LANIP --dport 22 -j ACCEPT

#Даем инет серверу(для обновлений безопастности и т.п.) iptables -A INPUT -d $INETIP -p TCP --dport $UNPRIVPORTS -m state --state ESTABLISHED,RELATED -j ACCEPT #Разрешить DNS для сервера iptables -A INPUT -p UDP -s $DNS_SERVER1 -m state --state ESTABLISHED --sport 53 -j ACCEPT iptables -A INPUT -p TCP -s $DNS_SERVER1 --sport 53 --dport $UNPRIVPORTS -m state --state ESTABLISHED,RELATED -j ACCEPT iptables -A INPUT -p UDP -s $DNS_SERVER2 -m state --state ESTABLISHED --sport 53 -j ACCEPT iptables -A INPUT -p TCP -s $DNS_SERVER2 --sport 53 --dport $UNPRIVPORTS -m state --state ESTABLISHED,RELATED -j ACCEPT #Обеспечиваем сетевой доступ обоих офисов друг к другу iptables -A FORWARD -p ALL -j ACCEPT #Внутри демократия iptables -A INPUT -p ALL -s 127.0.0.1 -i lo -j ACCEPT iptables -A INPUT -p ALL -i lo -j ACCEPT iptables -A INPUT -p ALL -i lo -j ACCEPT iptables -A OUTPUT -p ALL -j ACCEPT #Включаем IP forwarding echo "1" > /proc/sys/net/ipv4/ip_forward

Еще раз кратко, что нужно сделать: 1. Установить PPPoE соединение с провайдером и получить доступ в интернет. 2. Установить пакеты для VPN клиента. 3. Переписать и, если нужно, дополнить конфигурационный файл clientname.conf. 4. Создать/переписать файлы: /etc/openvpn/start_bridge /etc/ppp/ip-down.d/stop_bridge /etc/openvpn/fw-server2 Получился следующий механизм: Установка интернет соединения->Установка VPN соединения->организация подключения типа мост->запуск файрвола. Разрыв интернет соединения->остановка соединения типа мост. P.S. При доступе из удаленного офиса на сервер терминалов в программу 1С могут возникнуть временные простои при печати больших объемов на локальный принтер. Это связано с пропускной способностью исходящего трафика на канале ADSL.

Возможно два решения этой задачи: 1. Печать в файл(необходима установка драйвера-эмулятора принтера на сервере терминалов, например, Zan_Image_Printer_v5.0.2-Primax и настройка печать в файл типа TIFF) на локальную машину. Общая скорость печати больших объемов увеличится в 4-5 раз. 2. Установка SDSL технологии для доступа в интернет. 3. Комбинировать первый и второй пункт при очень больших объемах печати. Пример скрипта для восстановления связи после обрыва: #!/bin/sh sovok=0 PIDFILE="/var/run/openvpn.pid" ping -c 3 x.x.x.x > /dev/null sovok=$? if [$sovok != 0]; then poff #для VPN клиента, сервер можно не перезапускать kill `cat $PIDFILE` || true rm $PIDFILE # pon dsl-provider fi Добавляем его в /etc/crontab: */3 * * * * root /полный путь до скрипта раз в 3 минуты

оригинал: http://77.234.201.242/base/net/debian_netams.txt.html

{jcomments on}