Введение. Установка

1FreeBSD

FreeBSD: что это такое и зачем нужно. Установка FreeBSD. Основы работы с FreeBSD. Работа с pkg и freebsd-update.

[Для кого предназначен этот курс. План занятий](#_c713xnuw3xem)

[Что такое FreeBSD](#_25e23jgyexza)

[Установка FreeBSD](#_w7f40nphcut0)

[Скачиваем образ](#_qhiucgp2pp3j)

[Настраиваем виртуальную машину](#_t5sywn2efuom)

[Устанавливаем FreeBSD](#_9hkp2e85bl99)

[Знакомство с FreeBSD](#_w147ru4nll5q)

[Подключаемся по ssh](#_z9sdveejiyd1)

[Файловая система во FreeBSD](#_v69ifpy7gn44)

[Система портов, pkg, man](#_c7xs54m4a9dj)

[Обновление BSD](#_r6ytmojn4b01)

[Домашнее задание](#_3tfrjxxltv85)

[Дополнительные материалы](#_y937sk8fclye)

[Используемая литература](#_uvp6qax5r1ok)

# 

# Для кого предназначен этот курс. План занятий

Курс «FreeBSD и автоматизация» от GeekBrains – один из немногих курсов, посвященных операционной системе FreeBSD, которая сейчас доступна для изучения. Он может показаться сложным, поэтому до его изучения желательно познакомиться с UNIX-подобными операционными системами, например c GNU/Linux. Если вы прошли курс Linux, внимательно изучали материал и выполняли домашние задания, то и с этим курсом у вас проблем быть не должно.

Курс рассчитан на системных администраторов, которые хотят углубить знания по UNIX-подобным системам, изучить тонкости настройки FreeBSD и сделать еще один шаг на пути к DevOps.

Курс состоит из двух частей. Первая часть посвящена FreeBSD: это UNIX-подобная операционная система, как и уже известная вам Linux. Для системного администратора важно познакомиться с еще одной UNIX-подобной ОС и понять, что похожее окружение здесь работает на другом ядре. В отличие от Linux, FreeBSD – прямой потомок UNIX. Во многом FreeBSD похожа на Linux, тем не менее, у нее совсем другое ядро, и отличия между этими двумя системами тоже есть.

Вторая часть посвящена задачам автоматизации, умение работать с которыми повышает вашу ценность как специалиста на рынке труда и позволит в дальнейшем работать на позициях SRE/DevOps Engineer.

План занятий:

Урок 1. Введение, установка FreeBSD. Основы работы. Утилиты pkg и freebsd-update.

Урок 2. Инструментальные средства системного администрирования. Установка из портов и пакетов.

Урок 3. Сервисы. Развертывание веб-стека на основе nginx, php-fpm, perconadb во FreeBSD.

Урок 4. Диагностика системы и решение проблем. Поиск узких мест. Система мониторинга Zabbix.

Урок 5. Введение в автоматизацию с помощью Ansible.

Урок 6. Автоматизация с помощью Ansible.

Урок 7. Введение в Docker.

Урок 8. Дальнейшее развитие. Введение в профессию DevOps-инженера.

# Что такое FreeBSD

Операционная система FreeBSD происходит от BSD UNIX, которая появилась в Калифорнийском университете в Беркли (The University of California, Berkeley), расположенном в маленьком американском городке в штате Калифорния. BSD (Berkley Software Distribution System) – это система, созданная для обмена программным обеспечением между университетами. Когда в Университет Беркли были переданы исходные коды UNIX, это стало толчком к развитию к BSD UNIX, прямым потомком которой является FreeBSD (а также – через FreeBSD и NextStep – Mac OS X, хотя сейчас между этими системами много различий).

BSD UNIX примечательна тем, что в ней впервые был реализован стек TCP/IP, в настоящее время реализованный во всех современных операционных системах и сетевом оборудовании.

Неудивительно, что работа с сетью – одна из сильных сторон FreeBSD, которая позволяет решить «проблему десяти тысяч соединений» (С10k): в 1999 году было обнаружено, что на гигабитном канале сервер теоретически должен иметь возможность обслужить 10 000 соединений, но на практике программное обеспечение этого не позволяло. FreeBSD эффективна для сетевых задач и развертывания веб-сервисов и веб-приложений.

Также отметим, что лицензия BSD – одна из самых простых и лояльных. Она позволяет использовать код, доступный по лицензии BSD, как в открытом, так и в коммерческом программном обеспечении.

Символы FreeBSD – красный демоненок Beasty (на самом деле его зовут BSD-демон) и Devilette (Daemonette) – девушка в красном с рожками (таких девушек можно встретить на конференциях, посвященных FreeBSD).



Beasty и две Devilette

# Установка FreeBSD

## Скачиваем образ

Заходим на <https://www.freebsd.org>.

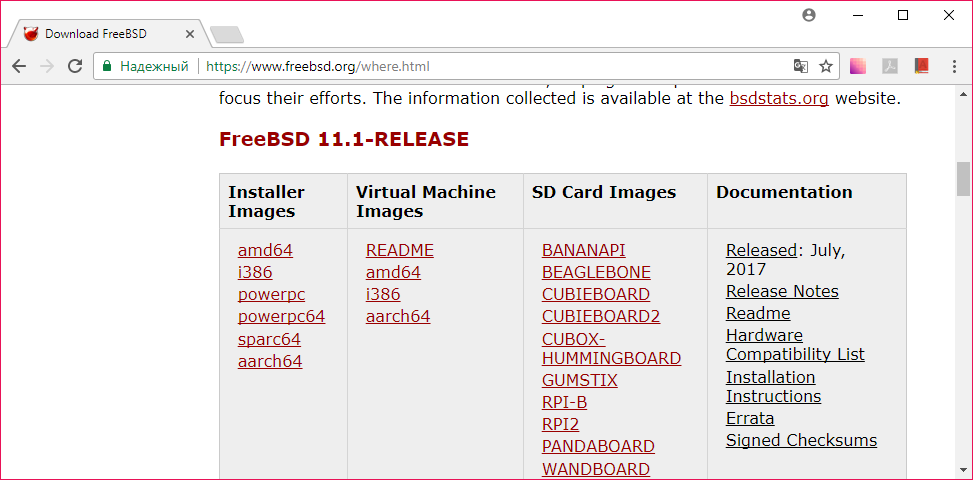


Нажимаем Download FreeBSD.

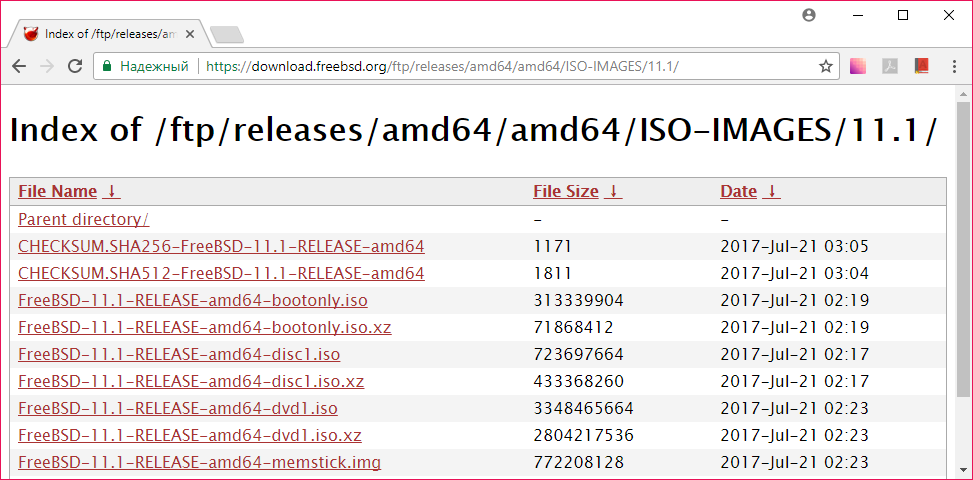
Открывшуюся страницу прокручиваем до перечня образов последнего релиза.

Здесь мы видим ссылки на образы для разных архитектур процессоров. Среди них:

* amd64 – 64-битная (пусть слово AMD вас не смущает, этот вариант подойдет и для Intel-процессоров: дело в том, что компания AMD разработала 64-битную архитектуру, которая применяется и в Intel-совместимых процессорах);
* i386 – 32-битная (i386 также не означает, что этот вариант предназначен только для процессора Intel 386: это знак того, что впервые Intel-совместимая 32-битная архитектура была реализована в процессорах Intel 386). Этот вариант следует выбрать, если ваш процессор 32-битный и не поддерживает 64-битную архитектуру (а вот на 64-битном процессоре 32-битные приложения выполняться могут).



Если у вас 64-битный процессор, смело выбирайте amd64.

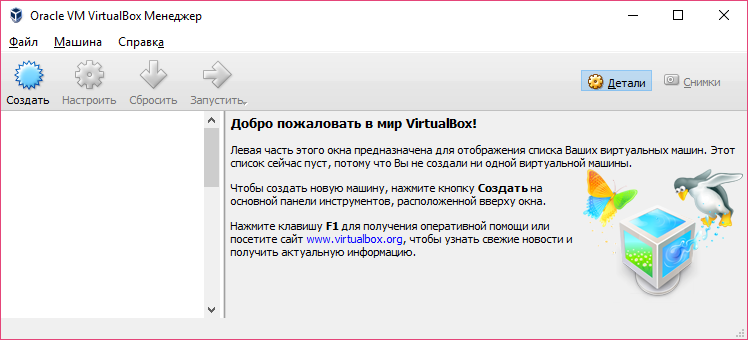


Выбираем вариант bootonly.iso и скачиваем.

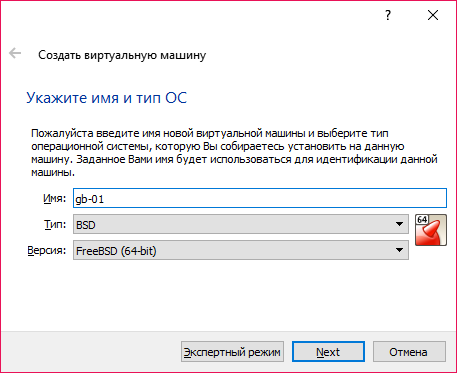
## Настраиваем виртуальную машину

Подойдет как Oracle VirtualBox, так и VMWare (в том числе и доступный бесплатно VMWare Player).

Рассмотрим создание виртуальной машины на примере VirtualBox.



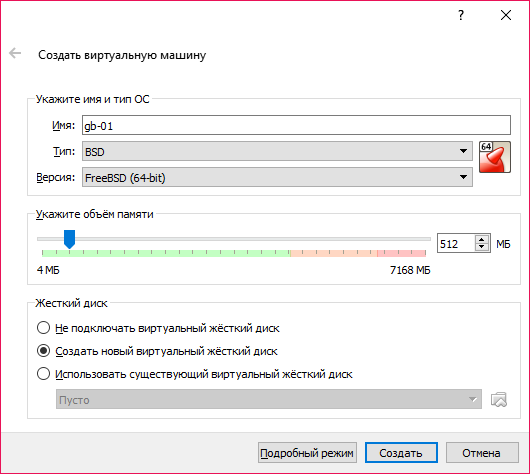
Кликаем на **Создать.**



Даем имя нашей виртуальной машине (на ваш выбор), тип выбираем BSD, версию – FreeBSD (32-bit), если у вас 32-битная система, либо FreeBSD (64-bit), если 64-битная. Если у вас 64-битный процессор, а предлагается выбрать только 32-битный режим, скорее всего, причина в том, что в процессоре отключена аппаратная поддержка виртуализации. Чтобы это исправить, необходимо перезагрузиться, войти в BIOS/Setup Utility (успеть нажать F2, либо F12, либо другой вариант в зависимости от вашего компьютера) и включить поддержку аппаратной виртуализации Intel VTx или AMD SVM либо AMD-V (та или иная технология может использоваться в зависимости от вашей системы), сохранить и выйти. После этого в вашей машине должна появиться возможность выбрать 64-битную версию.

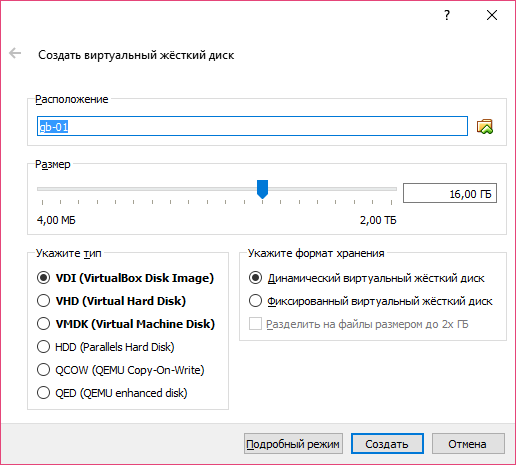
Далее жмем **Экспертный режим** (а лучше нажать эту кнопку сразу и только потом вводить данные, так как VirtualBox в этом случае их не сохранит, и все придется вводить заново).

Памяти для FreeBSD хватит 512 Mb, опции для диска оставляем без изменений – **Создать новый диск.**

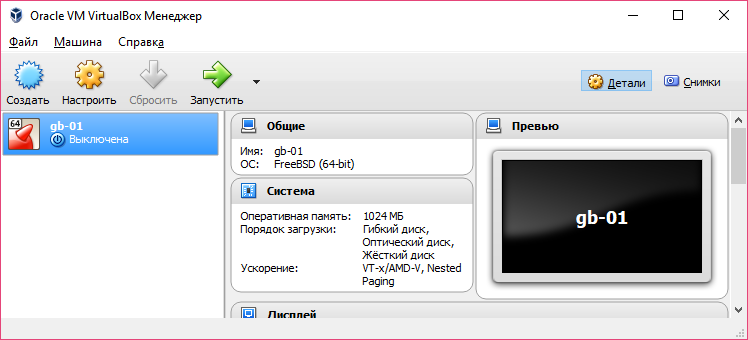


Далее жмем **Создать.**

Настройки жесткого диска подойдут по умолчанию:

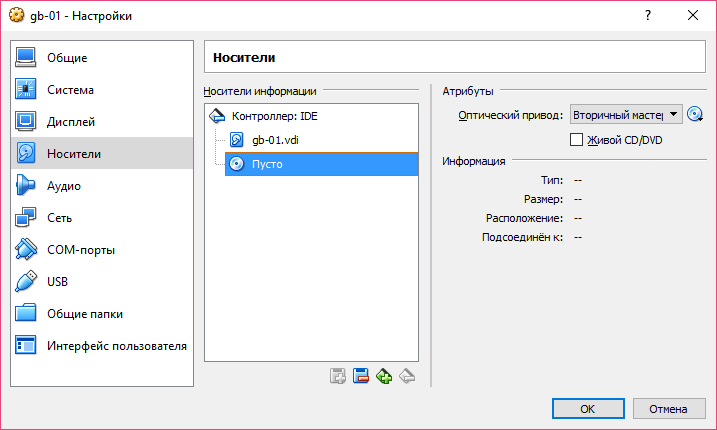


Также жмем **Создать.**

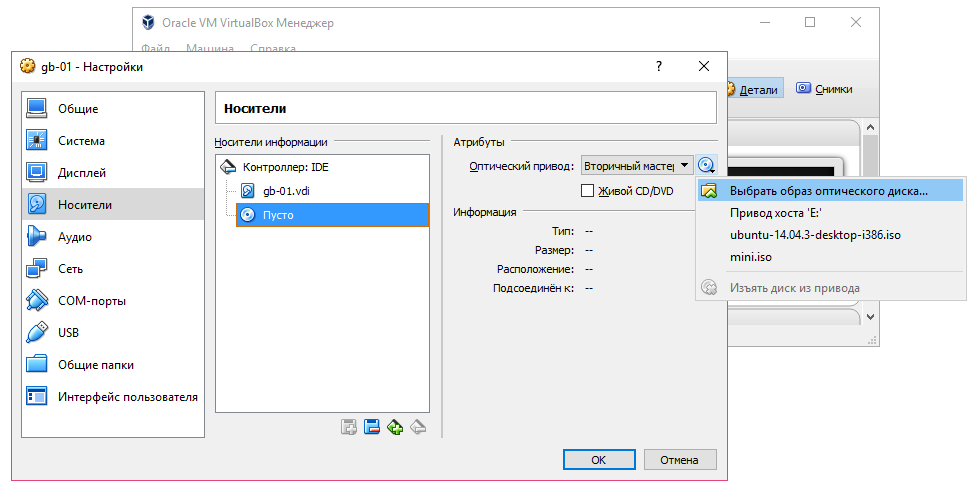


Теперь запустим образ установочного диска в виртуальном оптическом приводе.

Нажимаем кнопку **Настроить,** выбираем пункт **Носители:**

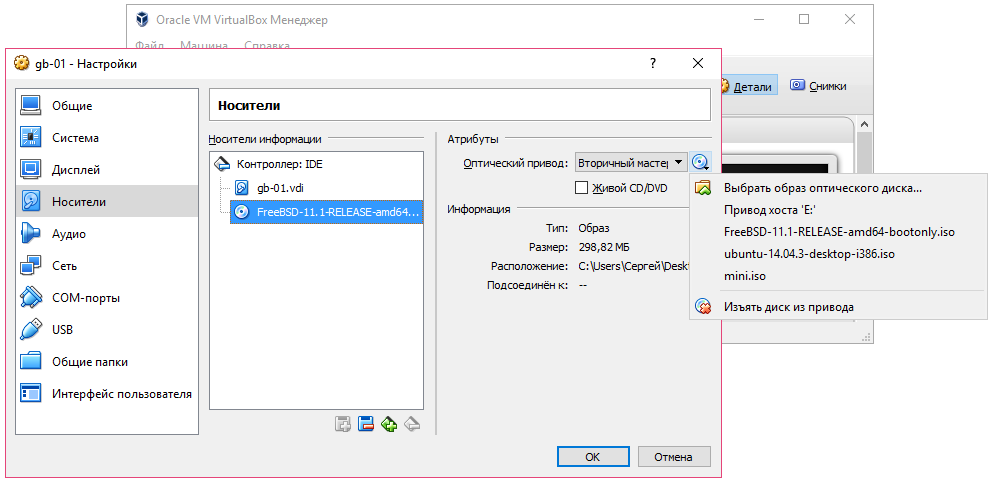


Кликаем на значок CD-диска в разделе **Атрибуты** > **Оптический привод:**

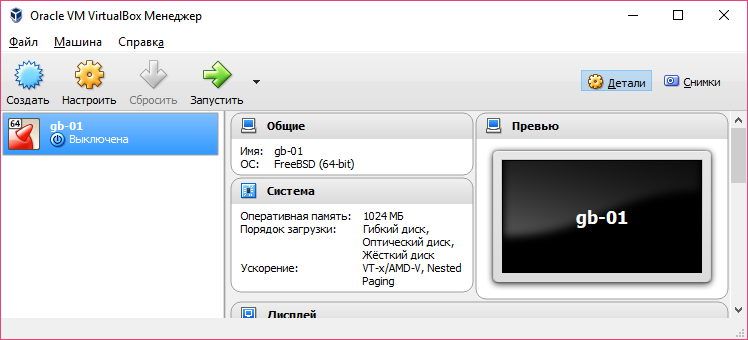


Кликаем на **Выбрать образ оптического диска.**

Готово:



Жмем **OK** и нажимаем кнопку **Запустить.**

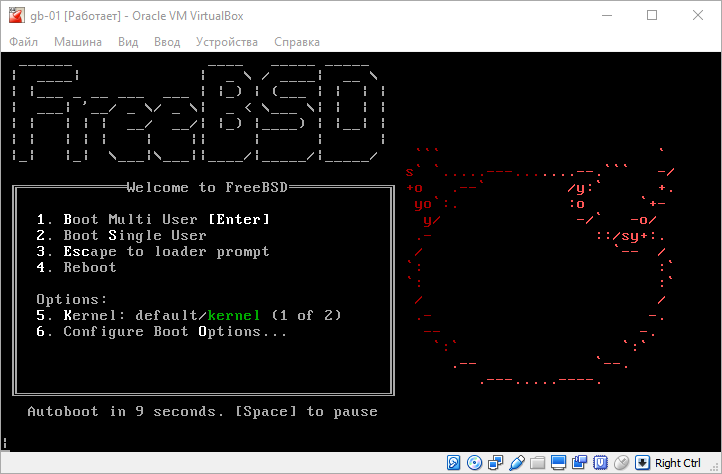


## Устанавливаем FreeBSD

Начинается загрузка с образа CD:

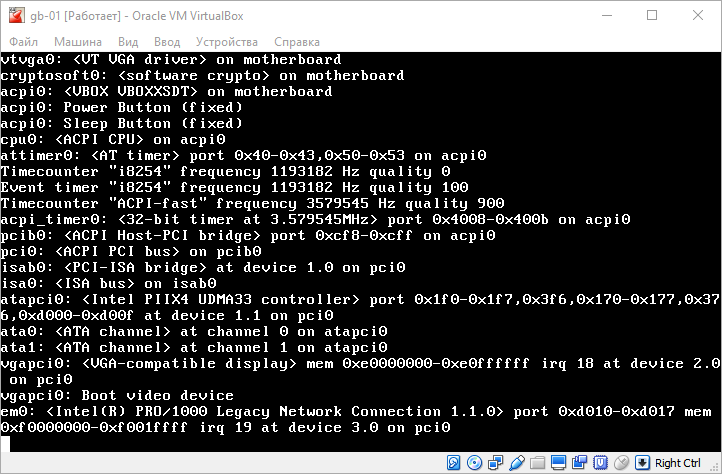


Вверху мы видим сообщение об автозахвате клавиатуры и мыши. В правой нижней части окна видим, что для этого используется клавиша RIght Ctrl. В зависимости от вашей операционной системы и системы виртуализации могут использоваться разные сочетания клавиш. Например для VMWare – сочетание Ctrl+Alt или Options+Alt+Fn.

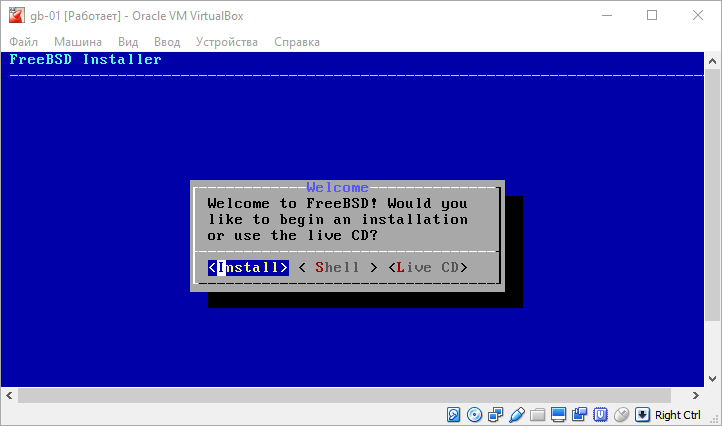


Можно ничего не делать, и через 10 секунд загрузка начнется сама. А можно выбрать опции ядра или сконфигурировать опции загрузки.

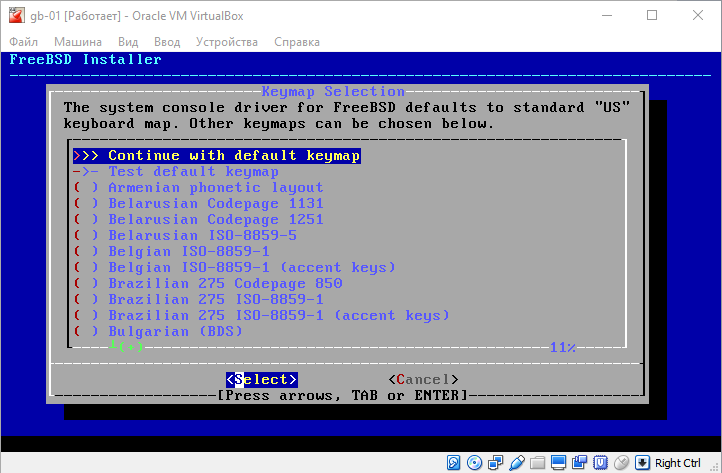
Мы просто ждем, пока система загрузится.



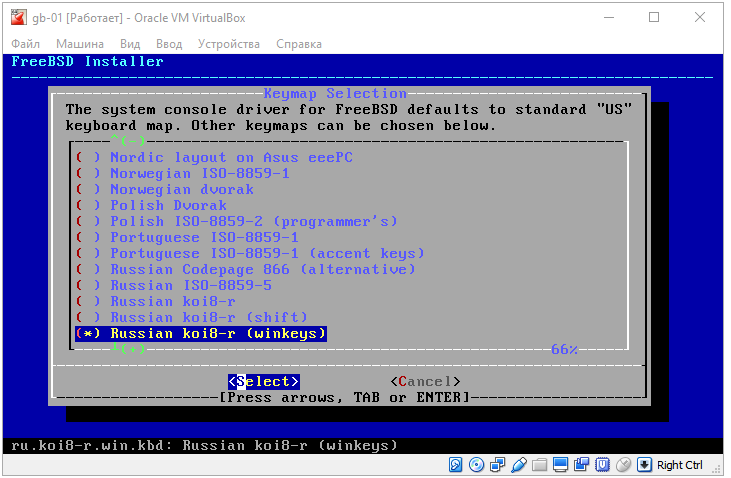
Затем появляется выбор: установить систему, либо перейти в оболочку, либо запуститься в режиме LiveCD.



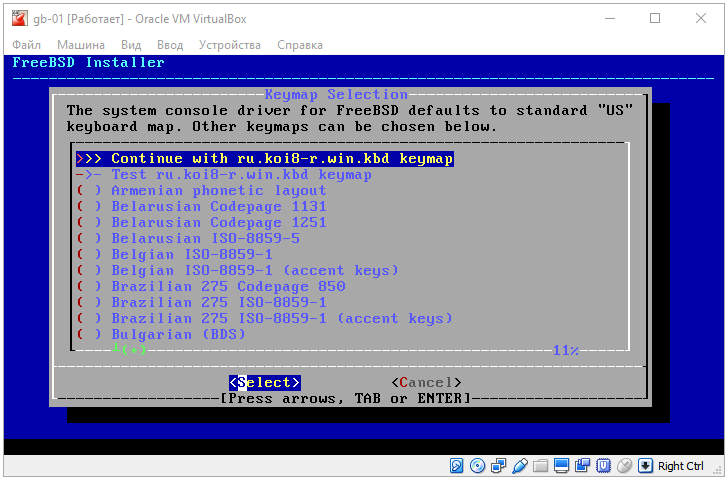
Жмем **Install.**



Пролистываем ниже и выбираем традиционную для UNIX-подобных систем восьмибитную кодировку koi8-r (winkeys). Winkeys означает, что раскладка будет переключаться как в Windows – с помощью сочетания Ctrl+Shift.

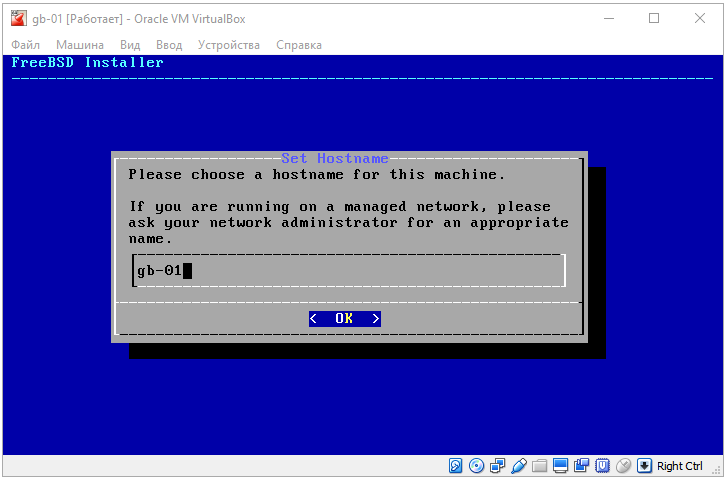


Жмем Enter. После этого будет выбран нужный вариант, а мы перемещаемся в начало списка, где можно протестировать переключение раскладки или продолжить дальше.

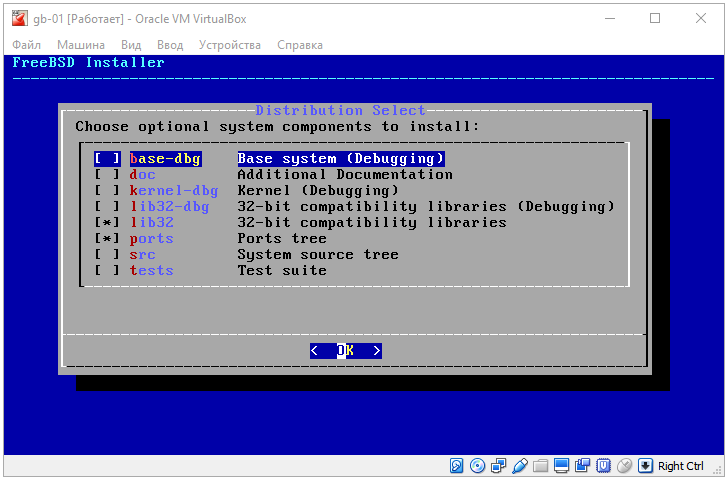


Выбираем **Продолжить дальше** с выбранной кодировкой, жмем Enter.

Вводим имя хоста машины. Например, gb-01. На практике каждая машина получает свое имя хоста, например: alpha, beta, gamma или srv01, srv02, srv03 и т. д.



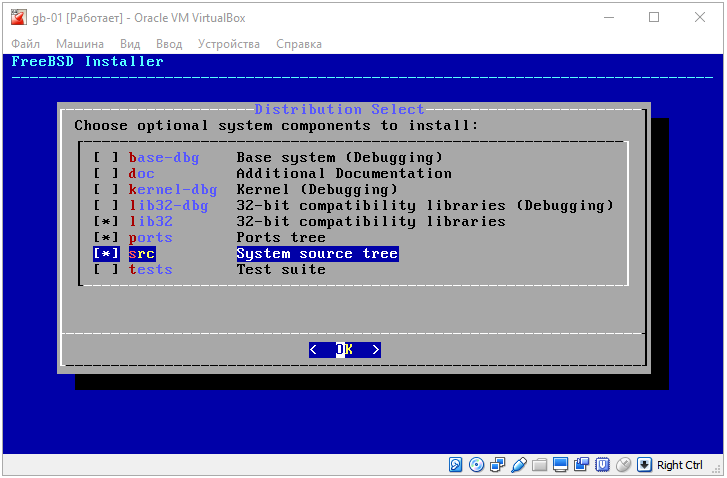
Жмем Enter.



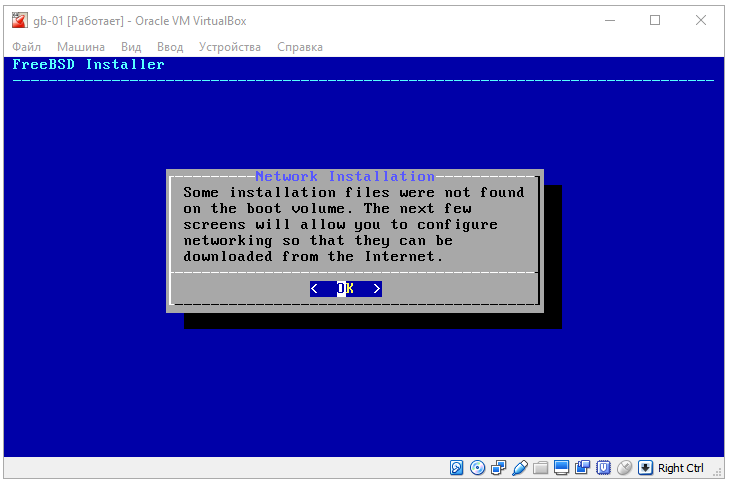
Необходимо выбрать компоненты.

Из этого набора по умолчанию установлены библиотеки для поддержки 32-битных программ (lib32) и порты (ports). Может возникнуть впечатление, что нужно поставить документацию (doc), чтобы были доступны мануалы (man). Это не так. Документация (doc) – это документация, примеры, но не мануалы. Мануалы у нас уже есть, поэтому ради них документацию устанавливать не надо. Более того, на сайте <https://www.freebsd.org/> достуна подробная документация, Handbook, к которой всегда можно обратиться.

Также добавим исходники (src), они нам понадобятся. Нажимаем пробел, чтобы выбрать src, и жмем Enter.

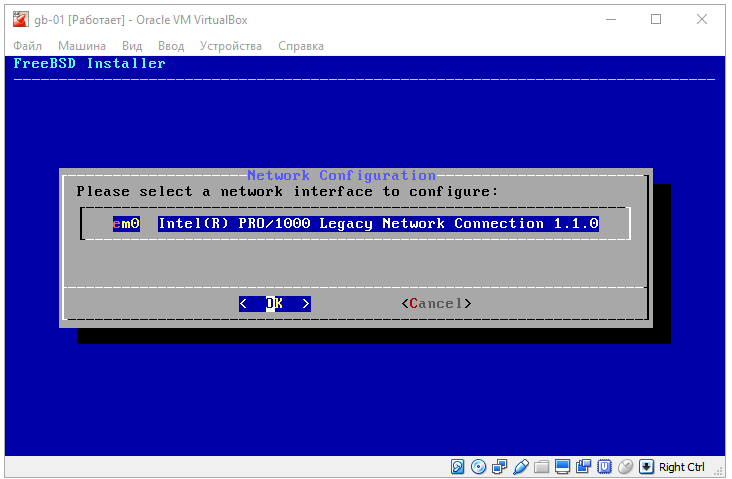


f

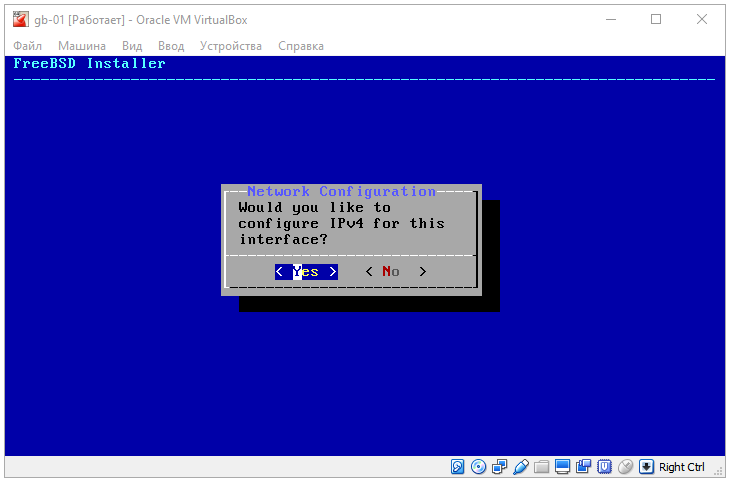


Из-за того, что мы выбрали Minimal образ, часть файлов будет скачиваться из Интернета.

Поэтому дальше следует настроить сеть:

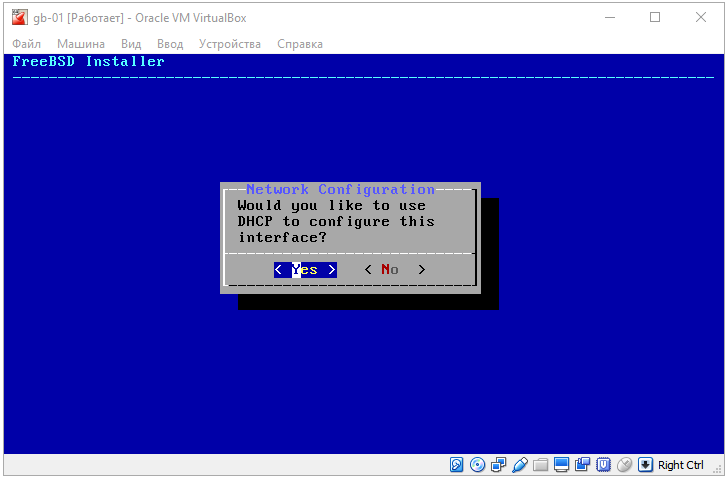


Жмем Enter.



Настроить IPv4? **Да.**

Нужно ли использовать DHCP? Это зависит от вашей сети. В данном случае выбираем **Да.**



Нужно ли настроить IPv6?

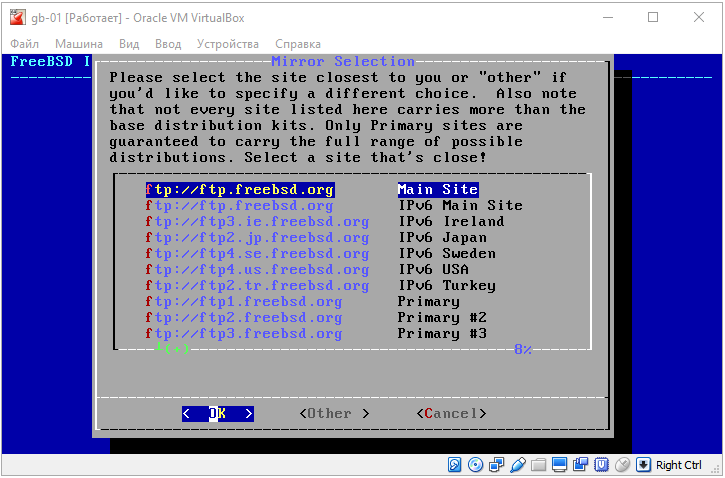


В этом курсе мы использование IPv6 не рассматриваем, потому выбираем **Нет.**

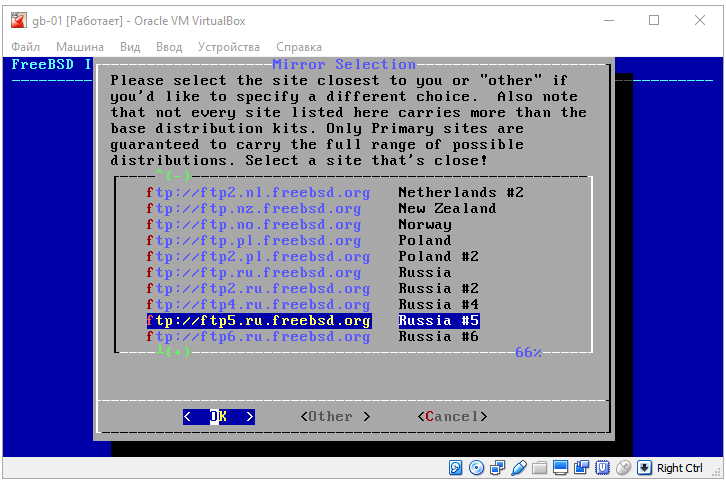


**DNS** – если все верно, оставляем как есть и нажимаем Enter.

Далее следует выбрать сервер, откуда будем скачивать недостающие файлы.

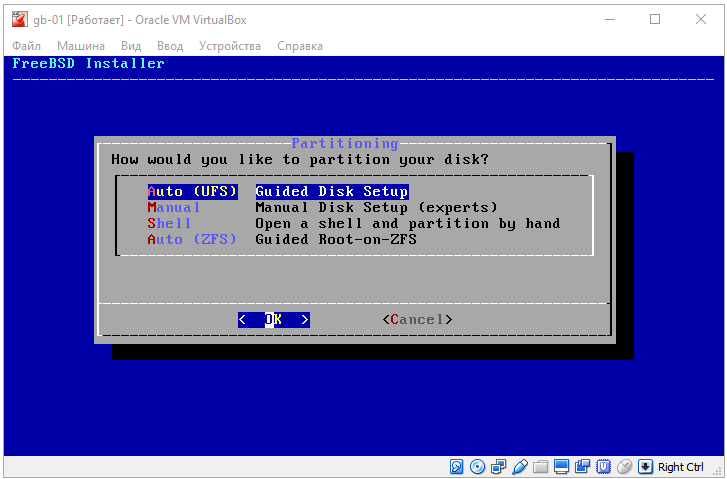


Предлагается выбрать ближайший, потому ищем российские серверы.



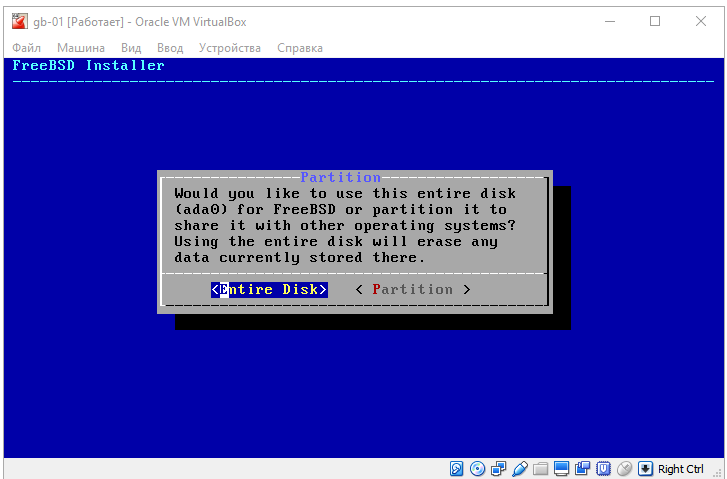
Первые два, как правило, – самые загруженные. Стоит отметить, что эти доменные имена – CNAME на ресурсы разных компаний, в частности 2 – mirror.macomnet.ru, 4 – earth.corbina.ru, 5 и 6 – mirror.yandex.ru. Выберем зеркало от Яндекса.

Жмем Enter. Теперь нам нужно определиться со способом разбивки диска.

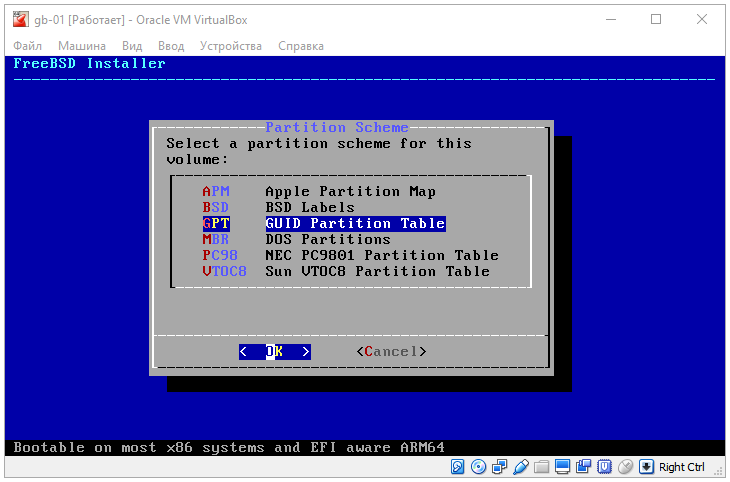


ZFS для корневого раздела мы использовать не будем. Можно выбрать ручную разбивку (Manual) или выполнить ее в шелле (Shell).

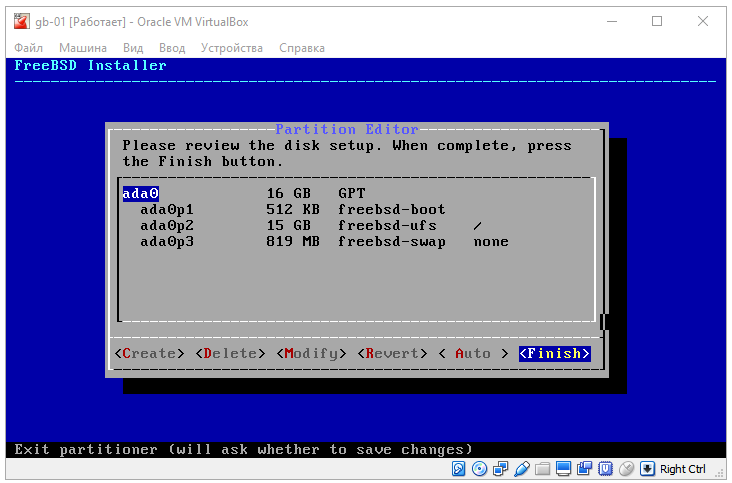
Остановимся на **Auto (UFS).**



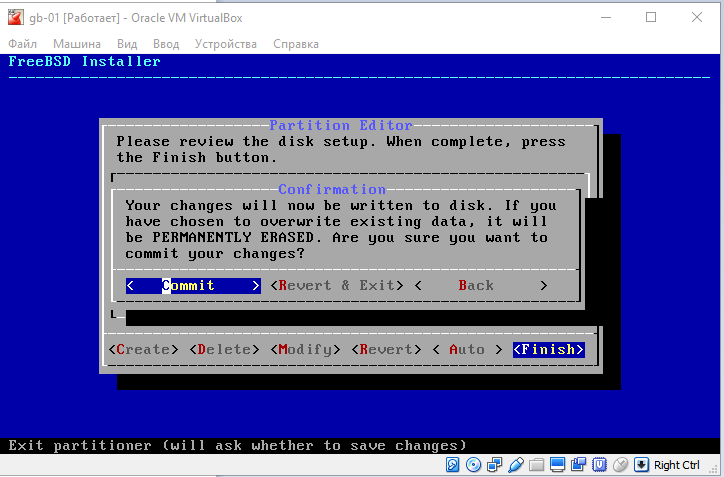
Нужно выбрать, будем ли мы использовать диск целиком или раздел для совместного использования с другими ОС. Выбираем **Диск целиком.**



Требуется выбрать тип загрузочной записи. Выберем **GPT.**

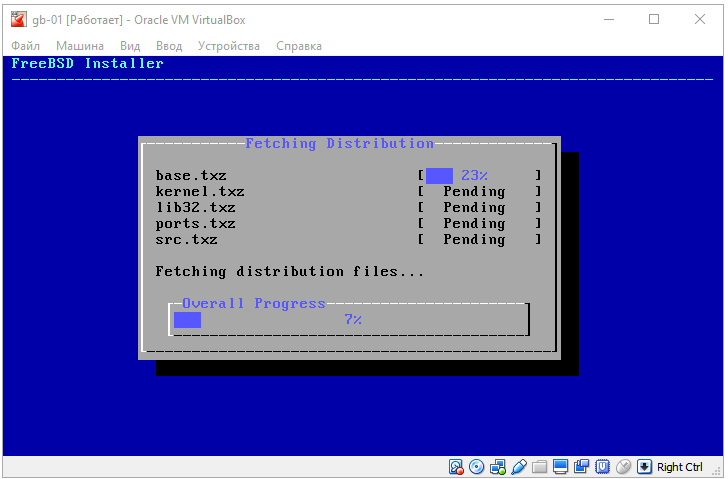


Настройки нас устраивают, жмем Enter.

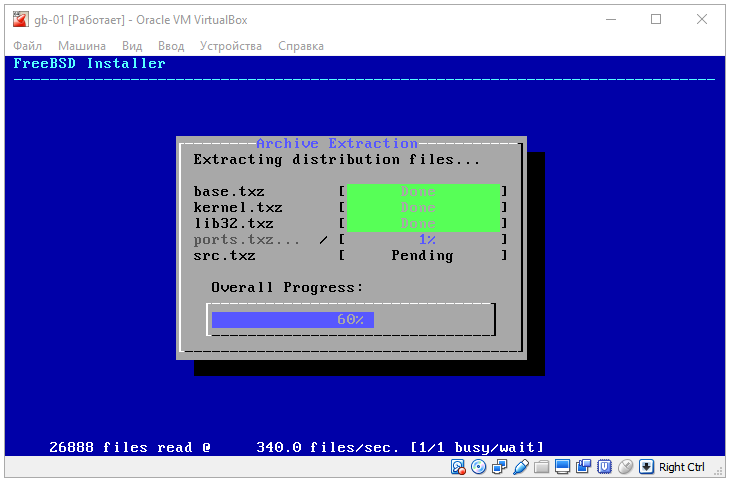


Выбираем **Commit.**

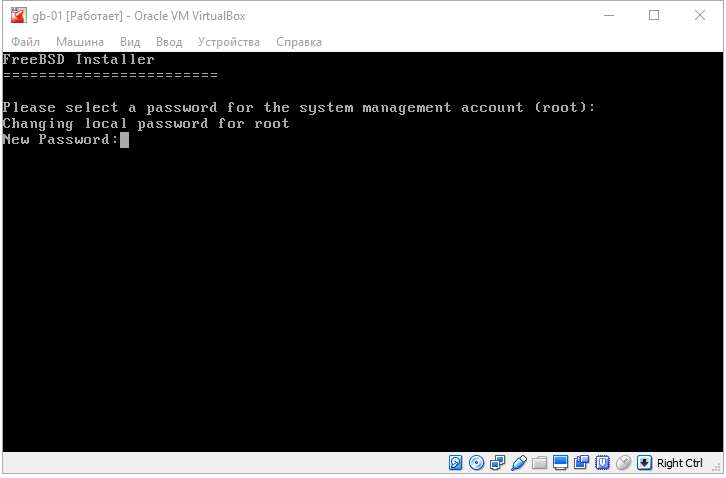
Идет процесс получения файлов.



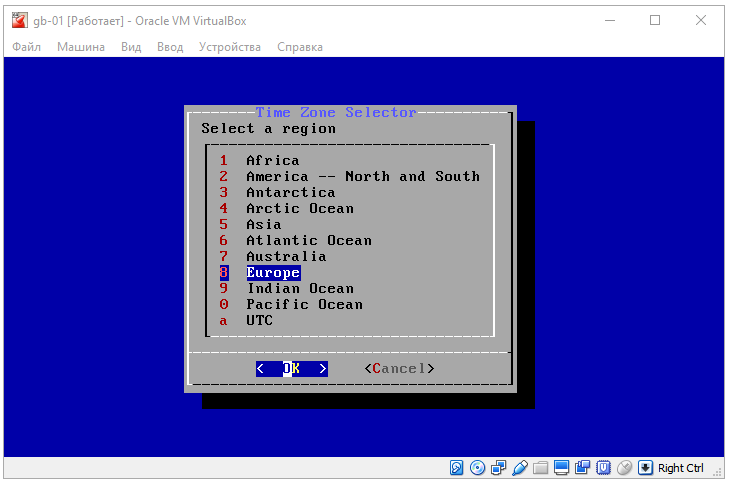
Идет распаковка.



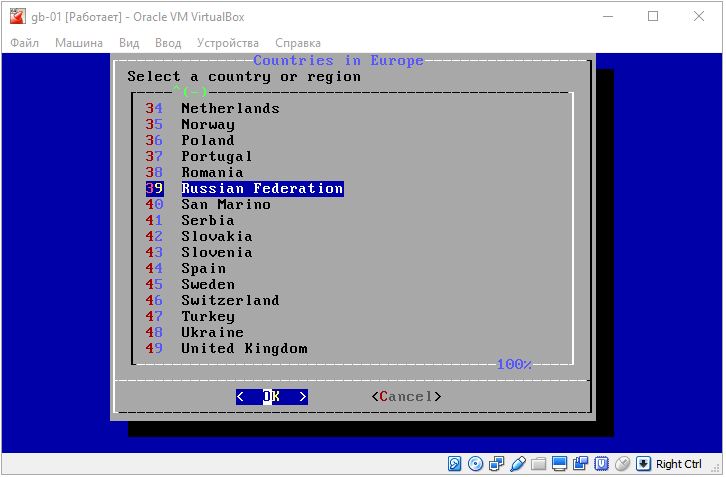
Задаем пароль суперпользователя – рута.



Обратите внимание, что при вводе символов ничего не отображается – ни звездочек, ни каких-либо иных знаков.

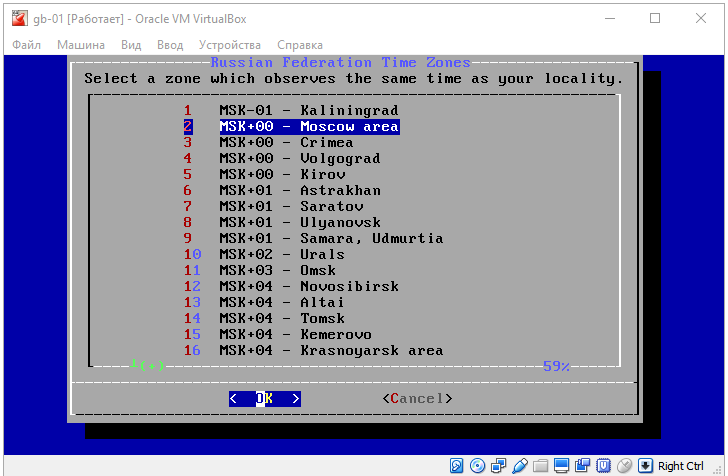


Выбираем регион.

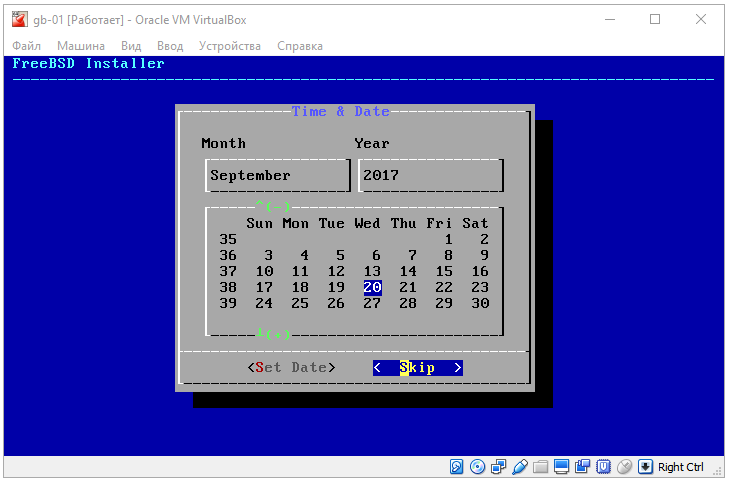


Выбираем страну.

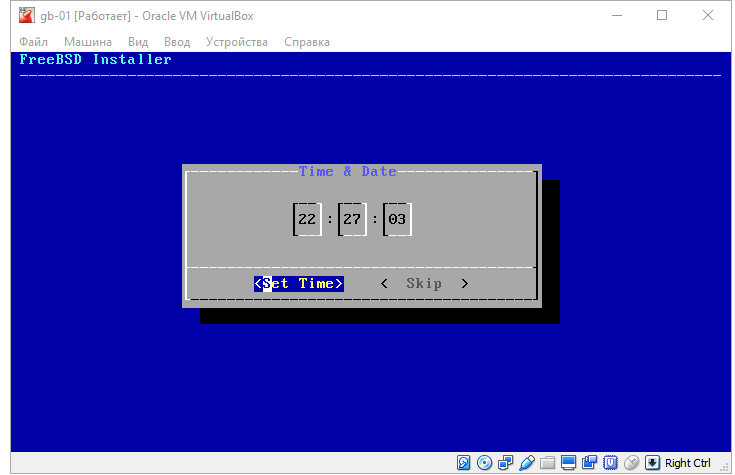
Выбираем часовой пояс (временную зону).



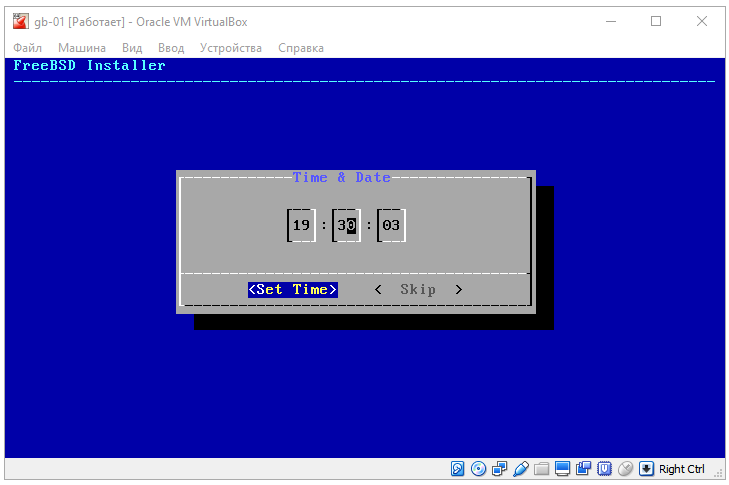
Далее дату.



Выберите **Set Date.**

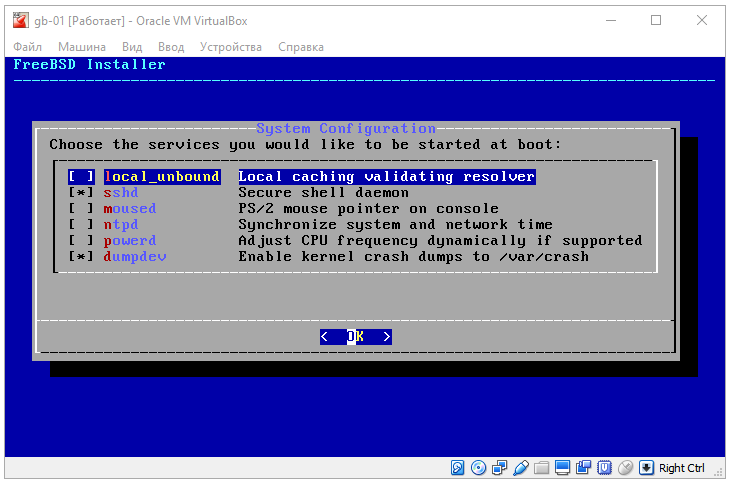


Можно исправить время. С помощью стрелок влево-вправо наведите курсор на часы и исправьте.



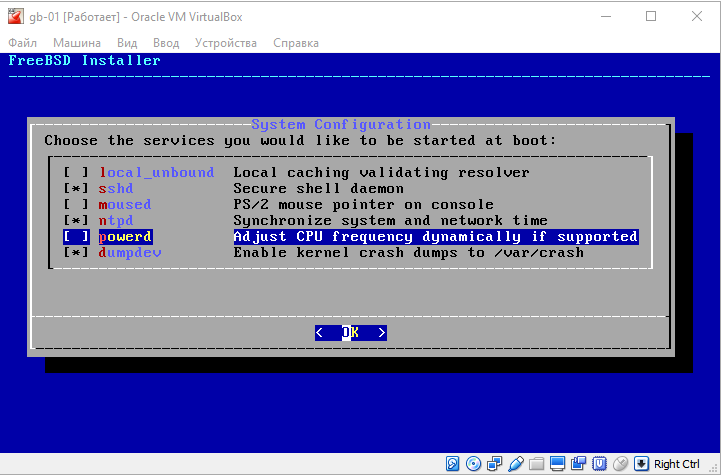
Изменение времени – стрелки вверх-вниз. Жмем **Set Time.**

Далее мы выбираем сервисы, которые должны быть запущены при загрузке:



* local\_unbound – локальный кеширующий сервер (не нужен);
* sshd – ssh-демон – нужен;
* moused – поддержка мыши в текстовой консоли – в принципе, не нужно;
* ntpd – демон для синхронизации времени по протоколу NTP, стоит включить;
* powerd – не требуется;
* dumpdev – для сохранения дампов ядра при падении ядра в /var/crash – может пригодиться.

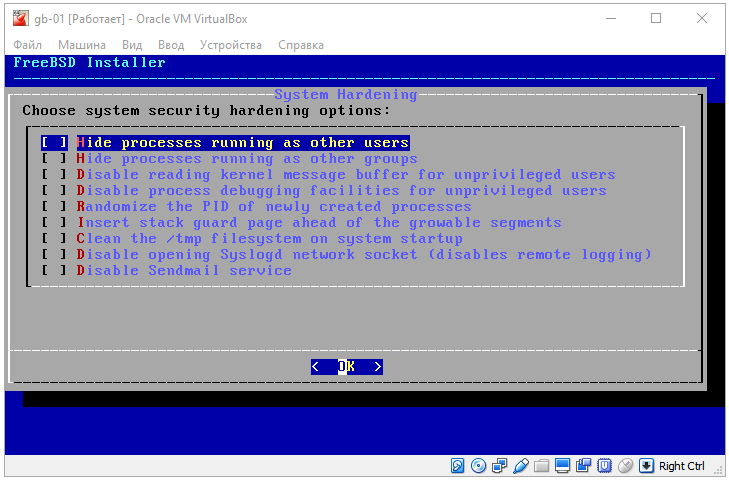
Отметку возле пункта ставим клавишей Пробел.



Жмем **OK.**

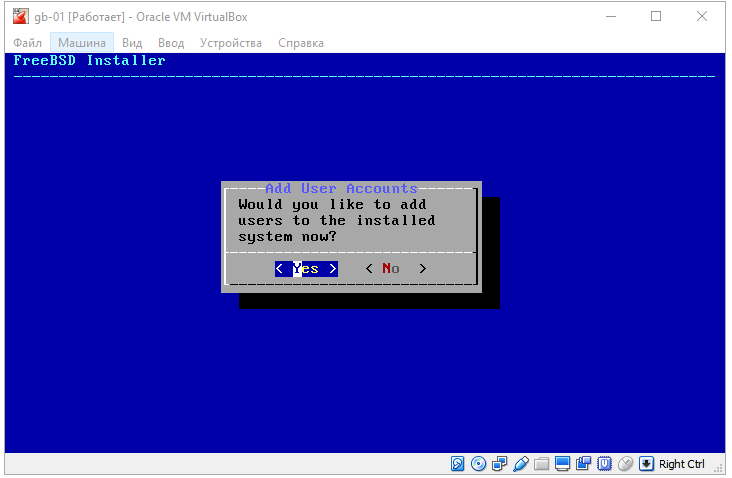
Далее – настройки безопасности.

Все пункты необходимо отметить. Далее объясним, почему.

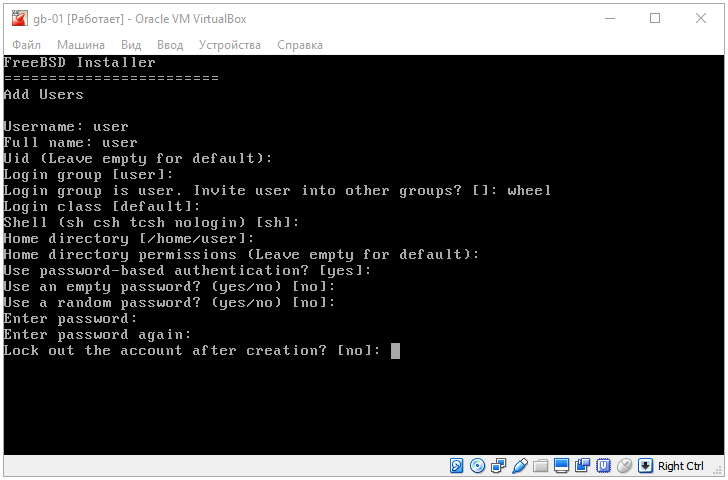


Отмечаем все пункты:

* Скрывать процессы, запущенные от имени других пользователей.
* Скрывать процессы, запущенные от имени других групп.
* Отключить чтение буфера сообщений ядра для непривилегированных пользователей.
* Отключить возможность отладки для непривилегированных пользователей.
* Рандомизировать PID (process ID) для вновь созданных процессов.
* Вставить страницу защиты стека перед подвижными сегментами.
* Очищать /tmp при старте системы.
* Отключить удаленное логирование syslogd.
* Отключить sendmail (довольно сложно настраиваемый почтовый сервис).



Хотим ли мы добавить пользователей в установленную систему сейчас? **Да.**



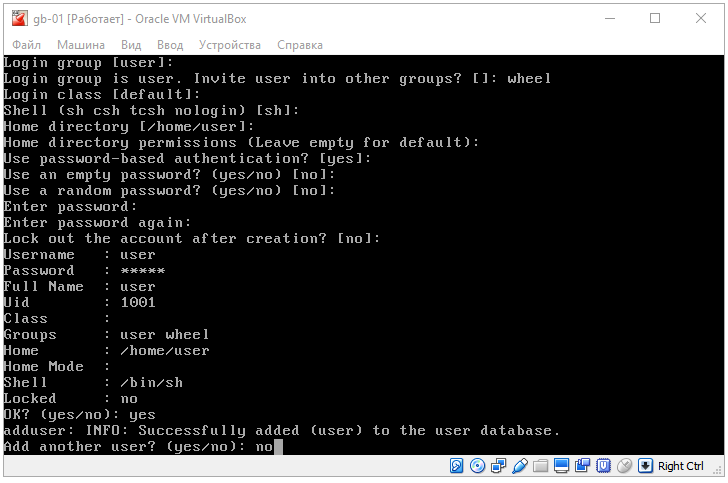
Вводим имя пользователя. Например, user. Полное имя – это может быть, например, ФИО; в отличие от имени пользователя, это не критично. UID оставим по умолчанию, основную группу – тоже по умолчанию, соответствующую имени.

Добавим пользователя в группу wheel – она похожа на группу sudo в современных версиях Linux, позволяет выполнять команды от имени суперпользователя.

Оболочку назначим по умолчанию – sh. Она похожа на bash, но в bash больше возможностей. По умолчанию во FreeBSD нет bash, хотя возможность поставить есть.

Аутентификацию по паролю мы будем использовать, пустой пароль – нет. На практике может быть интересна возможность сгенерировать рандомный пароль, но его понадобится записать. Для установки же в учебных целях мы создадим пароль сами.

Блокировать созданного пользователя не надо.

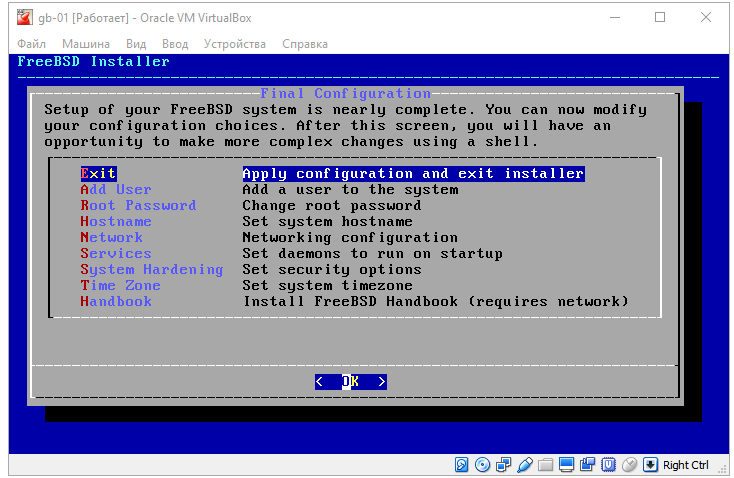


Проверяем, все ли правильно. Если да, то **yes.**

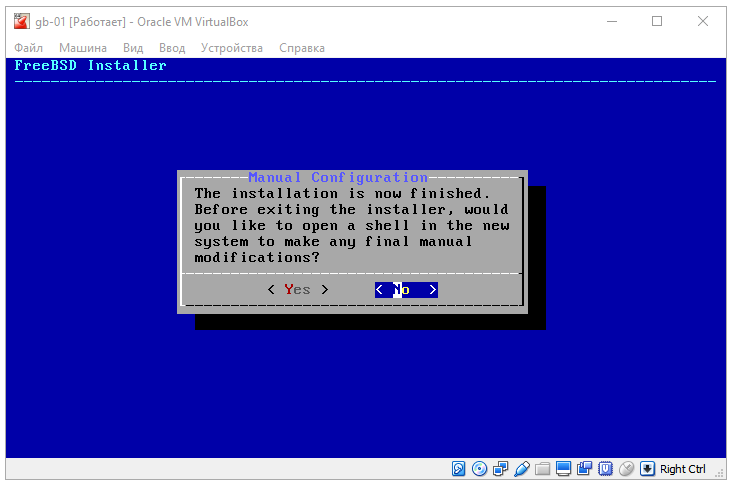
Еще одного пользователя создавать не будем.

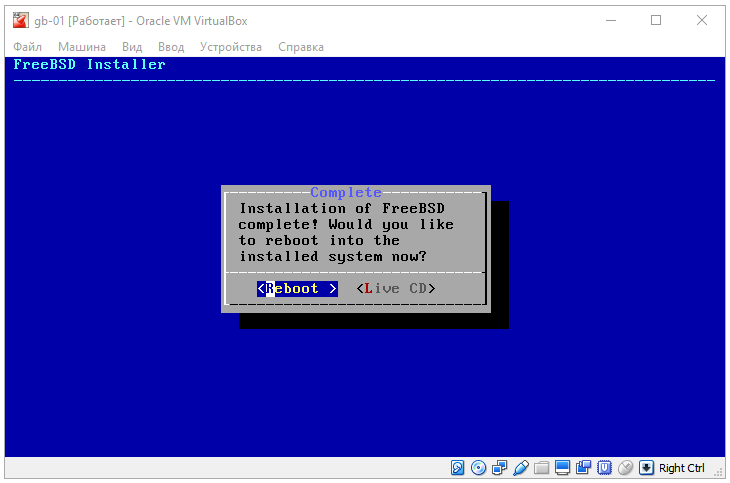
После этого у нас появляется еще одно меню.

В нашем случае выбираем «Применить конфигурацию и выйти».

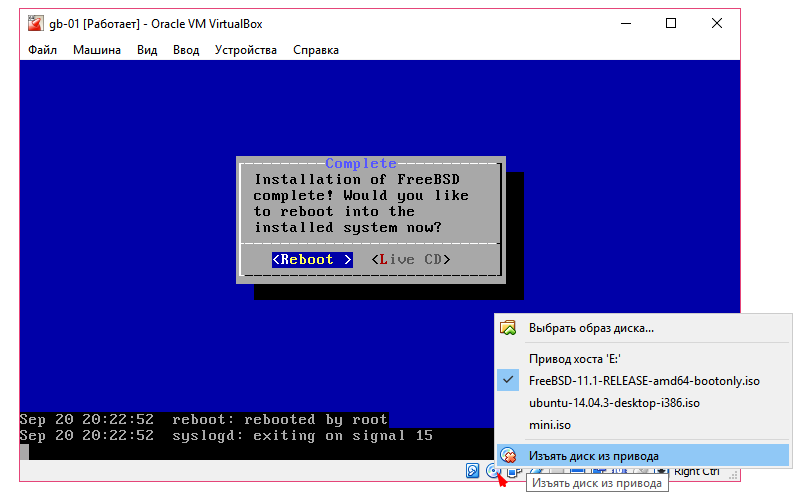


Можно запустить shell и выполнить дополнительные действия при необходимости. Нам это не нужно.





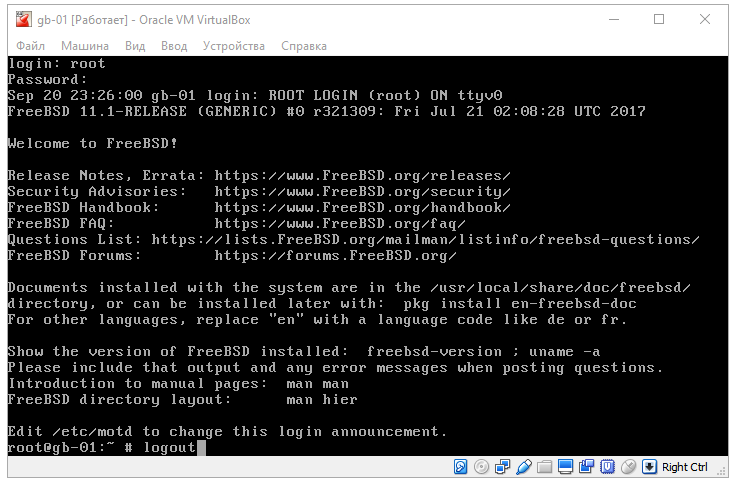
Жмем **Reboot,** нажимаем правый Ctrl, кликаем на значок CD в нижней части окна правой кнопкой мыши и кликаем **Изъять диск из привода.**



После этого система загрузится уже с жесткого диска – установленная нами.

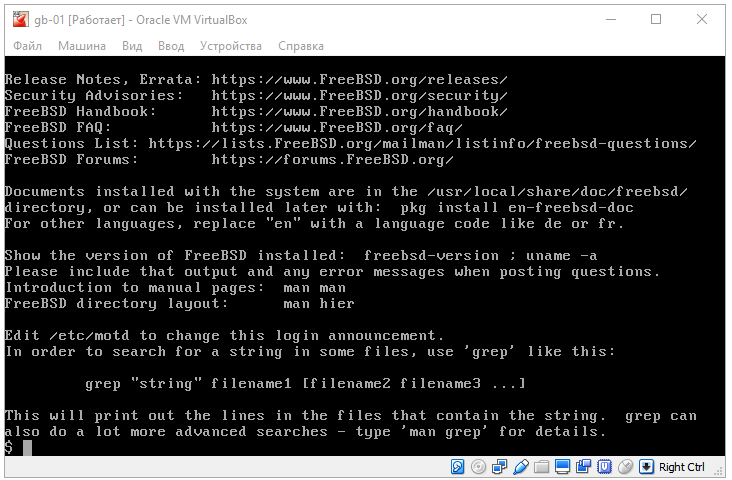
# Знакомство с FreeBSD

Загрузились.



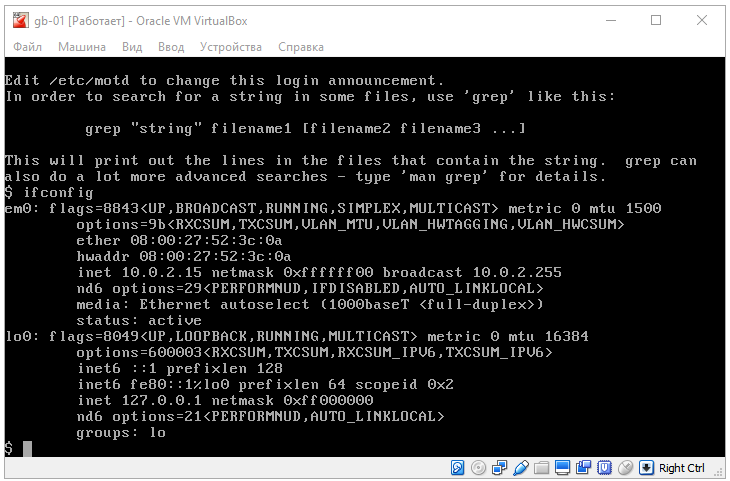
Проверяем: зашли рутом и вышли.

Зайдем пользователем (user).



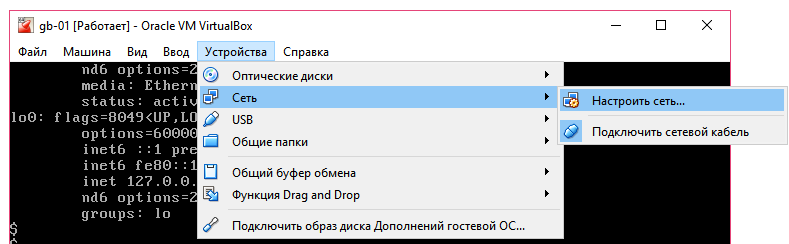
## Подключаемся по ssh

C помощью ifconfig посмотрим наш IP-адрес.

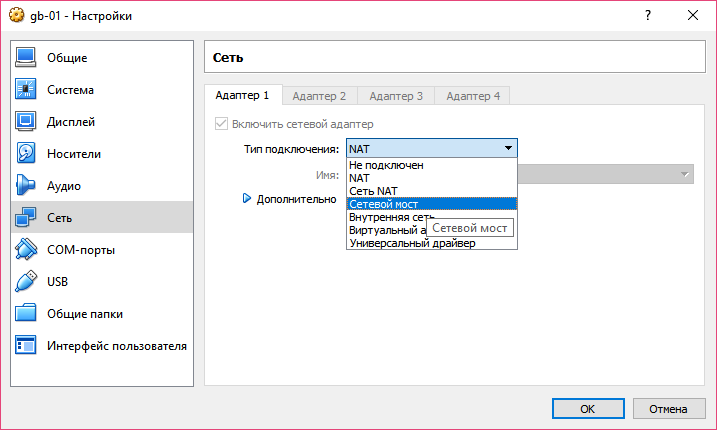


Возможно, подключения с хостовой машины к гостевой по такому IP-адресу не будет.

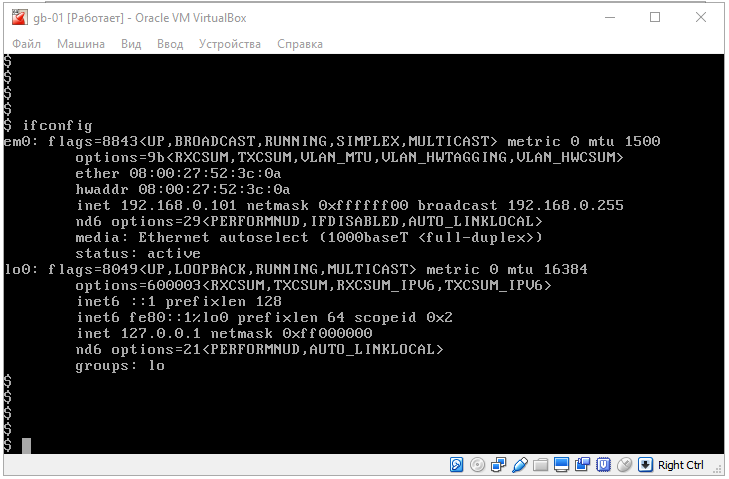
Смотрим настройки.



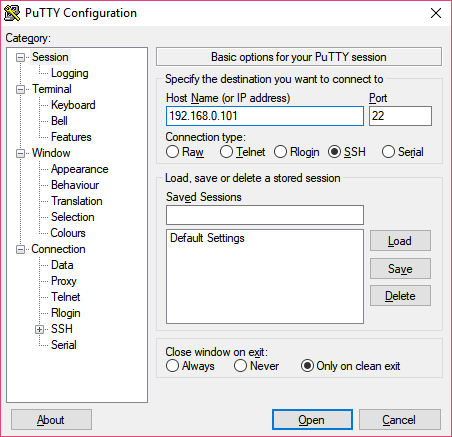
Если у нас был NAT, выбираем **Сетевой мост.**

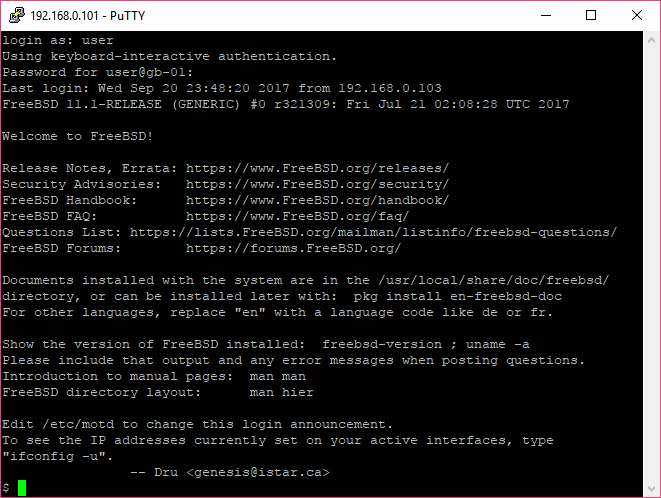


После этого гостевая машина будет находиться в той же сети, что и хостовая. При этом IP-адрес будет получен от вашего dhcp-сервера (например, от домашнего маршрутизатора). Проверяем:



Теперь мы можем подключиться по ssh с помощью любого клиента, будь то ssh (в Linux и Mac OS X), или putty, или другой любой клиент в Windows:





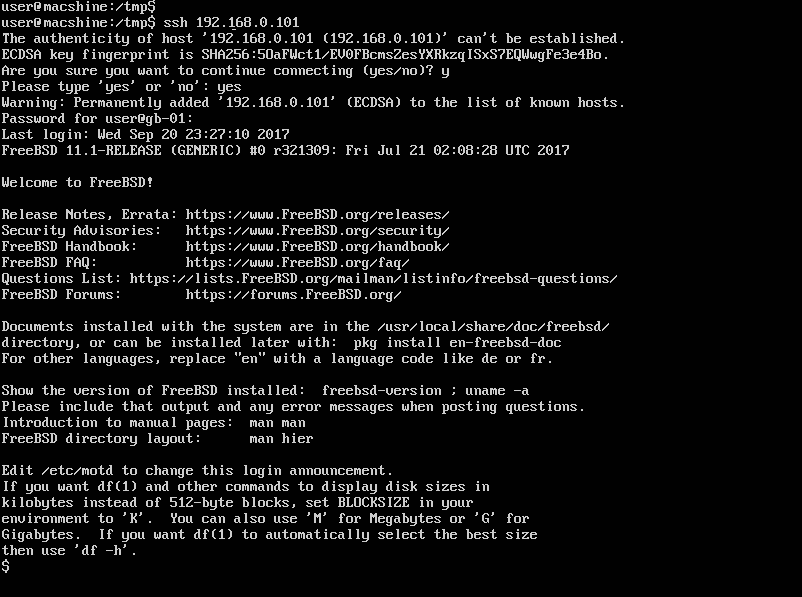
(пример с использованием ssh-клиента putty)

или

|  |
| --- |
| $ ssh 192.168.0.101 |

или даже

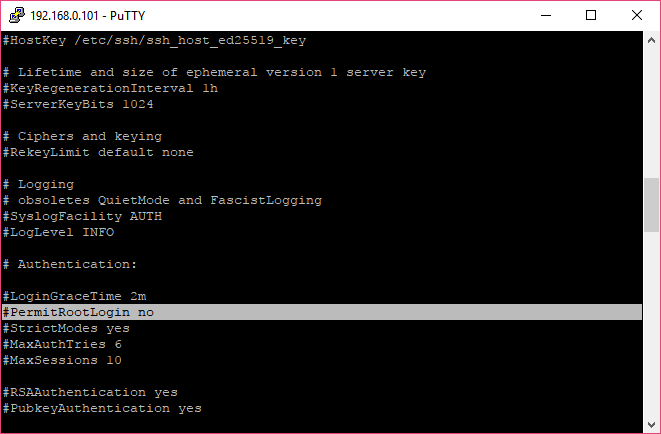
|  |
| --- |
| $ ssh user@192.168.0.101 |



(пример с использованием ssh в Ubuntu для подключения к виртуальной машине с FreeBSD).

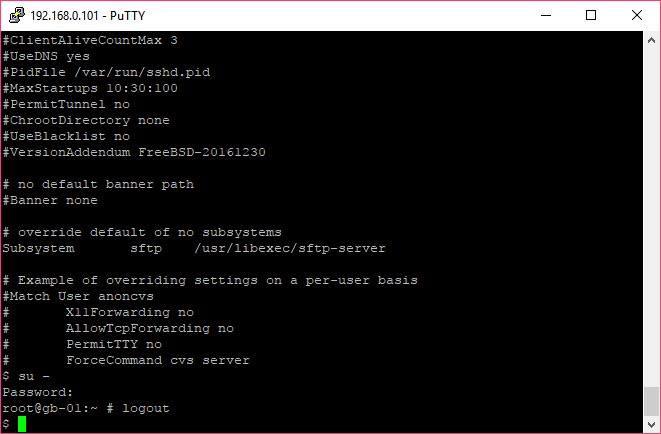
Мы сразу зашли пользователем. Стоит отметить, что попытка зайти пользователем root во FreeBSD не получится (хотя будет предложено ввести пароль, и даже если вы введете правильный, войти вы не сможете): в настройках ssh возможность входа рутом запрещена по умолчанию.

|  |
| --- |
| $ cat /etc/ssh/sshd\_config |



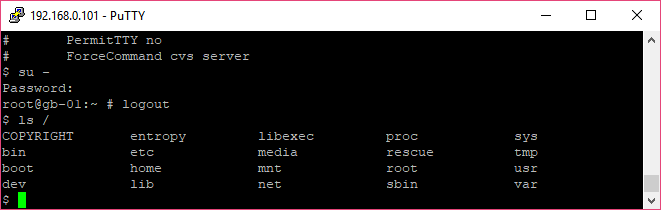
Видим, что, несмотря на то, что запись закомментирована, во FreeBSD именно это значение является значением по умолчанию.

Поэтому для работы следует использовать su.

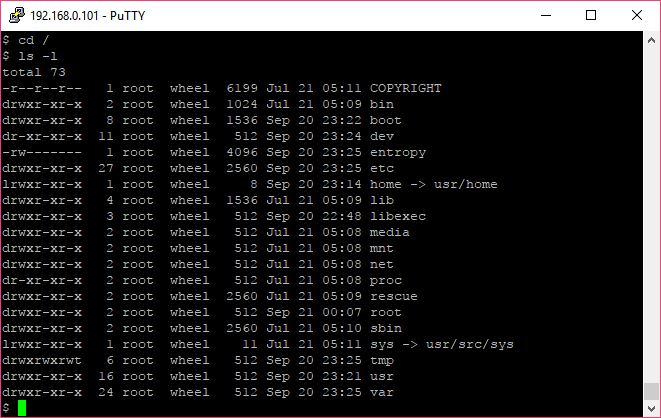


## Файловая система во FreeBSD

Файловая система во FreeBSD похожа на файловую систему в Linux, что неудивительно, ведь обе системы являются UNIX-подобными.

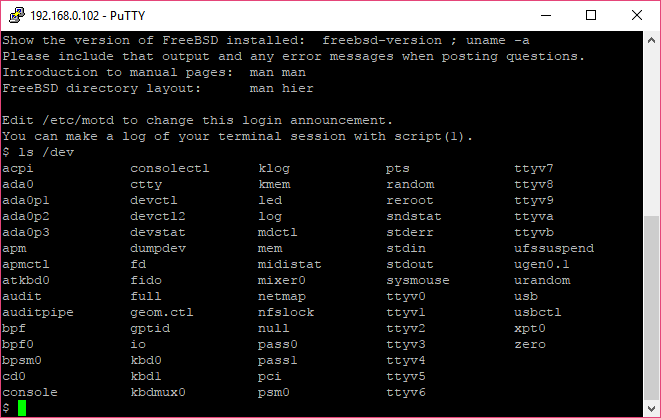


Более подробно:

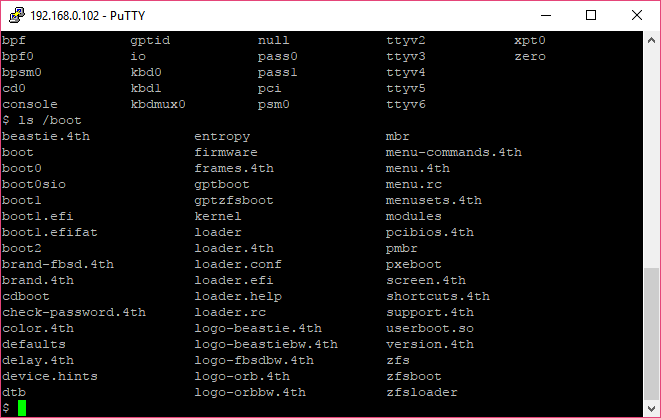


Вы видите знакомые директории bin, boot, dev, etc, lib, точно такие же media и mnt и т. д. Но различия есть. Если /etc похож по содержанию на аналогичный каталог в Linux (но не во всем), то /boot и /dev очень сильно отличаются. В отличие от Linux, где все настройки хранятся в /etc, во FreeBSD в /etc находятся только системные настройки. Все пользовательские данные находятся в /usr, отсюда мы видим, что /home является символической ссылкой на /usr/home а /sys – на /usr/src/sys

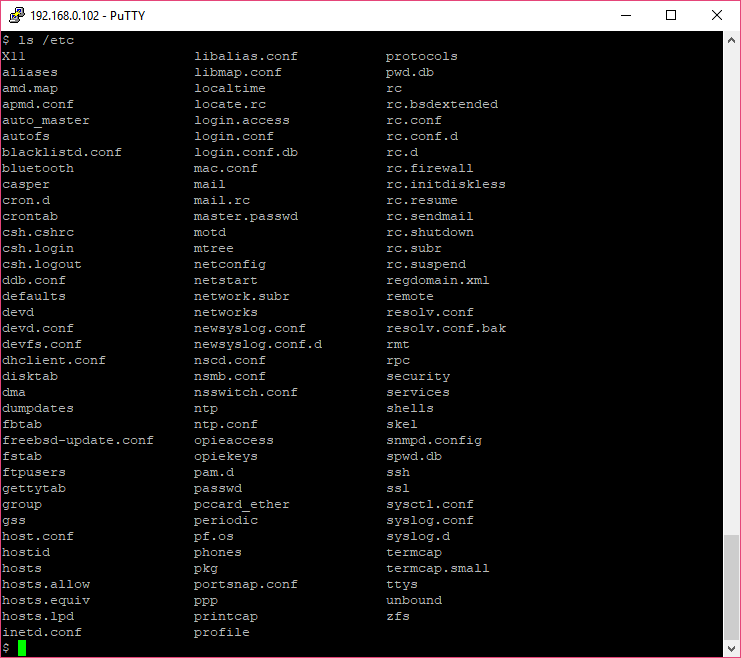
Директория /dev также содержит файлы устройств, но называются они иначе, чем в Linux.



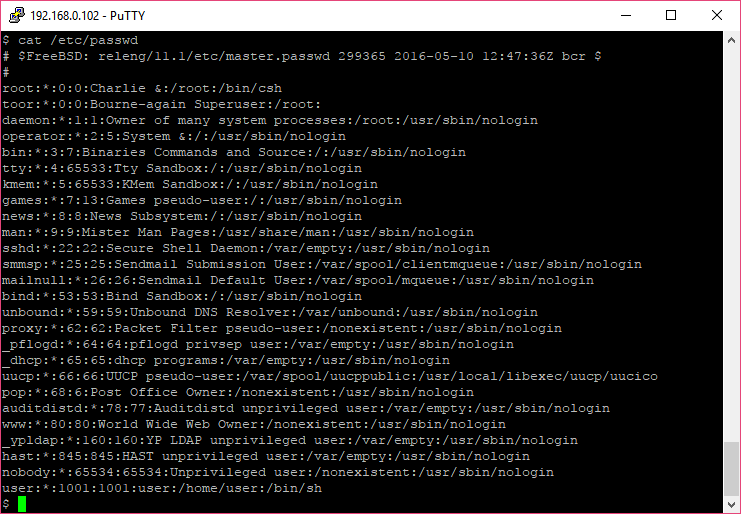
Отличается и boot. Если в Linux используется, как правило, GRUB или LILO, во FreeBSD используется loader.



Рассмотрим директорию /etc.



Многие файлы окажутся знакомыми, например, /etc/passwd.

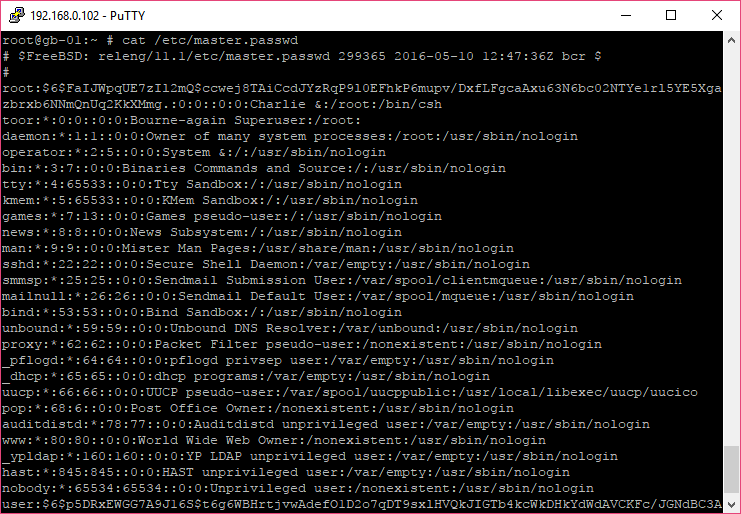


Здесь мы видим знакомого нам root с UID=0, заданного нами пользователя с UID=1001 и стандартных псевдопользователей.

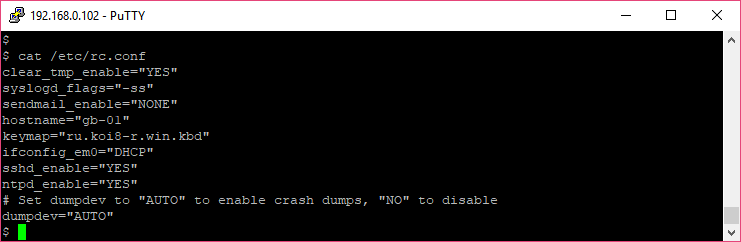
Тем не менее есть и отличия. Кроме root появился еще один пользователь, toor. Технически это тот же пользователь, что и root, так как система работает не с именами, а с UID. У toor UID также равен 0, да и само имя toor – это root наоборот. Нужен этот пользователь для работы с другими оболочками, отличными от csh. Таким образом, для режима восстановления в однопользовательском режиме может использоваться root, а для администрирования с использованием нестандартной оболочки – toor. Сразу войти под этим пользователем не получится, так как у него не установлен пароль.

Звездочка во второй колонке означает, что пароли хранятся в другом файле. Называется он не /etc/shadow, а /etc/master.passwd.

Как видим, по структуре он больше похож на /etc/passwd, чем на линуксовый /etc/shadow.

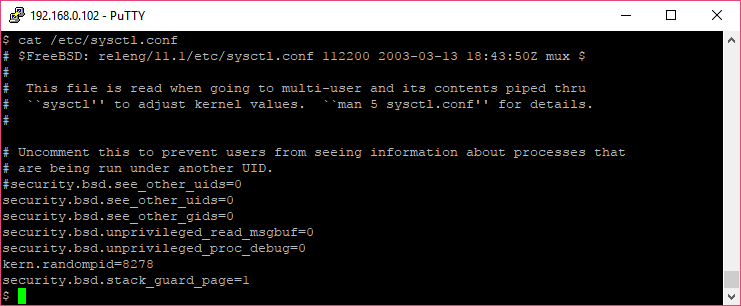


Файл /etc/rc.conf хранит настройки, знакомые нам еще по установке.

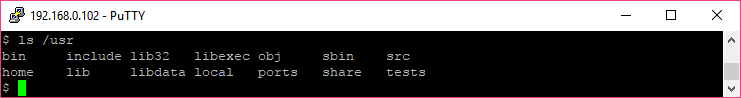


Именно здесь мы видим указание на запуск демонов (sshd, nntpd).

Отличается и формат /etc/sysctl.conf.

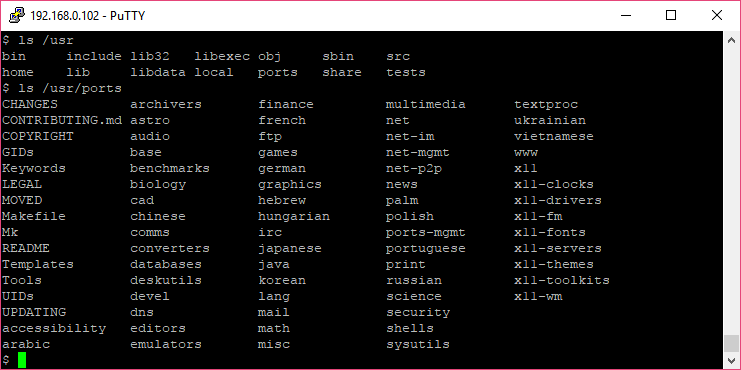


Директория /usr повторяет структуру корневого каталога, но служит только для пользовательских файлов.



## Система портов, pkg, man

Посмотрим с помощью ls /usr/ports (или ls -l /usr/ports).



Здесь находятся порты, которые разбиты на категории. Каждая директория – категория с соответствующим программным обеспечением.

В каждой содержится Makefile для сборки. Из минусов портов – скорость сборки.

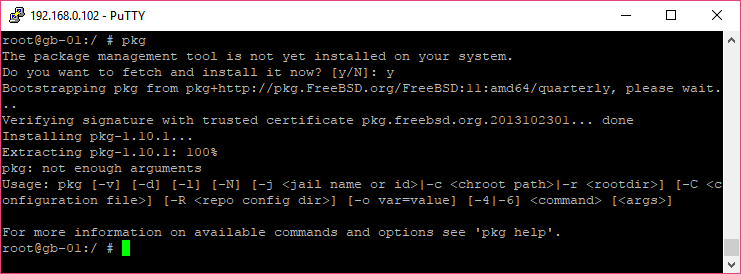
Также во FreeBSD появились пакеты (похожие на аналогичные в Linux).

Мы используем source-based дистрибутив, и нам понадобятся зависимости (gcc, perl, библиотеки).

Один из способов установить все необходимое – поставить знакомый многим Midnight Commander (mc). Даже если им не пользоваться, он потянет необходимые для дальнейшей работы зависимости.

Установку из портов мы рассмотрим на следующем занятии, а пока установим его через pkg.

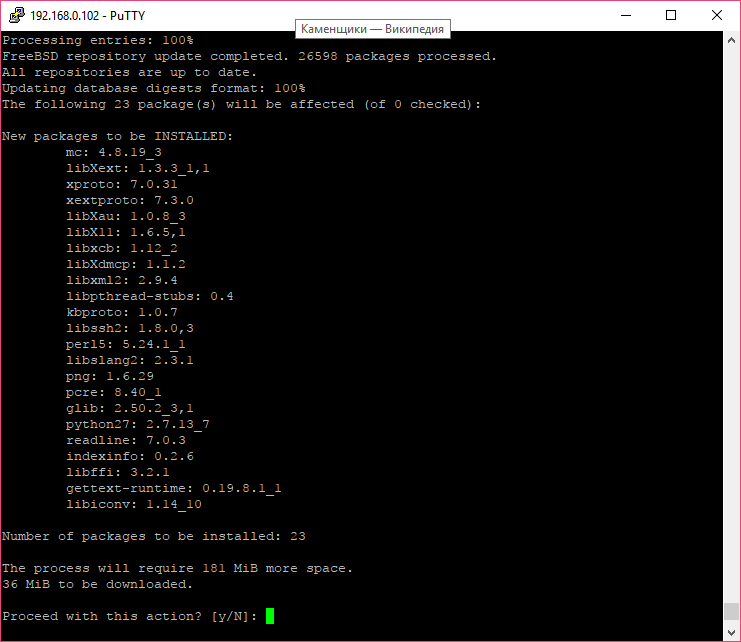
При первом запуске нам необходимо скачать базу с пакетами.



Потребуется некоторое время.

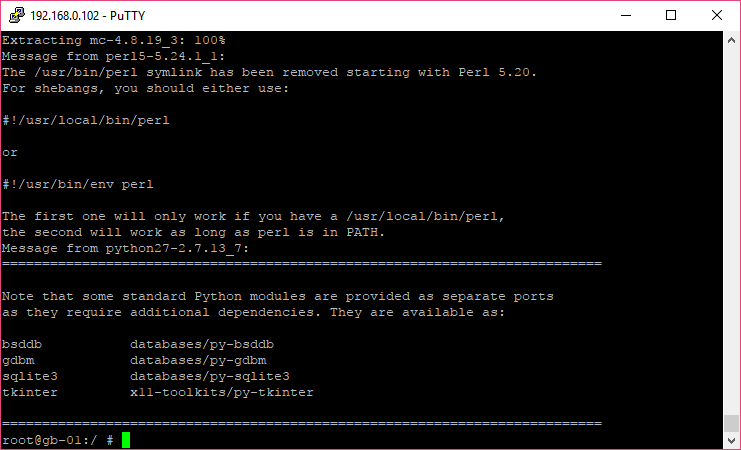
Теперь установим mc:

|  |
| --- |
| # pkg install mc |



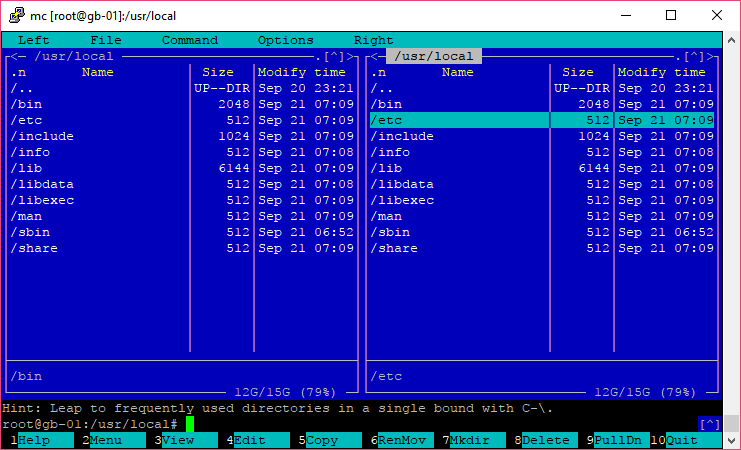
Видим все зависимости, которые понадобятся для работы. Нажимаем **y.**

Снова ждем, пока программа скачается и установится.



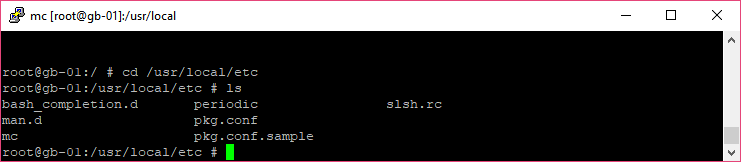
Теперь можно запустить mc

|  |
| --- |
| $ mc |



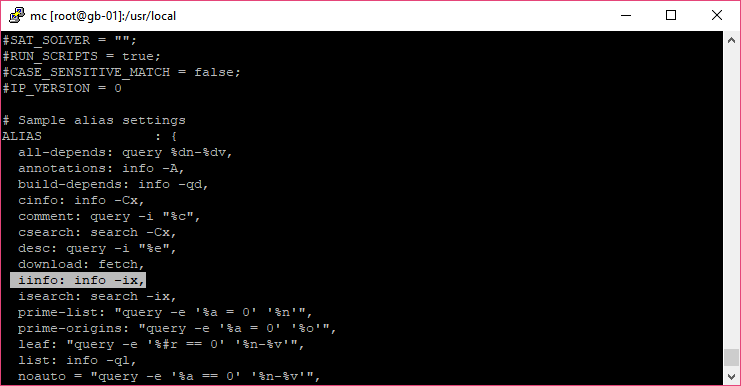
Видим, что в /usr/local появилась директория /etc.

Выходим (F10) и посмотрим содержимое.

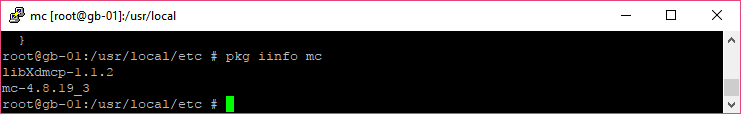


Заметим, что rc.d пока отсутствует, так как мы не устанавливали демонов.

Можем посмотреть pkg.conf (cat pkg.conf или less pkg.conf).



Обратите внимание на алиас. Например:



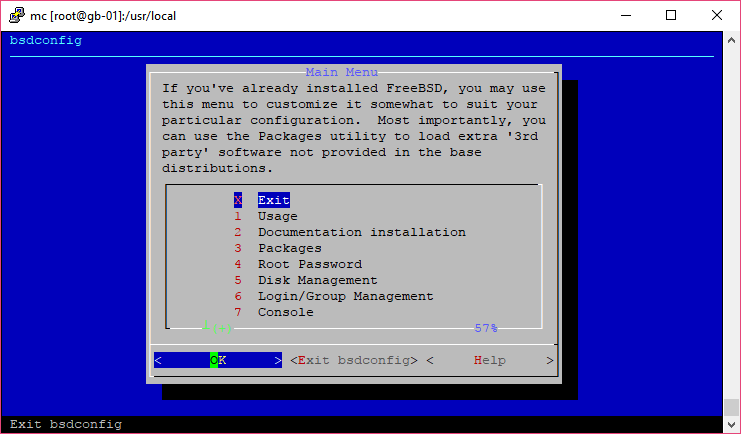
Имеется man.

Посмотрите самостоятельно:

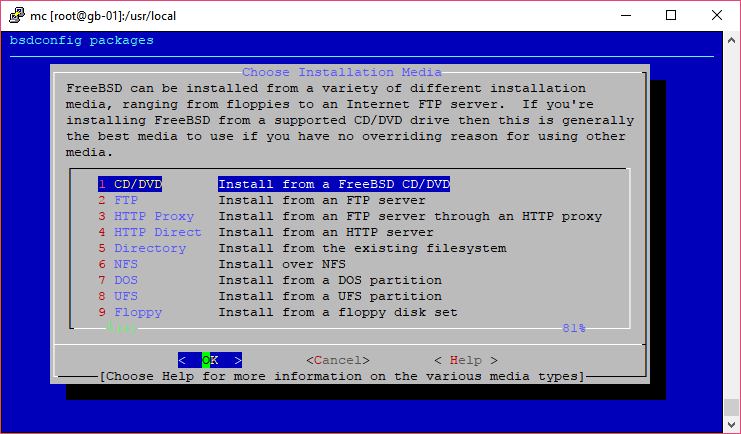
|  |
| --- |
| $ man mc $ man pkg |

Имеется псевдографическая утилита для администрирования bsdconfig (раньше именовалась sysconfig).

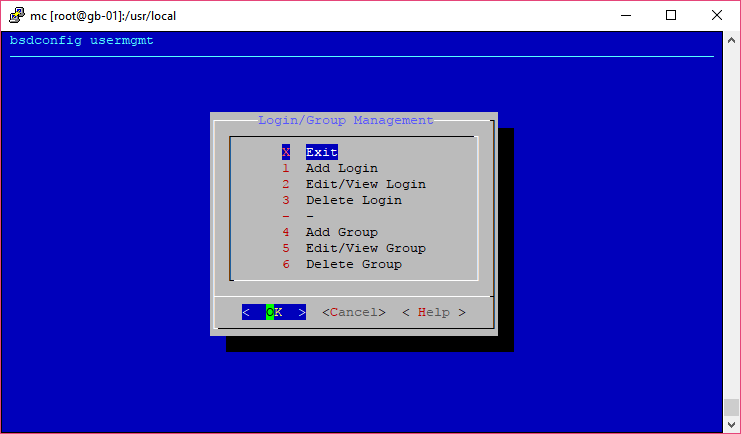
|  |
| --- |
| # bsdconfig |



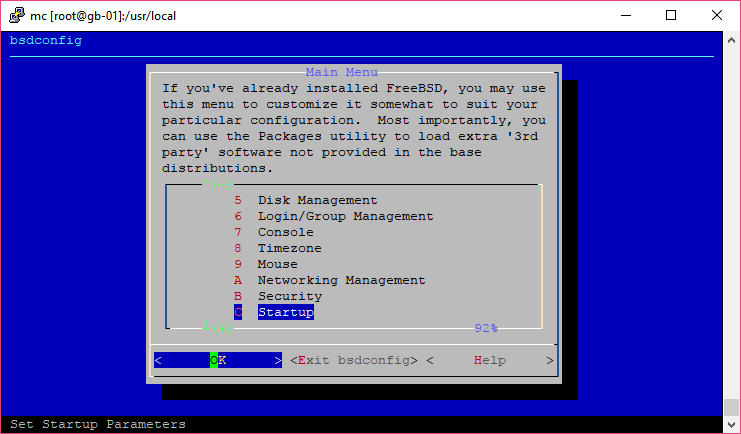
Можно работать с пакетами,

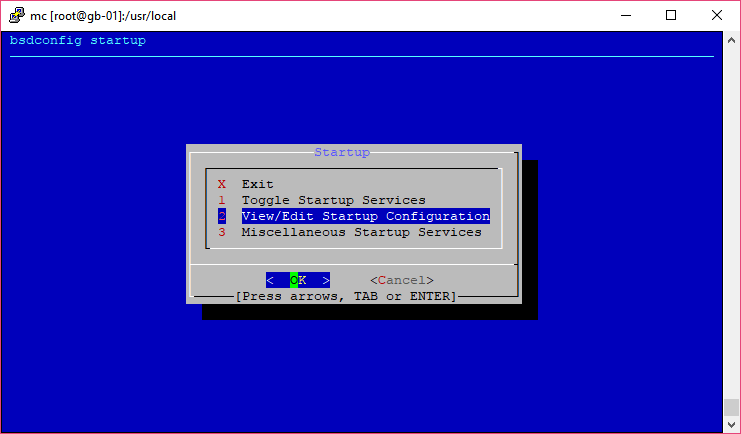


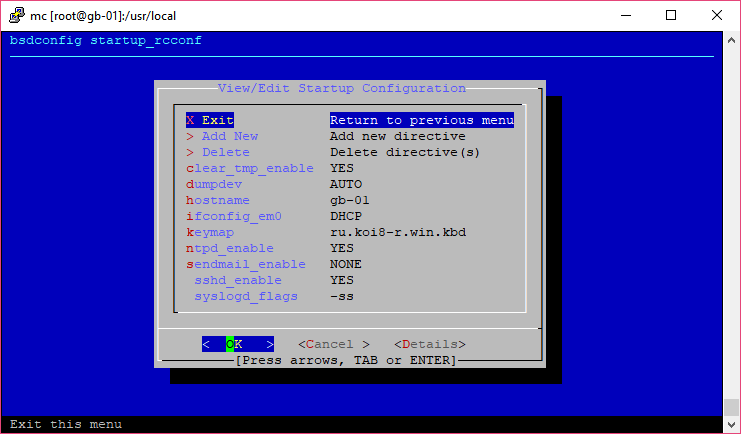
пользователями и группами,



остальными необходимыми настройками (клавишей «вниз» прокручиваем ниже и жмем Enter).





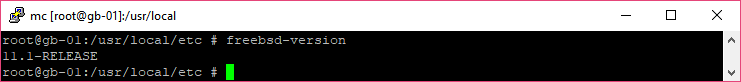


На самом деле мы работаем с конфигурационными файлами, хранящимися в /etc.

## Обновление BSD

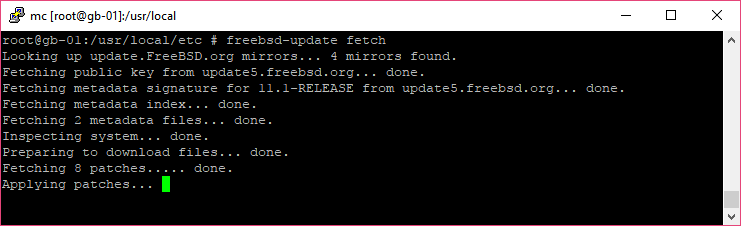
Посмотреть версию:

|  |
| --- |
| # freebsd-version |
|  |

Сначала получим метаданны

е:

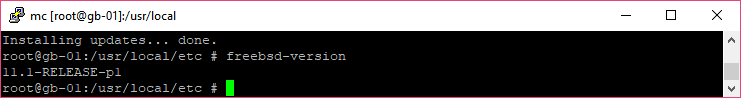
|  |
| --- |
| # freebsd-update fetch |



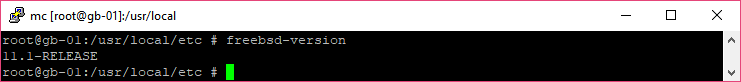
Установим обнония:

|  |
| --- |
| # freebsd-update install |

Не забудьте посмотреть версию:

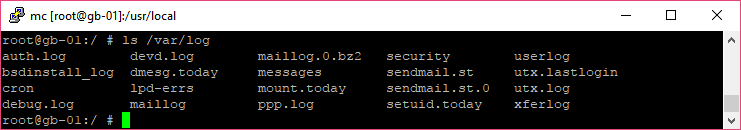


|  |
| --- |
| # freebsd-version |

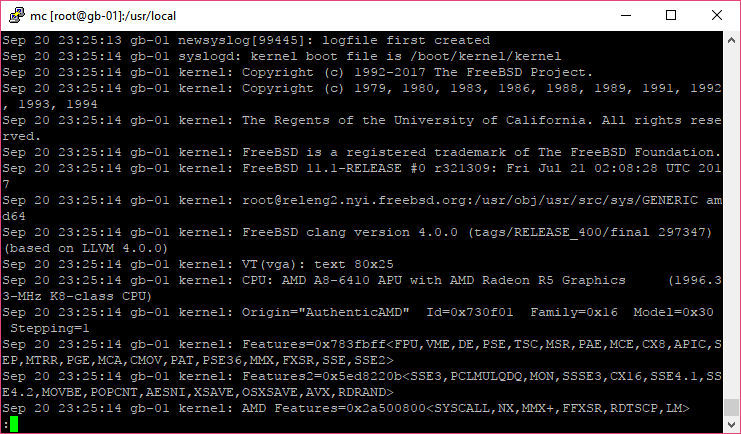


Просмотр логов.

Логи хранятся в /var/log.



|  |
| --- |
| # less /var/log/messages |



# Домашнее задание

1. Установить FreeBSD 11.
2. С помощью pkg установить mc.
3. Обновить FreeBSD.

# Дополнительные материалы

1. <https://www.freebsd.org/>
2. <https://www.freebsd.org/doc/handbook/>
3. <https://wiki.freebsd.org/FrontPage>
4. <https://www.virtualbox.org/>
5. <http://www.kegel.com/c10k.html>

# Используемая литература

Для подготовки данного методического пособия были использованы следующие ресурсы:

1. <https://www.freebsd.org/doc/ru/books/faq/security.html>
2. <https://www.freebsd.org/doc/ru/books/handbook/index.html>
3. <http://freebsd-image-gallery.netcode.pl/_daemonette/bsd-devilettes.jpg>
4. <http://freebsd-image-gallery.netcode.pl/?gallery=Daemonette>
5. <https://kb.vmware.com/selfservice/search.do?cmd=displayKC&externalId=1001675>
6. <http://www.ekzorchik.ru/2016/12/virtualbox-does-not-see-the-x64-templates/>
7. [https://help.qnap.ru/hc/ru/articles/214541045-Как-включить-поддержку-аппаратной-виртуализации-Intel-VTx-AMD-SVM-в-сетевом-хранилище-QNAP-](https://help.qnap.ru/hc/ru/articles/214541045-%D0%9A%D0%B0%D0%BA-%D0%B2%D0%BA%D0%BB%D1%8E%D1%87%D0%B8%D1%82%D1%8C-%D0%BF%D0%BE%D0%B4%D0%B4%D0%B5%D1%80%D0%B6%D0%BA%D1%83-%D0%B0%D0%BF%D0%BF%D0%B0%D1%80%D0%B0%D1%82%D0%BD%D0%BE%D0%B9-%D0%B2%D0%B8%D1%80%D1%82%D1%83%D0%B0%D0%BB%D0%B8%D0%B7%D0%B0%D1%86%D0%B8%D0%B8-Intel-VTx-AMD-SVM-%D0%B2-%D1%81%D0%B5%D1%82%D0%B5%D0%B2%D0%BE%D0%BC-%D1%85%D1%80%D0%B0%D0%BD%D0%B8%D0%BB%D0%B8%D1%89%D0%B5-QNAP-)