# Файловое хранилище на Linux

В этой статье мы расскажем как установить и настроить файловый сервер на операционной системе Linux, а точнее будет использована серверная Ubuntu 16.04 LTS. Аналогичным образом настраивается большинство deb-based дистрибутивов.

Такой сервер можно использовать для сетевой установки файловой базы 1С:Предприятие - это гораздо надежнее, чем хранить ее на одном из рабочих компьютеров пользователей. Или такой сервер можно приспособить под сетевое хранилище резервных копий.

Только не используйте один и тот же сервер для установки информационной базы и хранения ее резервных копий.

Почему Linux? Во-первых это бесплатно и при этом совершенно легально. Во-вторых Linux потребляет гораздо меньше аппаратных ресурсов, и даже старая, списанная в утиль техника отлично справится с задачей файлового хранилища. В-третьих, хорошо настроенный Linux практически не нуждается во вмешательстве системного администратора, эксплуатируются по принципу «настроил и забыл».

И так, начнем...

## Выбор оборудования

Как я уже написал, оборудование нам подойдет практически любое, но все же кое-какие пожелания у нас есть. Поскольку сервер будет файловый, то и пожелания наши будут касаться дисковой системы. Было бы неплохо найти машину с RAID контроллером на борту. Если мы делаем сервер для размещения рабочей файловой базы, было бы неплохо разместить ее на RAID-5, если хранилище резервных копий, отличным вариантом будет RAID-1.

При этом у нас нет особых требований к оперативной памяти, хватит и 1 Гбайта. К процессору тоже нет особых требований, Linux будет работать на всем, что еще живо.

Пожалуй, самый оптимальный вариант - приобрести восстановленный сервер «с пробегом». Берите самый дешевый, какой найдете, главное, что бы перед этим он прошел профилактику, его очистили от пыли и прогнали все системные тесты.

За неимением лучшего, можно использовать любой старый компьютер, но помните, что Вы это делаете на свой страх и риск. Самое уязвимое место файлового сервера - дисковая подсистема. Если она у Вас будет состоять из одного единственного старого диска, Вы очень сильно рискуете.
Если не удалось найти RAID-контроллер, можно попробовать настроить программный RAID средствами операционной системы. Учтите, что это повысит требования к процессору и оперативной памяти.

## Установка операционной системы

Сначала определимся с архитектурой сервера. Если Вам известна марка процессора,

установленного в сервер, ознакомившись с его спецификацией Вы узнаете, совместим ли он с архитектурой x86-64 (64 бит) или только i386 (32 бит). Косвенный признак - размер оперативной памяти, 32-битная архитектура не может работать с оперативной памятью объемом более 3 Гбайт, иногда в эту архитектуру устанавливали 4 Гбайт памяти, но в системе было видно только 3 Гбайт.

Идем на страницу загрузки Ubuntu Server и скачиваем дистрибутив, соответствующей архитектуры. Дистрибутивы Ubuntu распространяются в виде образов загрузочных DVD дисков.

Для установки Вам потребуется записать загрузочный DVD диск из скачанного образа, или, что как правило удобнее, подготовить загрузочную флешку специальной утилитой. Вставляйте диск или флешку в сервер и загружайтесь с нее.

Выбирайте русский язык и в меню Установить `Ubuntu Server`.

Далее Вам предложат указать страну, выбрать раскладку клавиатуры, дать имя серверу, указать имя и пароль суперпользователя (аналог администратора в Ubuntu) и подтвердить временную зону.

Некоторое затруднение может вызвать разметка диска. Если сомневаетесь, выбирайте автоматическую разметку и использовать весь диск. Но лучше выделить домашние папки пользователей в отдельные логические диски.

Так будет удобнее обновлять операционную систему, когда выйдет новая LTS версия 18.04.

Создавать или нет раздел подкачки зависит от объема оперативной памяти. Если у Вас ее немного, создайте раздел подкачки с таким же объемом. Впрочем, это не обязательно, можно после установки создать файл подкачки.

Далее в процессе установки Вам нужно будет выбрать каким образом Вы хотите управлять обновлением системы. Рекомендую устанавливать обновления безопасности автоматически.

И ближе к концу установки Вам предложат выбрать готовые наборы серверного программного обеспечения. Нам понадобятся:

- Samba file server
- Standart system utilites
- OpenSSH server

Инсталлятор завершит свою работу, перезапустит сервер, Вы увидите протокол загрузки операционной системы, который завершится приглашением ввести логин и пароль пользователя в консоль.

Добро пожаловать в Linux!

# Настройка сервера

Вводите логин и пароль суперпользователя, созданного при установке операционной системы.

Ввод пароля никак не отображается в командной консоли - это нормально.

Первым делом настроем сетевое подключение.

Во время установки инсталлятор продиагностировал установленное оборудование и определил имеющиеся в системе адаптеры. По умолчанию Ethernet адаптер настраивается на получение IP адреса через DHCP, нас это не устраивает, т.к. у нас не будет возможности обращаться к серверу по его логическому имени, мы настроим статический IP адрес.

Откройте конфигурационный файл сетевых интерфейсов командой

```
$ sudo nano /etc/network/interfaces
```

Здесь использована команда `sudo` - специальная конструкция deb-based дистрибутивов Linux для выполнения команд с правами root. Когда Вы делаете это первый раз система попросит Вас ввести пароль и на какое-то время запомнит его.

и приведите его к такому виду

```
# The loopback network interface - этот раздел не трогаем, оставляем как есть
auto lo
iface lo inet loopback

# The primary network interface - этот раздел настраивает Ethernet адаптер
auto enp0s3               # имя интерфейса оставляем без изменений
iface enp0s3 inet static  # меняем опцию dhcp на static
address 192.168.1.9       # укажите свободный IP адрес в Вашей сети
            # за пределами диапазона адресов, выдаваемых DHCP сервером,
            # если таковой используется
netmask 255.255.255.0     # маска подсети
gateway 192.168.1.1       # шлюз по умолчанию, обычно IP адрес сетевого
маршрутизатора
```

Сохраните файл нажав Ctrl-O и закройте редактор Ctrl-X. После редактирования перезапустим сеть:

```
$ sudo /etc/init.d/networking restart
```

и проверим что у нас получилось

```
$ ifconfig
```

В выдаче этой команды внимательно смотрим на значения `inet addr` - в нашем примере там должен быть статический адрес 192.168.1.9.

Дальнейшую настройку удобнее производить с рабочей станции, подключившись по протоколу SSH. От сервера можно отключить монитор, клавиатуру и разместить его там, где он не будет никому мешать.

Для дистанционного управления сервером с рабочей станции Windows мы будем использовать PuTTy. Скачайте, установите и подключайтесь. Адрес сервера в нашем примере указывается так `user@192.168.1.9`, где `user` - имя суперпользователя, порт по умолчанию 22.

Мы не будем использовать анонимный доступ к нашему файловому серверу, для того, что бы что-то записать или прочитать с сервера потребуется указать логин и пароль. И нам потребуется создать пользователя на сервере, от имени которого будут производиться все соответствующие файловые операции в хранилище.

```
$ sudo adduser storageuser
```

При создании пользователя так же будут созданы одноименные группа и домашняя папка. В домашней папке этого пользователя мы и организуем сетевое файловое хранилище

```
$ sudo -u storageuser mkdir /home/storageuser/nas
```

Пакет samba мы уже установили вместе с системой, дополнительно что-либо устанавливать не требуется.

Добавим пользователя в Samba

```
$ sudo smbpasswd -a storageuser
```

- тут нужно указать пароль пользователя Samba, и включим пользователя

```
$ sudo smbpasswd -e storageuser
```

Сделаем на всякий случай копию файла настроек и приступим к настройкам файлового сервера Samba.

```
$ sudo cp /etc/samba/smb.conf /etc/samba/smb.bak
$ sudo nano /etc/samba/smb.conf
```

Конфигурационный файл сопровождается подробными комментариями, можете пройтись по настройкам самостоятельно, а можете скопировать рекомендуемые настройки полностью

```
[global]
    workgroup = WORKGROUP        # Здесь укажите имя рабочей группы одноранговой сети
    server string = %h server (Samba, Ubuntu)
        name resolve order = wins lmhosts hosts bcast
    dns proxy = no

        wins support = yes              # только если в сети нет Wins сервера (он может быть, например, в роутере)
        ;wins server = 192.168.1.1      # только если wins support = no и по указанному адресу действительно есть Wins сервер

    log file = /var/log/samba/log.%m
    max log size = 1000
    syslog = 0
    panic action = /usr/share/samba/panic-action %d
```

```
    server role = standalone server
    passdb backend = tdbsam
    obey pam restrictions = yes

    unix password sync = yes

    passwd program = /usr/bin/passwd %u
    passwd chat = *Enter\snew\s*\spassword:* %n\n
*Retype\snew\s*\spassword:* %n\n *password\supdated\ssuccessfully* .

    pam password change = yes

    security = user
    username map = /etc/samba/smbusers

    map to guest = bad user

    usershare allow guests = yes

[storage]
    comment = nas storage
    writable = yes
    browseable = yes
    public = yes
    path = /home/storageuser/nas
    guest ok = no
    directory mask = 755
    create mask = 644
    valid users = @storageuser
```

Перезапустим службу

```
$ sudo service smbd restart
```

Пробуем зайти с какой-либо рабочей станции Windows, указав в проводнике путь `\\192.168.1.9`.

В сетевом окружении сервер появится через какое-то время, когда служба Wins обновит свои данные.

Windows сначала попробует открыть папку под своей локальной учетной записью, у нее это не получится и она запросит логин и пароль для доступа к сетевому ресурсу - это как раз тот пользователь, которого мы создали специально для доступа к сетевому хранилищу.

Готово!

При необходимости можно добавить новых пользователей и новые разделы. Разграничение доступа к разделам производится через опцию `valid users` в соответствующем блоке конфигурационного файла Samba.

# Антивирус

Операционные системы на базе Linux практически не подвержены риску заражения компьютерными вирусами, от части потому, что вирусов способных им навредить крайне мало, а в основном потому, что без получения привилегий суперпользователя эти вирусы ничем не могут навредить операционной системе.

Но эти вирусы могут использовать файловый сервер Samba для распространения от одной Windows системы на другие. Что бы поддерживать наше файловое хранилище в чистоте, мы установим антивирус и настроим автоматическое сканирование.

Установим антивирус ClamAV

```
$ sudo apt install clamav
```

Сразу же после установки в фоновом режиме запуститься обновление сигнатур, в дальнейшем мы настроим автоматическое обновление сигнатур по расписанию.

Удалять подозрительные файлы мы сразу не будем, мы их будем перемещать в карантин, где они никому не навредят. Если среди этих файлов было что-то важное, администратор сможет найти их в карантине и что-то сделать. Создадим папку карантина и ограничим доступ к ней

```
$ sudo mkdir /quarantine
$ sudo chmode 600 /quarantine
```

Попробуем просканировать домашние папки пользователей

```
$ sudo clamscan -i -r --move=/quarantine /home
```

После сканирования получим протокол

```
----------- SCAN SUMMARY -----------
Known viruses: 6278963
Engine version: 0.99.2
Scanned directories: 13
Scanned files: 13
Infected files: 0
Data scanned: 4.79 MB
Data read: 1.59 MB (ratio 3.00:1)
Time: 22.176 sec (0 m 22 s)
```

Все хорошо, вирусов не обнаружено. Если бы нашлось что-то подозрительное, оно было бы перемещено в папку карантина.

Нам остается настроить автоматическое расписание обновления сигнатур и сканирования домашних папок. Редактируем файл расписания демона cron

```
$ sudo crontab -e
```

Для обновления сигнатур и сканирования нам потребуются привилегии суперпользователя,

поэтому crontab запускается через sudo, сами команды в файле расписания нужно указывать без sudo.

Добавьте две строчки

```
0 1 * * * freshclam
0 2 * * * clamscan -i -r --move=/quarantine /home
```

Каждый день в 1:00 ночи будет автоматически запускаться обновление сигнатур, а в 2:00 ночи будет запущено сканирование всех домашних папок пользователей, инфицированные файлы будут перемещены в папку карантина.

# Мониторинг

## Регулярность резервного копирования

Если Вы пользуетесь мессенджером Telergam, у нас для Вас есть утилита мониторинга резервного копирования. Она умеет сканировать папки сетевого хранилища и сообщать о наличии или отсутствии новых файлов. Например, если резервное копирование запланировано на ночь, а утром в сетевом хранилище нет новых файлов, значит что-то пошло не так и нужно с этим разобраться.

Утилита написана на Python, сам Python в Ubuntu установлен по-умолчанию, нужно установить дополнительный модуль.

```
$ sudo apt install python-pip
$ sudo pip install --upgrade pip
$ sudo pip install python-telegram-bot
```

Сама утилита устанавливается из репозитория GitHib

```
$ cd ~
$ git clone https://github.com/kuleshovdv/backtracker.wiki.git
$ cd backtracker
```

Создайте для себя нового Telegram бота. Подробная инструкция как это сделать приведена тут (англ).

Свяжитесь с Отцом Ботов, отправьте ему сначала команду `/start`, затем `/newbot`. Далее отвечайте на вопросы Отца Ботов, в итоге Вы получите от него токен и ссылку на Вашего бота.

Открываем конфигурационный файл

```
$ nano backtracker.conf
```

и настраиваем

```
[Telegram]
```

```
token = # Тут нужно указать токен telegram-бота, полученный от Отца Ботов
failonly = # False если хотите получать сообщения о наличии новых файлов или True если только об их отсутствии

[Scan]
path = # Укажите путь к сканируемым папкам
hours = # Укажите "свежесть" файлов в часах, например 8
```

Запускайте утилиту

```
$ ./backtracker.ry
```

Первый запуск нужен для того, что бы автоматически определить ID абонента Telegram, который будет получать сообщения (это не номер его телефона). Подключайтесь к своему боту по ссылке, которую Вам дал Отец Ботов и отправляйте ему команду `/start`. В ответ Вы получите сообщение, что Ваш ID определен, а утилита самонастроится и закроется. Запустите ее повторно для выполнения сканирования.

После настройки и проверки работы утилиты, добавьте ее в расписание демона cron

```
$ crontab -e
```

Добавьте строчку

```
0 8 * * * ~/backtracker/backtracker.py
```

Проверка будет запускаться каждый день в 8 утра. Если ночью что-то пошло не так, Вы узнаете об этом.

## Системные ресурсы

Мониторить ресурсы сервера можно консольной утилитой `top` или ее более красочной версией `htop`. Установим и запустим ее

```
$ sudo apt install htop
$ htop
```

Периодически контролируйте использование оперативной памяти. Если часто наблюдается загруженность около 100%, настройте файл подкачки.

```
$ sudo dd if=/dev/zero of=/swapfile bs=1M count=1024
$ sudo chmod 600 /swapfile && sudo mkswap /swapfile
$ sudo swapoff -a
$ sudo swapon /swapfile
$ echo "/swapfile swap swap defaults 0 0"| sudo tee -a /etc/fstab
```

Здесь `count=1024` - размер файла подкачки в мегабайтах.

## Дисковое пространство

Для мониторинга файловой системы удобно пользоваться файловым менеджером Midnight Commander. Если Вы застали времена MS DOS и Notron Commander, то объяснять ничего не нужно.

Устанавливаем и запускаем

```
$ sudo apt install mc
$ mc
```

Так удобно наблюдать за файловым хранилищем, карантином, свободным дисковым пространством.

Есть вопросы, нужна консультация или помощь в настройке - обращайтесь, будем рады помочь.


- Twitter
- Facebook
- Google+
- LinkedIn
- Pinterest
- Tumblr
- Reddit
- Taringa
- StumbleUpon
- Telegram
- Hacker News
- Xing
- Vk
- Email



From:
https://wiki.lineris.ru/ - **ЛИНЕРИС**

Permanent link:
**https://wiki.lineris.ru/nas_linux?rev=1494493673**

Last update: **2017/05/11 09:07**