Мне давно знаком дистрибутив для быстрого и удобного развертывания шлюза с удобной web панелью для управления. Я решил описать установку и настройку clearos в бесплатной редакции community для организации шлюза и прокси сервера. Данный продукт вполне подойдет для малого и среднего офиса со стандартными требованиями к маршрутизатору.

Содержание:

Введение

Что такое ClearOS? Если кратко, то это программный шлюз на базе операционной системы CentOS для малого и среднего бизнеса, в случае необходимости и для домашнего использования. Все управление происходит через браузер в удобной web панели. Освоить настройку и управление не трудно даже если

вы вообще ничего не знаете о linux. Подробнее о системе можно почитать на официальном сайте. Приведу оттуда любопытное видео, мне понравилось.

Watch this video on YouTube

Теперь расскажу своими словами что это такое и чем оно мне нравится:

1. **Во-первых**, clearos действительно легко настраивается. Я не скажу, что прям каждый сможет это сделать, какое-то понимание в работе шлюзов должно быть, но знаний непосредственно линукса может не быть совсем. Через браузер реально все настроить и заставить работать. Функционал достаточный для среднестатистического офиса.
2. **Во-вторых**, в clearos все сделано удобно в том плане, что если зайти на сервер по ssh, то увидите обычные конфиги привычного софта — squid, iptables, dnsmasq и другие. Сверху у вас обертка из веб панели, а внутри привычные конфиги. Панель не уродует их, не прячет, не приводит к неудобочитаемому виду. Если вы неплохо разбираетесь в centos, то без проблем сможете настроить в консоли все, что вам нужно. Например, меня не устраивал функционал openvpn, реализованный через стандартное управление. Я просто зашел в консоль и настроил так, как мне нужно. Конечно, я потерял возможность управлять ей через web панель, но тем не менее смог получить необходимый функционал. Это удобно еще и потому, что в панели управления очень мало настроек, большинство остается по-умолчанию и не настраивается. Если вы понимаете что делаете и хотите узнать больше, вы просто смотрите конфиг программы через консоль.
3. **В-третьих**, внутри у вас обычная CentOS. Вы с ней можете делать все, что делаете с обычным сервером. Установить новые пакеты, посмотреть другие настройки. Главное не делать то, что мешает работе веб панели управления.
4. **В-четвертых**, она легко бэкапится. Я проверял несколько раз. Есть нюансы, о которых нужно знать. Я расскажу об этом ниже. Но в общем и целом, можно без лишних телодвижений забэкапить настройки сервера и восстановить их на другом железе.

Теперь о минусах, с которыми я сам сталкивался. На всякий случай нужно отключать автоматическое обновление. Один раз после обновления перестал работать dnsmasq, который выступал в роли dns и dhcp сервера в локальной сети. Была небольшая ошибка в конфиге, которая возникла после обновления. Для меня не составило никакого труда зайти в консоль сервера, посмотреть логи, понять и исправить ошибку. Но для офиса, где не было системного администратора linux это обернулось серьезными проблемами и перебоями в работе сети. Все, что получало сетевые настройки по dhcp не работало некоторое время.

Вторым минусом, хотя это и не совсем минус, является маленькое количество настроек. Я понимаю, что это сделано для простоты как для самих пользователей, так и создателей дистрибутива. Чем больше возможностей, тем труднее поддерживать и настраивать. Простой пример — авторизация в openvpn сделана по сертификату и паролю. Мне, например, достаточно только авторизации по сертификату, но стандартный функционал не позволяет так сделать. Либо на прокси списки контроля доступа возможны только при авторизации по имени пользователя, по ip сделать не получится, хотя последнее мне кажется удобнее.

Но в общем и целом продукт годный, можно пользоваться, отключив на всякий случай автообновление.

Установка ClearOS

Прежде чем установить, нам надо скачать clearos. Зайдя на сайт, может показать неочевидным что и где качать. На сегодняшний день существуют 3 редакции clearos:

1. **Community** — бесплатная версия с базовым функционалом, платные пакеты для расширения функционала можно покупать отдельно.
2. **Home** — для домашнего использования. Стоит 3 доллара в месяц. У нее свои пакеты в магазине для этой редакции.
3. **Business** — расширенная версия со своим магазином и пакетами. Стоит 9 долларов в месяц.

Нас интересует установка clearos в редакции Community. Скачать ее можно по ссылке, выбрав соответствующую редакцию. Про саму установку рассказывать нечего, так как используется стандартный установщик centos. Там все идентично стандартной установке centos. Обязательно настройте сетевое подключение во время установки clearos. Без него невозможна дальнейшая настройка.

Качайте, устанавливайте и начинайте настройку.

Начальная настройка шлюза

После установки, первый запуск нас встречает информационной страницей:

Здесь же сделаем первоначальную настройку сетевых интерфейсов. Жмем на ссылку **Network Console**. Заходим под учеткой root и настраиваем сеть.

eth0 192.168.1.101	В моем случае это внешний интерфейс, по которому я получаю интернет
eth1 10.0.0.1	Внутренний интерфейс, который будет подключен к локальной сети

Для продолжения настройки clearos необходимо в браузере открыть страницу по ip, указанному в установке. В моем случае это *https://192.168.1.101:81*.

При входе браузер предупредит о недоверенном сертификате. Так и должно быть, все равно переходите на страницу. Вас встречает окно логина в систему.

Вводите указанную при установке учетную запись root. Язык рекомендую везде использовать английский. Во-первых, перевод на русский не очень понятный, во-вторых, все инструкции на английском языке. В рунете нет документации по clearos. Основной источник информации — база знаний и форум на официальном сайте.

Я не буду приводить описание всех шагов настройщика. Остановлюсь только на ключевых. Вам зададут вопрос про **Network Mode**. Мы настраиваем шлюз, поэтому выбираем **Gateway Mode**.

Сеть уже настроили. Если ничего менять не надо, то двигаетесь дальше. Потом указываем dns сервер либо вручную, либо, если настройки по dhcp получали, то он уже будет прописан. Дальше выбираете редакцию, в нашем случае **Community**. Затем надо зарегистрировать установку. Вам нужна учетная запись на сайте clearos. Идите и зарегистрируйтесь. Данные от этого аккаунта нужно будет ввести в регистрации установки.

На следующем этапе устанавливаются обновления. Это не так быстро, придется подождать минут 5-10, пока все установится. Хотя не факт, зависит от скачанного дистрибутива. Дальше указываем имя домена и сервера. Можете писать все, что угодно. Если сервер будет смотреть в интернет и есть доменное имя, можно реальное указать. Если не хочется, то придумайте что-то вроде gate.local.

Потом идет установка времени и самое интересное — выбор дополнительных пакетов. Пока ничего не выбирайте. Будем ставить все необходимое по мере настройки функционала, поэтому выбирайте **Skip Wizard**.

На этом все, начальная настройка clearos завершена. Можете настроить dashboard так, как вам нравится. Выбирать особо не из чего, поставьте те виджеты, что есть в наличии. Давайте сразу включим **DHCP сервер**. Идем в раздел Network -> Infrastructure -> DHCP Server и включаем его на локальном интерфейсе:

Настройки можно оставить по-умолчанию, только выбрать диапазон ip, из которого будут выдаваться адреса. Я выбрал 100-254.

В таком виде уже можно подключать устройства в сеть. Они будут получать ip адреса и иметь доступ в интернет.

Настройка firewall

Продолжим настройку clearos настройкой firewall. Изначально у нас уже есть возможность настраивать блокировку входящих соединений сервера. Но этого мало. Нам понадобится как минимум проброс портов и создание более сложных правил блокировки. Для этого нам необходимо установить несколько дополнительных пакетов из Marketplace. Идем туда и выбираем:

- **Custom Firewall** для создания любых правил iptables.
- **Port Forwarding** для проброса портов.

После установки в разделе firewall есть 3 пункта:

- Custom Firewall
- Incoming Firewall
- Port Forwarding

По-умолчанию у нас есть одно входящее правило во фаерволе, которое разрешает подключение к web панели управления по 81 порту. Но, допустим, вам надо ограничить доступ по этому порту и разрешить только с определенных адресов. Стандартный функционал раздела **Incoming Firewall** не позволяет это сделать. Нам необходимо написать полное правило для iptables и поместить его в **Custom Firewall**. Подробнее о написание правил для фаервола рассказано в отдельной статье, посвященной настройке iptables. Я здесь не буду останавливаться на написании правил. Это отдельная тема. Приведу только пару примеров.

Но сразу хочу предупредить. С правилом доступа к серверу экспериментировать не советую. Сам я с первой попытки отключил себе доступ к серверу по

81-му порту. И вот я сижу, сервер передо мной, но я не могу ничего сделать. Через браузер зайти не могу в панель управления. Через консоль сервера никаких вариантов нет, перезагрузка не помогает, ssh по-умолчанию выключен. Для админа, который не разбирается в линуксе это тупик, ничего не сделать, надо переустанавливать сервер.

Я конечно же решил проблему. Открыл на сервере вторую консоль через Alt+F2, зашел root'ом в систему. Посмотрел какие сейчас действуют правила iptables.

```
# iptables -L -v -n
```

Сбросил их.

```
# iptables -F
```

И сделал фаервол открытым.

```
# iptables -P INPUT ACCEPT
# iptables -P OUTPUT ACCEPT
```

После этого снова подключился к серверу через браузер.

Сразу уточню, что по-молчанию в firewall заблокировано все, что не разрешено явно. То, что в разделе Incoming Firewall указано как открытый входящий порт 81, на сервере выглядит немного не так. Открыт не только входящий порт, но и исходящий. Я же в ручном правиле открыл только входящий порт, не указав исходящий. В итоге доступ к серверу потерял. Не повторяйте моих ошибок. С фаерволом работайте очень аккуратно и без особой надобности не трогайте правило, которое разрешает подключение к серверу по 81-му порту.

Вот и примеры. Разрешаем доступ к ssh только с адреса 192.168.1.112.

```
iptables -A INPUT -s 192.168.1.112 -d 192.168.1.101/32 -p tcp -m tcp --dport 22 -j ACCEPT
iptables -A OUTPUT -s 192.168.1.101/32 -d 192.168.1.101/32 -o eth0 -p tcp -m tcp --sport 22 -j ACCEPT
```

Добавляете эти правила в **Custom Firewall** и сможете подключаться по ssh, предварительно запустив сервис в разделе Network -> Infrastructure -> SSH

Server.

Вот другой пример. Вам надо какому-то ip адресу полностью запретить доступ в интернет. Делаете правило в Custom Firewall для ip 10.0.0.15.

```
iptables -I FORWARD 1 -s 10.0.0.15 -j DROP
```

Щелкнут мышкой по правилу, вы можете увидеть его целиком.

Добавлять можете какие угодно правила. Синтаксис идентичен обычному синтаксису для iptables.

Теперь настроим проброс порта. Это сделать очень просто. Допустим, у нас есть web сервер по адресу 10.0.0.5. Нам надо настроить forward 80-го порта. Идем в раздел Network -> Firewall -> Port Forwarding и создаем новое правило, выбрав **Add by: Port**.

И сохраняете. Больше вам ничего делать не надо. Все необходимые правила будут созданы автоматически. Можно посмотреть на сервере по ssh,что это за правила:

С настройкой firewall в clearos закончили. Обозначил основные функции.

Настройка proxy сервера

Теперь рассмотрим настройку proxy сервера на clearos. Идем в Marketplace и ставим пакеты **Web Proxy Server**, **Content Filter Engine** и **Filter and Proxy Report**. Первый это прокси сервер squid, второй — контент фильтр dansguardian. Прокси сервер может работать в одном из трех режимов:

1. **Transparent Mode + No User Authentication**. В этом режиме все http запросы из локальной сети автоматически перенаправляются на порт 3128 прокси сервера с помощью правила на фаерволе. При данной настройке на компьютерах пользователей не нужно делать никаких настроек. Выход в интернет для пользователей полностью прозрачен. Существенный минус такого режима — невозможно проксировать https соединения. А их сейчас все больше и больше. Вы ни статистику по ним не сможете увидеть, ни заблокировать. С учетом того, что сейчас все популярные сайты перешли на https, данный режим работы прокси бесполезен.
2. **Non-Transparent + No User Authentication**. В данном режиме в браузере пользователя необходимо в обязательном порядке установить адрес прокси сервера, иначе доступ в интернет будет закрыт. Пользователи авторизуются прозрачно, доступ в интернет у всех один и тот же. Возможно блокировать доступ как к обычным, так и https сайтам. Минус этого режима в том, что нет возможности настроить разный доступ разным группам пользователей. Можно либо всем все закрыть, либо всем открыть.
3. **Non-Transparent + User Authentication**. В этом режиме больше всего настроек. Выход в интернет осуществляется с авторизацией по имени пользователя. Можно настраивать списки доступа для различных групп пользователей. Как и в предыдущем режиме необходима настройка прокси сервера в свойствах браузера. Главным минусом этого режима — после запуска браузера необходимо вручную ввести имя пользователя и пароль для доступа в интернет.

С такими вводными лично для меня является приемлемым только второй вариант работы прокси, на нем я и остановлюсь. Без фильтрации https трафика смысла в прокси нет вообще, а вводить вручную каждый раз пароль логин и пароль, я считаю издевательством над пользователями.

Идем в раздел Gateway -> Content Filter and Proxy -> Web Proxy Server, выбираем режим работы **Non-Transparent + No User Authentication**. После сохранения страница настроек выглядит следующим образом.

Для запуска Proxy не забудьте нажать Start. После этого прокси сервер будет запущен с заданными параметрами. В данном случае доступ в интернет будет как без прокси, так и с прокси. Если у пользователя в браузере указан адрес прокси сервера, то он будет выходить через прокси, будет логироваться вся его активность в интернете. Если прокси не указан, то доступ в интернет будет прямой. Может кому-то понадобится именно такой режим, можно на этом остановиться. Если мы хотим блокировать доступ к определенным сайтам, то продолжаем настройку.

Блокировка сайтов в ClearOS

Перейдем к запретам. Открываем Gateway -> Content Filter and Proxy -> Content Filter Engine. Нажимаем **Configure Policy** для добавления блокировки сайтов.

Открываем на редактирование список **Banned Sites** и добавляем туда адрес сайта. В моем примере это будет популярная социальная сеть vk.com.

Сохраняем настройки и запускаем модуль.

Теперь доступ в интернет будет заблокирован всем, у кого в настройках браузера не установлен прокси сервер. При попытке открыть любую страницу, пользователь будет получать сообщение.

Добавляем, как указано, адрес proxy server в браузер.

Теперь интернет заработал, а на заблокированный сайт зайти не получится. Браузер будет сообщать, что невозможно открыть страницу. По идее, пользователя должно перенаправлять на страницу с сообщением о том, что сайт заблокирован. Но этого функционала нет, либо я не понял, как его настроить. Сайт просто блокируется без перенаправления и какой-либо информации. Просто недоступен и все. Факт блокировки доступа отражается в лог файле модуля *dansguardian/access.log*. Посмотреть его можно в разделе Reports -> Log Viewer, выбрав соответствующий лог и сделав поиск по слову DENIED.

Посмотреть отчеты по какому адресу кто куда ходил можно в разделе Reports -> Filter and Proxy Report, конкретно отчет **Top IPs**. Там будет список всех IP адресов, которые выходили в интернет с помощью proxy. При нажатии на IP, открывается подробная статистика по нему. Ответ вообще не информативен. Сортировка по размеру выглядит странно, смотрите сами.

Понять из этого отчета, куда пользователь ходил чаще у меня не очень получается. Все как-то криво и не наглядно. К слову, в предыдущих версиях clearos интерфейс был более удобный. Все было мельче и более наглядно, так как умещалось больше информации. Ну а тут что есть, то есть.

Настроек в модуле фильтрации достаточно много, можно самому посмотреть, почитать в документации. Долго все расписывать. В раздел **Exception IPs** можно добавить ip адреса сайтов, на которые всегда будет доступ, в **Banned IPs** наоборот, заносятся адреса, на которые доступ нужно запретить. Если вы будете использовать авторизацию по имени пользователя, то в настройках модуля можно создать различные политики для разных групп пользователей. Кому-то все разрешить, кому-то частично. Я тестировал этот функционал, он работает. Нужно только настроить модуль Directory Server в разделе System -> Account Manager и в нем создать пользователей с группами. Но мне показалось неудобным вводить логин с паролем для выхода в интернет, поэтому я не стал акцентировать внимание на этом режиме.

Настройка vpn на базе openvpn

Настроим еще одну важную функцию — vpn сервер. Тут мне в целом все понравилось. Не так много решений, где можно без проблем, удобно и быстро создавать сертификаты для пользователей openvpn и сразу же из web интерфейса забирать их с файлом конфигурации. Идем в Markeplace и ставим пакет **OpenVPN**. Открываем раздел Network -> VPN -> OpenVPN и начинаем настраивать.

Первым делом нам предложат сконфигурировать сертификаты для сервера. Сделаем это.

В поля для ввода можно писать все, что угодно. Принципиального значения это не имеет.

После создания сертификатов снова возвращаемся в раздел OpenVPN и разрешаем создать в фаерволе необходимые правила.

В общем-то все, теперь справа жмем Start и запускаем модуль. Нам нужно создать пользователя для доступа по vpn. Идем в раздел System -> Account Manager и жмем **Install and initialize Built-in Directory**.

Ждем окончания процесса, потом идем в раздел Accounts и создаем там пользователя.

Выходим из панели управления под пользователем root и логинимся под только что созданным пользователем. После входа нам нужно будет еще раз подтвердить пароль. Делаем это.

В правом верхнем углу нажимайте на имя пользователя и выбирайте User Certificates.

Вводите пароль и создавайте сертификат. Теперь его надо скачать. Вам нужны 3 файла:

- Certificate
- Certificate Authority
- Private Key

Ниже выбирайте тип системы, для которой хотите скачать конфигурационный файл openvpn и скачивайте его. Я буду показывать на примере Windows. В конфигурационном файле сразу же отредактируйте параметр remote. Если у вас там не указан реально существующий домен, то его необходимо заменить на внешний ip адрес сервера clearos. В моем случае строка выглядела так:

```
remote gate.local 1194
```

Я заменил на

```
remote 192.168.1.101 1194
```

Напомню, что в моем случае это внешний ip адрес, который смотрит в интернет. Так как это не публичный ip адрес, мне нужно будет пробросить порт 1194 еще на вышестоящем роутере. Но у вас скорее всего это будет уже внешний IP адрес, если вы настраиваете шлюз в интернет.

Конфигурационный файл вместе с сертификатами копируйте в папку *C:\Program Files\OpenVPN\config*, запускайте openvpn и пробуйте подключиться. Вы должны подключиться к серверу и увидеть все компьютеры в локальной сети за шлюзом. Во время подключения вам нужно будет ввести логин и пароль созданного пользователя vpn.

Если не подключается, то в первую очередь проверьте, запущен ли модуль openvpn, затем создано ли правило на фаерволе, разрешающее входящие подключения по порту 1194.

Каждому клиенту будет назначаться один и тот же ip адрес в туннеле. По-умолчанию, туннель образует подсеть 10.8.0.0/24. Конфиг сервера можно подсмотреть в консоли в файле */etc/openvpn/clients.conf*. Менять его через консоль не рекомендую, он будет снова изменен при перезапуске сервиса. Полезно просто узнать остальные настройки. Через веб панель их не видно.

В общем и целом все нормально работает. OpenVPN завелся без плясок с бубном, что, считаю, неплохо. Функционал достаточный, пользоваться удобно.

Backup настроек

Расскажу про backup настроек clearos, так как есть некоторый опыт. Существуют 2 пакета для бэкапа:

1. **Configuration Backup and Restore**. По-умолчанию уже установлен в системе.
2. **Baremetal Backup and Restore**. Ставится отдельно из магазина.

Первый позволяет просто сохранить текущие настройки в файл и скачать файл на компьютер. Второй может работать по расписанию и регулярно сохранять backup на флешку. Сам бэкап из себя представляет список пакетов и файлов конфигурации к ним. В общем и целом он нормально работает, я восстанавливал системы, но есть нюанс.

Backup clearos будет работать, только если вы восстанавливаете настройки на ту же версию системы. А версия системы достаточно часто меняется. Простой пример. Вы установили систему и настроили бэкап. Пол года все было нормально, вы регулярно делали бэкапы. Но сервер неожиданно ломается. Вы скачиваете образ с сайта, устанавливаете его, накатываете бэкап, а он вам сообщает, что не может восстановить настройки, так как версия системы отличается.

На ум приходит следующий вариант. Вы ставите сервер и сохраняете образ диска, с которого его ставили. Когда надо восстановить настройки, вы ставите сервер из старого образа и думаете, что все будет нормально. Но опять засада. Во время установки сервер скачивает последние обновления. Этот процесс нельзя ни отменить, ни пропустить!!! Такая вот загвоздка. Выход только один — держать постоянно актуальную версию системы. Но как я уже говорил, иногда могут возникнуть проблемы после обновления. Хотя у меня они реально были только один раз. Незначительная ошибка, которую я быстро исправил. Но тем не менее. Имейте ввиду такую особенность и решите, как вам лучше бэкапить сервер. Если это будет виртуальная машина, то бэкапить лучше на уровне виртуалки. Как установить шлюз на виртуальную машину, я рассказывал на примере настройки proxmox.

Не буду подробно расписывать, как сделать сам бэкап. Достаточно зайти в меню модуля и кликнуть пару раз мышкой. Там нет ничего сложного.

Дополнительные возможности

Упомяну еще несколько полезный бесплатных модулей, которые я сам проверил и мне они показались полезными:

1. **Bandwidth and QoS Manager**. В работе я его не проверял, но поставил и посмотрел настройки. Ничего сложного нету, думаю он будет работать. Позволяет настроить приоритизацию трафика и ширину канала по ip адресам.
2. **Network Map**. Регулярно сканирует сеть и обнаруживает устройства. Позволяет вручную описать устройство на основе его MAC.
3. **Network Report и Network Visualizer**. Строят отчеты и отображают текущую сетевую загрузку по интерфейсам. Не сильно информативно, но лучше, чем ничего.
4. **NTP Server**. Сервер времени, по которому другие компьютеры в сети могут синхронизировать время. Полезно, когда вся сеть имеет одно и то же время.
5. **Services**. Отображает список сервисов на сервере. Позволяет добавлять или исключать их из автозагрузки.

Я устанавливал и проверял работу почтового сервера. У меня все получилось настроить через веб панель. Каких-то проблем не возникло. Установил пакет, добавил пользователей, настроил подключение клиентом по imap. Не понравилось то, что нет бесплатного пакета с web интерфейсом. Roundcube стоит 25 долларов. В принципе, немного, если вам действительно нужен почтовый сервер с веб интерфейсом. Но лично я считаю, что почтовый сервер должен быть отдельной ролью с очень гибкими настройками. Это мое мнение, я просто умею настраивать почтовые сервера. Если у вас такого навыка нет, вам возможно подойдет почтовый сервер на clearos. Настроить его не трудно.

ClearOS может выступать в роли контроллера домена, но не уверен, что это будет нормально работать. Самба еще далека от идеала в качестве контроллера. Есть функционал файлового сервера. Но лично я всегда рассматривал ClearOS как удобный шлюз в интернет, который можно настроить мышкой. Остальной функционал не использовал. Не люблю системы, где все в одном. Предпочитаю их разделять.

Заключение

Я рассмотрел только основные возможности ClearOS, но даже это получилось очень объемным материалом. В целом, система дружелюбная, разобраться можно просто пробуя настройки и проверяя изменения. Я рекомендую эту систему тем, кто непременно хочет программный шлюз, но при этом не очень разбирается в настройках. Какое-то понимание, конечно, необходимо, но тут даже методом тыка можно что-то получить. Настроек очень мало, основная их часть от пользователя скрыта. В этом вижу основное преимущество данной сборки.

У меня есть опыт нескольких лет использования ClearOS. Ставил не я, досталось в наследство, так и познакомился. Система работает и свои функции выполняет. Что-то я в консоли сам донастраивал, когда было нужно. Например, соединил две ClearOS с помощью OpenVPN по схеме сервер — сервер. Не клиент — серверная модель, как описано тут, а просто два сервера. Можно ставить дополнительные пакеты по необходимости. Тот же **MC** поставить не проблема, подойдет стандартная команда centos — yum install mc.

Система интересная и вполне надежная, можно пользоваться.

Помогла статья? Есть возможность отблагодарить автора