

Севастопольский государственный университет

Методические указания  
к выполнению лабораторных работ  
по дисциплине «Основы облачных  
и GRID-технологий»

А.Р. Умеров<sup>1</sup>

Е.Н. Мащенко<sup>2</sup>

25 февраля 2015 г.<sup>3</sup>

---

<sup>1</sup>admin@amet13.name

<sup>2</sup>elmachenko@mail.ru

<sup>3</sup>Дата последней правки документа

## Содержание

<b>1 Лабораторная работа №1</b>	<b>3</b>
1.1 Теоретические основы виртуализации . . . . .	3
1.2 Порядок выполнения работы . . . . .	7
1.3 Контрольные вопросы . . . . .	8
<b>2 Лабораторная работа №2</b>	<b>9</b>
2.1 Теоретические основы облачных вычислений . . . . .	9
2.2 Порядок выполнения работы . . . . .	11
2.3 Контрольные вопросы . . . . .	12
<b>Список литературы</b>	<b>13</b>
<b>A Пример создания виртуальной машины в Oracle VM VirtualBox</b>	<b>14</b>
<b>B Пример установки Debian 7 GNU/Linux в VirtualBox</b>	<b>21</b>
<b>C Пример установки ownCloud Server в Debian 7 GNU/Linux</b>	<b>33</b>
<b>D Настройка подключения ownCloud Client к серверу</b>	<b>37</b>

## 1 Лабораторная работа №1.

### Знакомство с виртуализацией. Система виртуализации Oracle VM VirtualBox

**Цель работы:** ознакомиться с основными понятиями виртуализации, системой виртуализации VirtualBox, научиться настраивать виртуальную машину (ВМ), совершать простейшие операции с ней, устанавливать операционную систему на ВМ.

#### 1.1 Теоретические основы виртуализации

Виртуализация — предоставление наборов вычислительных ресурсов или их логического объединения, абстрагированное от аппаратной реализации, и обеспечивающее изоляцию вычислительных процессов.

Виртуализацию можно использовать в [1]:

- Консолидации серверов (позволяет мигрировать с физических серверов на виртуальные, тем самым увеличивается коэффициент использования аппаратуры, что позволяет существенно сэкономить на аппаратуре, электроэнергии и обслуживании);
- Разработке и тестировании приложений (возможность одновременно запускать несколько различных ОС, это удобно при разработке кроссплатформенного ПО, тем самым значительно повышается качество, скорость разработки и тестирования приложений);
- Бизнесе (использование виртуализации в бизнесе растет с каждым днем и постоянно находятся новые способы применения этой технологии, например, возможность безболезненно сделать снапшот<sup>1</sup> и быстро восстановить систему в случае сбоя);
- Организации виртуальных рабочих станций (так называемых «тонких клиентов»<sup>2</sup>).

Общая схема взаимодействия виртуализации с аппаратурой и программным обеспечением (ПО) представлена на рис. 1.

<sup>1</sup>Снапшот (англ. snapshot) — снимок состояния ВМ в определенный момент времени. Сюда входят настройки ВМ, содержимое памяти и дисков

<sup>2</sup>Тонкий клиент (англ. thin client) — бездисковый компьютер-клиент в сетях с клиент-серверной или терминальной архитектурой, который переносит все или большую часть задач по обработке информации на сервер

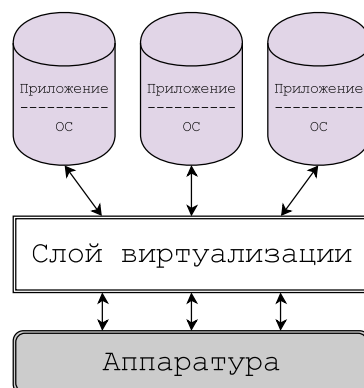


Рис. 1: Схема взаимодействия виртуализации с аппаратурой и ПО

Взаимодействие приложений и операционной системы (ОС) с аппаратным обеспечением осуществляется через абстрагированный слой виртуализации.

Существует несколько подходов организации виртуализации:

- Эмуляция оборудования (QEMU, Bochs, Dynamips);
- Полная виртуализация (KVM, HyperV, VirtualBox);
- Паравиртуализация (Xen, L4, Trango);
- Виртуализация уровня ОС (LXC, OpenVZ, Jails, Solaris Zones).

Эмуляция оборудования является самым сложным и трудоемким методом виртуализации (рис. 2). Однако, большим преимуществом для данного метода является, к примеру, управление неизменной ОС, предназначенной для PowerPC на системе с ARM процессором.

В случае полной виртуализации поверх уже установленной ОС, устанавливается программа-гипервизор<sup>1</sup>, которая осуществляет взаимосвязь между гостевыми ОС и хост-компьютером (рис. 3).

Преимуществом технологии полной виртуализации является установка различных ОС, а недостатком — меньшая производительность, за счет накладных расходов на гипервизор, а также понижение скорости работы с подсистемой ввода/вывода из-за необходимости изоляции.

Паравиртуализация имеет некоторые сходства с полной виртуализацией. В данном методе также используется гипервизор для разделения

<sup>1</sup>Гипервизор (англ. hypervisor)— программа или аппаратная схема, позволяющая одновременное, параллельное выполнение нескольких ОС на одном и том же компьютере, обеспечивает изоляцию операционных систем друг от друга

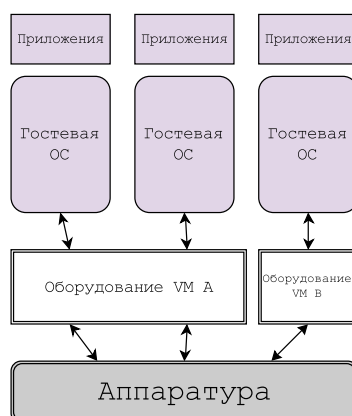


Рис. 2: Эмуляция оборудования моделирует аппаратные средства



Рис. 3: Полная виртуализация использует гипервизор

доступа к аппаратуре, но объединяется код, касающийся виртуализации, в ОС [2] (рис. 4).

Недостатком паравиртуализации является необходимость изменения гостевой ОС для гипервизора, однако таким образом гораздо увеличивается производительность.

Виртуализация уровня операционной системы не нуждается в гипервизоре. Для ее работы необходимо модифицированное ядро на хост-системе с набором патчей и утилит для управления контейнерами<sup>1</sup> (рис. 5).

<sup>1</sup>Контейнер или VPS/VDS (англ. Virtual Private/Dedicated Server) — виртуальный выделенный сервер, эмулирует работу физического сервера

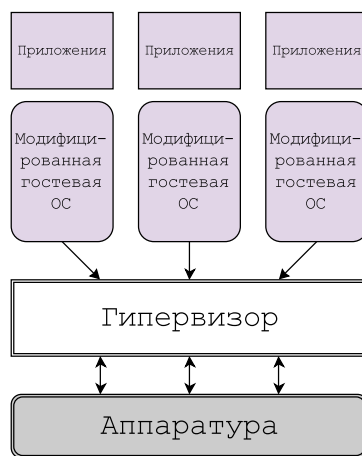


Рис. 4: Паравиртуализация разделяет процесс с гостевой ОС

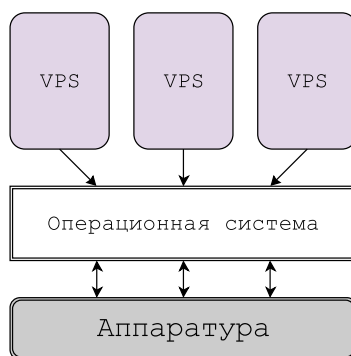


Рис. 5: Виртуализация уровня ОС изолирует серверы

За счет того, что контейнер напрямую взаимодействует с ядром, а не через гипервизор, обеспечивается максимальное быстродействие. Но, так как для всех контейнеров используется общее ядро, то нет возможности использовать разные ОС в контейнерах.

Oracle VM VirtualBox — система виртуализации, разработанная компанией Innotek в 2007 году, позже приобретена компанией Sun Microsystems. Ключевыми возможностями системы является кроссплатформенность, наличие графического интерфейса, локализация, поддержка аппаратной виртуализации, экспериментальное 3D-ускорение, поддержка различных образов жестких дисков, возможность установки дополнений гостевой ОС, например для корректной работы проброшенных USB-устройств или возможности изменения разрешения рабочего стола гостевой ОС.

На рис. 6 изображен пример одновременной работы двух виртуальных машин (Windows 7 и CentOS 7).

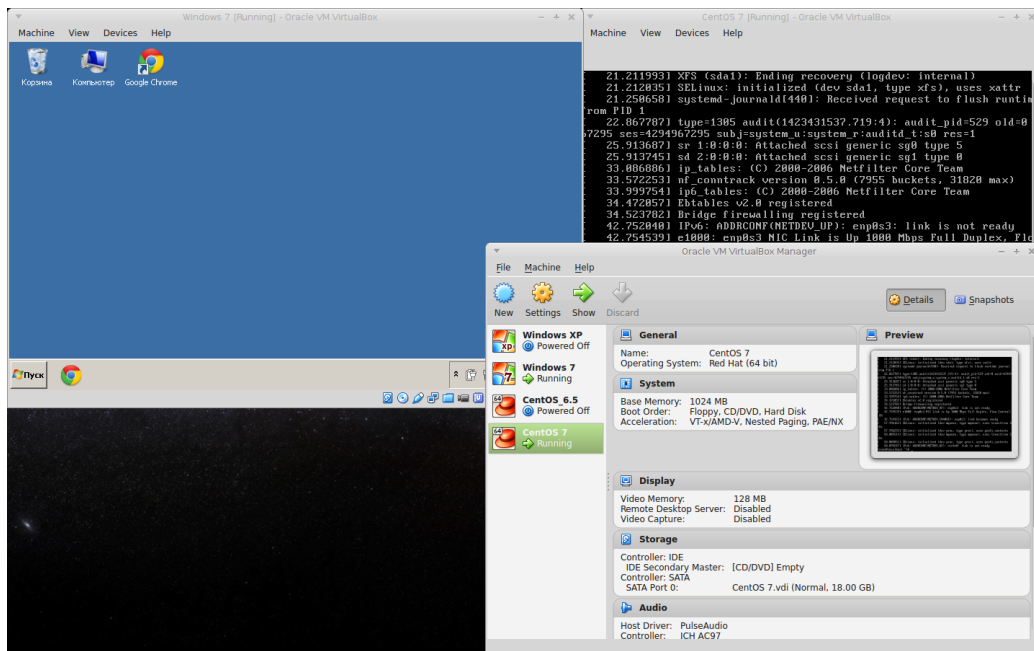


Рис. 6: Пример одновременной работы Windows 7 и CentOS 7

## 1.2 Порядок выполнения работы

1. Скачать<sup>1</sup> и установить Oracle VM VirtualBox;
2. Изучить интерфейс программы и создать<sup>2</sup> первую виртуальную машину на основе образа Debian 7 GNU/Linux<sup>3</sup>;
3. Подключить к виртуальной машине ранее скачанный образ Debian 7 GNU/Linux;
4. Загрузиться с подключенного образа и установить<sup>4</sup> дистрибутив на виртуальный жесткий диск;

<sup>1</sup><https://www.virtualbox.org/wiki/Downloads>

<sup>2</sup>Пример создания виртуальной машины описан в прил. А

<sup>3</sup><https://www.debian.org/distrib/>

<sup>4</sup>Процесс установки Debian 7 GNU/Linux описан в прил. В

5. Во время установки дистрибутива необходимо будет задать пароль суперпользователя, пароль: `toor`, пароль для локального пользователя можно задать произвольный, но лучше его запомнить;
6. С помощью команд `free`, `df`<sup>1</sup>, `cat /proc/cpuinfo`, `ifconfig`, посмотреть параметры виртуальной операционной системы;
7. С помощью команды `ping` проверить доступность виртуальной машины в сети как с хост-ноды, так и с соседних компьютеров сети;
8. С помощью SSH-клиента (например Putty<sup>2</sup>) подключиться по SSH к виртуальной машине;
9. Ответить на контрольные вопросы.

### 1.3 Контрольные вопросы

1. Какие типы виртуализации Вы знаете? В чем между ними различия?
2. К какому типу виртуализации относится система Oracle VM VirtualBox?
3. Как может использовать виртуализацию в работе веб-программист, системный администратор, сетевой инженер?
4. В чем состоит основное преимущество и недостаток полной виртуализации от контейнерной?
5. \*В чем различие между режимами сети NAT и сетевой мост (Network Bridge)?

---

<sup>1</sup>Команды `free` и `df` используют ключ `-h` для показа информации в удобном для человека виде

<sup>2</sup><http://www.chiark.greenend.org.uk/~sgtatham/putty/>



## 2 Лабораторная работа №2.

### Модель обслуживания SaaS (Software-as-a-Service) на примере развертывания облачного хранилища ownCloud

**Цель работы:** ознакомиться с основными моделями представления облачных услуг (SaaS, PaaS, IaaS), развернуть собственное облачное хранилище, доступное пользователям в пределах локальной сети.

#### 2.1 Теоретические основы облачных вычислений

Облачные вычисления («облака», Cloud computing) — модель предоставления вычислительных ресурсов, охватывающая все, начиная от приложений и до центров обработки данных (ЦОД), через Интернет при условии оплаты за фактическое использование. [3]

Одной из первых, кто стал внедрять услугу облачных вычислений стала компания Amazon, в то время (2002 г.) она еще являлась книжным Интернет-магазином, который впоследствии перерос, благодаря этим внедрениям, в одну из мощнейших технологических компаний. Уже в 2006 г. был запущен проект под названием Computing Cloud (Amazon EC2), после этого в 2009 г. компания Google представила Google Apps. После этих событий были сформированы общие понятия об облачных вычислениях, в частности выделены наиболее важные модели обслуживания и модели развертывания.

Различают три основные модели обслуживания:

1. Программное обеспечение как услуга (Software-as-a-Service, SaaS);
2. Платформа как услуга (Platform-as-a-Service, PaaS);
3. Инфраструктура как услуга (Infrastructure-as-a-Service, IaaS).

SaaS — модель, при которой не требуется приобретать, устанавливать, обновлять и поддерживать ПО, эту задачу берет на себя поставщик услуги. Кроме того, осуществить регистрацию и использование облачных приложений можно немедленно, приложения и данные приложений доступны с любого устройства, подключенного к Интернету. При поломке устройства данные не теряются, они хранятся в облаке, пользователю, как правило, доступны локальные настройки конфигурации приложения. Примерами моделей SaaS могут служить: почтовая служба

Gmail, Skype, CRM (система управления взаимоотношения с клиентами) и ERP (планирование ресурсов предприятия) системы и другие.

PaaS — модель, при которой пользователю предоставляется возможность использования облачной инфраструктуры для размещения базового ПО. В таком случае конфигурирование программного обеспечения целиком ложится на пользователя, предоставляется только платформа для развертывания ПО. Примером модели PaaS является предоставление услуги VPS, в рамках которой пользователю предоставляется виртуализированное окружение, как правило с «чистой» операционной системой, пригодной для развертывания любого приложения. Примеры: Digital Ocean, Microsoft Azure, Google App Engine и т.д;

IaaS — модель, при которой пользователю, помимо виртуальной доступна также и физическая инфраструктура облака. Потребитель облака имеет контроль над операционными системами, сетевыми сервисами. Данная модель подходит задачам, для которых характерны быстрое изменение нагрузки. Примером такой модели является аренда сервера или сетевой инфраструктуры в ДЦ, (Hetzner Online AG, leaseweb, Селектел и пр.). Некоторые крупные компании, такие как Google, IBM, Apple, eBay и другие размещают все свои сервера в своих же дата центрах.

Модели развертывания облака можно разделить на четыре вида:

1. Частное облако (private cloud);
2. Публичное облако (public cloud);
3. Общественное облако (community cloud);
4. Гибридное облако (hybrid cloud).

Частное облако предназначено для использования одной организацией, публичное — для широкой публики, общественное, как правило для сообщества или организации, гибридное облако состоит из двух или более различных облачных инфраструктур.

Основными преимуществами использования облачных технологий являются:

- Снижение расходов на закупку оборудования и построения центров обработки данных (ЦОД);
- Удобство использования приложений с большого количества устройств, в том числе и мобильных;
- Обеспечение надежности хранения данных, производительности приложений, за счет простоты использования ПО, мониторинга, балансировки нагрузки, миграции данных и т.д.

ownCloud — свободное веб-приложение, предназначенное для синхронизации данных между сервером и клиентами, как правило данными являются документы и медиаконтент. ownCloud является альтернативой таким облачным сервисам как Dropbox, Google Drive, Яндекс Диск, MEGA и др. Отличие состоит в том, что приложение можно развернуть на собственном сервере как для домашнего использования, так и для использования в организациях. Так как приложение распространяется бесплатно, имеет открытый исходный код<sup>1</sup> и периодически обрastaет новыми функциями (планировщик задач, календарь, фотогалерея, просмотрщик документов, гибкая аутентификация пользователей и другие) проект обрел популярность как у пользователей, так и у Open-source разработчиков.

Установка ownCloud происходит в несколько простых шагов. ownCloud Server доступен для платформ Windows и Linux, ownCloud Client также доступен для Windows, Linux, а также Mac.

## 2.2 Порядок выполнения работы

Выполнять работу рекомендуется группами студентов по 3-5 человек. В качестве сервера может использоваться виртуальная машина с установленным Debian 7, ранее установленная в лабораторной работе №1. Виртуальная машина должна иметь доступ в сеть Интернет для скачивания нужных пакетов.

1. Установить<sup>2</sup> ownCloud Server в виртуальную машину;
2. На хост-машине или на другом компьютере сети скачать<sup>3</sup> и установить ownCloud Client;
3. Подключить ownCloud Client к серверу во время первого запуска клиента, а также синхронизировать все данные с сервера<sup>4</sup>;
4. Проверить работу синхронизации данных между клиентом и сервером (создать, удалить, переместить файл или каталог);
5. Предоставить публичную ссылку на файл или каталог и проверить его доступность в пределах локальной сети;
6. Ознакомиться с дополнительными возможностями ownCloud.

---

<sup>1</sup><https://github.com/owncloud/>

<sup>2</sup>Пример установки ownCloud Server представлен в прил. С

<sup>3</sup><https://owncloud.org/install/>

<sup>4</sup>Скриншоты настройки ownCloud Client представлены в прил. D

### **2.3 Контрольные вопросы**

1. Каковы основные преимущества использования облачных технологий?
2. В чем состоит отличие SaaS от PaaS и IaaS?
3. В чем состоят преимущества ownCloud по сравнению с Dropbox или другими облачными хранилищами? Недостатки?
4. \*Почему Skype можно отнести к модели SaaS?

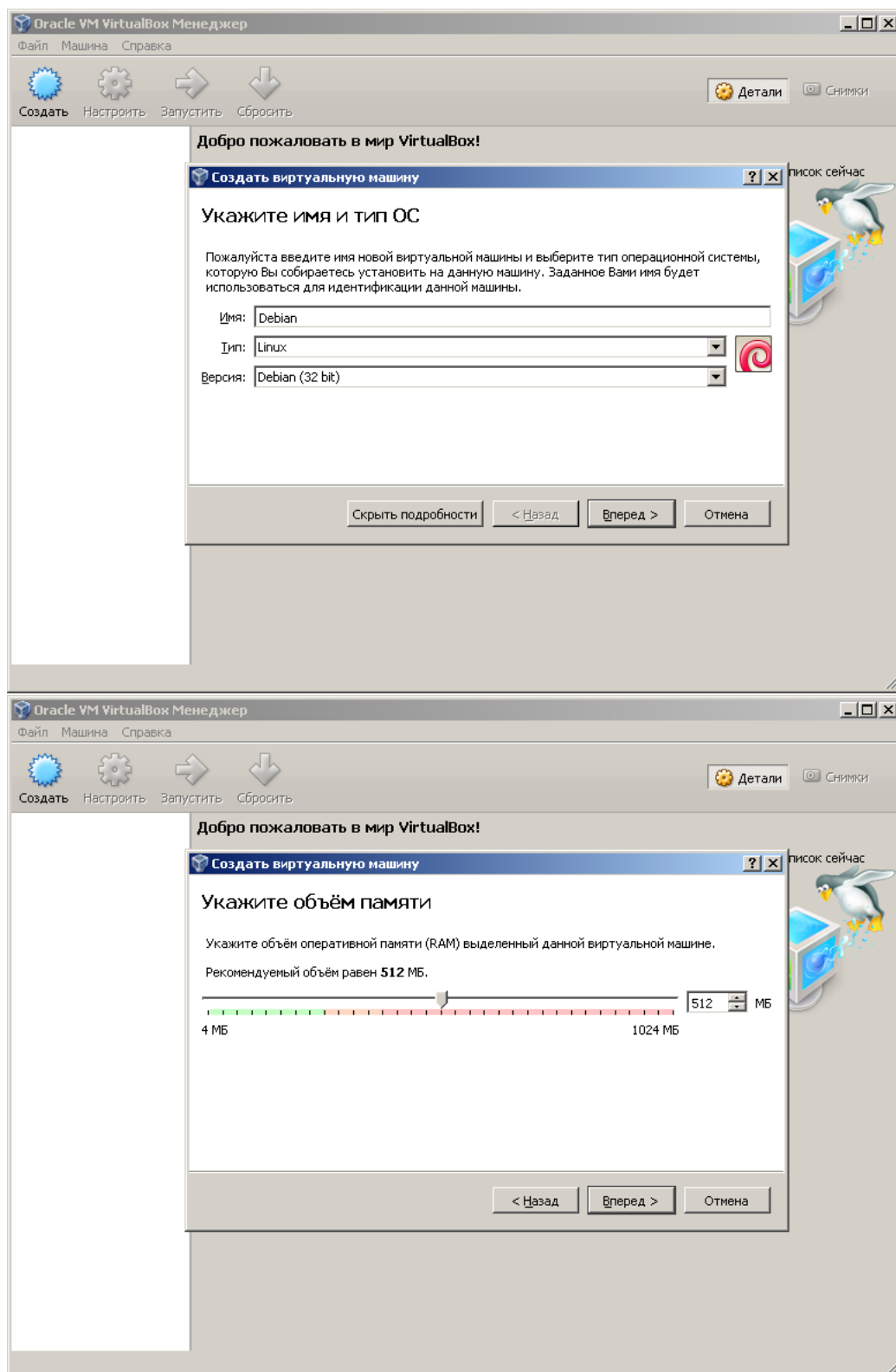
## Список литературы

- [1] А.Р. Умеров и Е.Н. Машенко. «Анализ технологий контейнерной виртуализации». В: *Мир компьютерных технологий. Материалы внутривузовской студенческой научно-технической конференции*. Севастополь: СевНТУ, 2014, с. 32.
- [2] М. Тим Джонс. «Виртуальный Linux. Обзор методов виртуализации, архитектур и реализаций». В: (2007). URL: <http://www.ibm.com/developerworks/ru/library/l-linuxvirt/>.
- [3] «Облачные вычисления». В: (2015). URL: <http://www.ibm.com/cloud-computing/ru/ru/learn.html>.

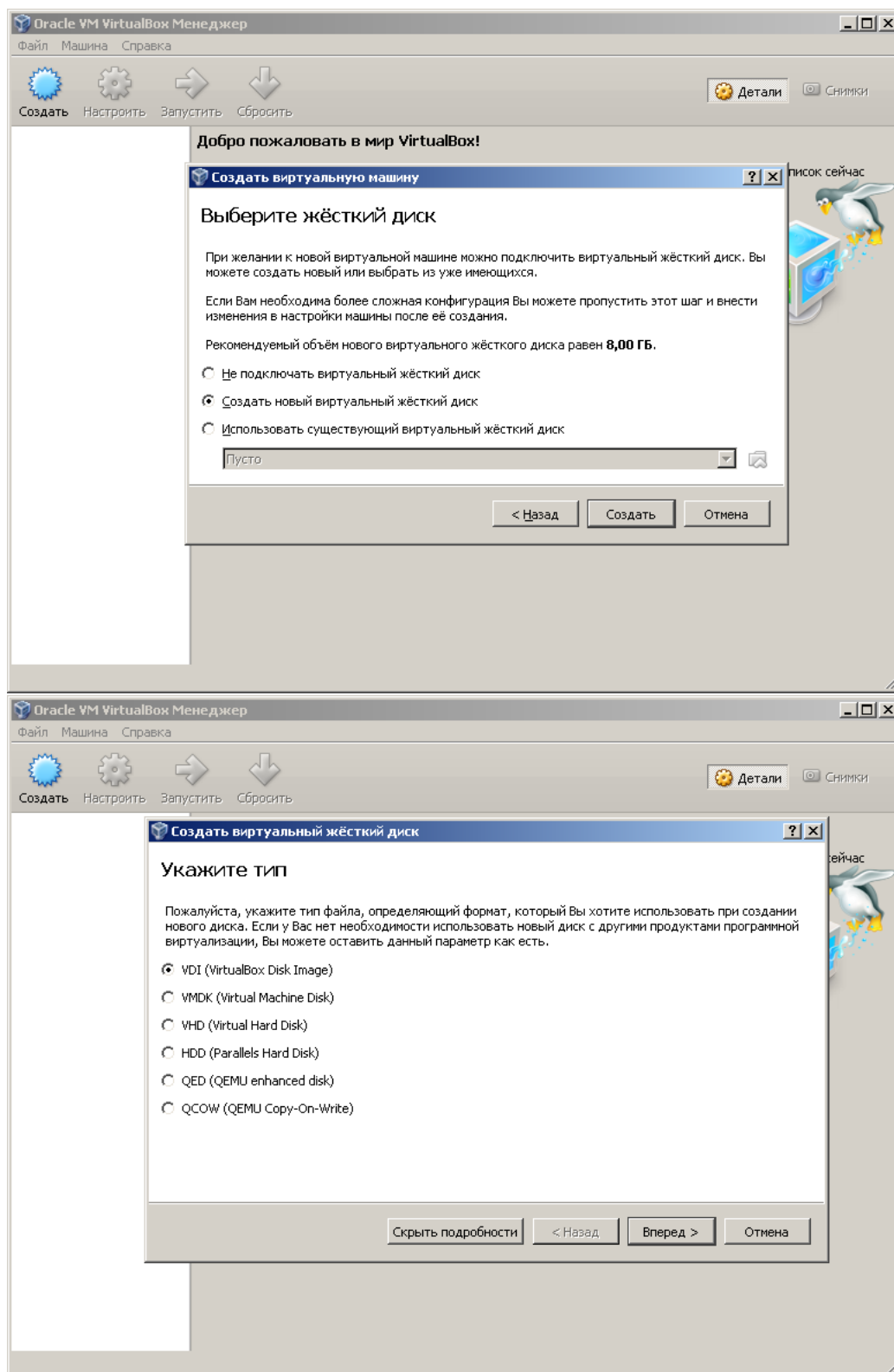
## А Пример создания виртуальной машины в Oracle VM VirtualBox



## 15 А. Пример создания виртуальной машины в Oracle VM VirtualBox

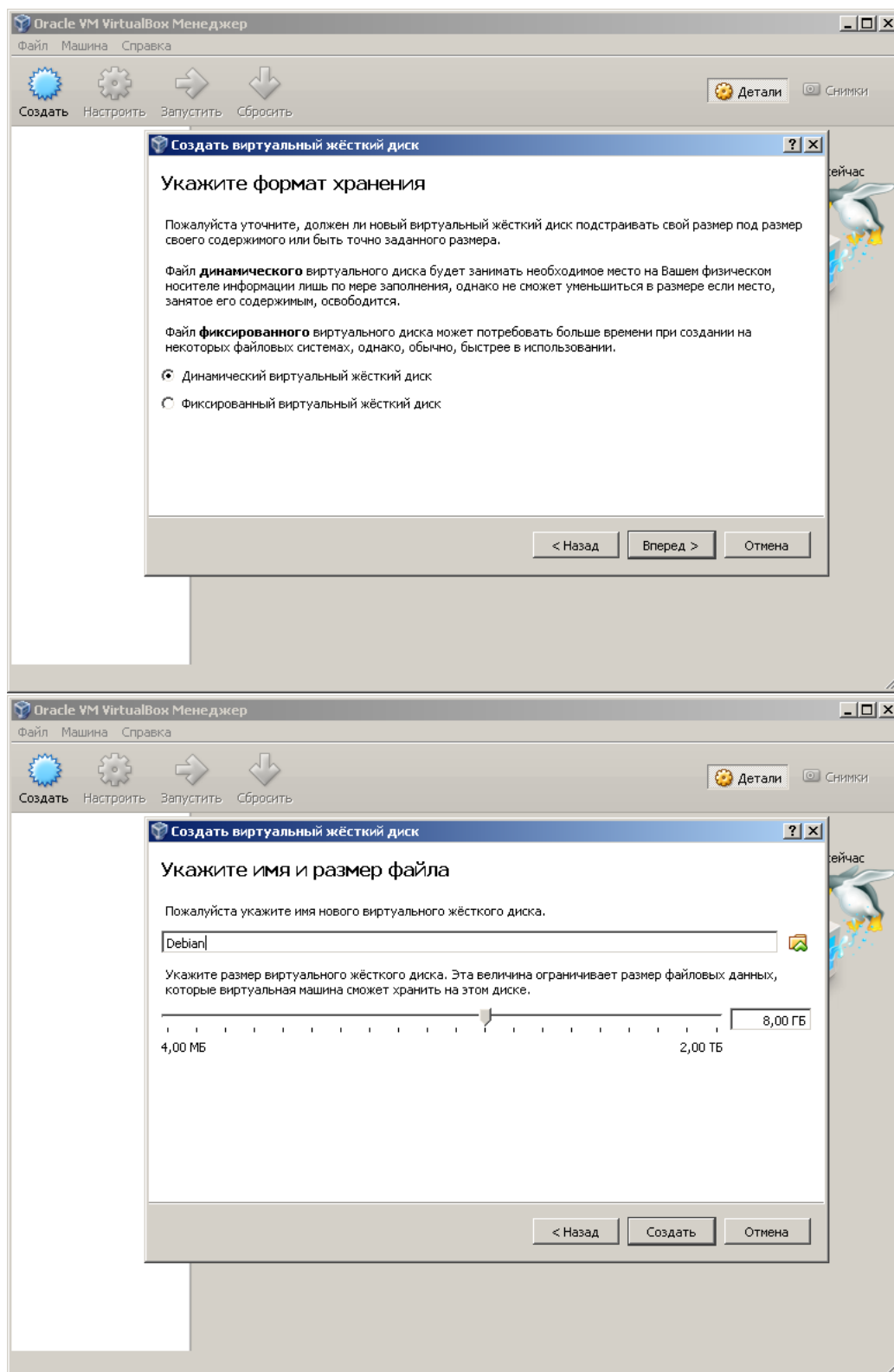


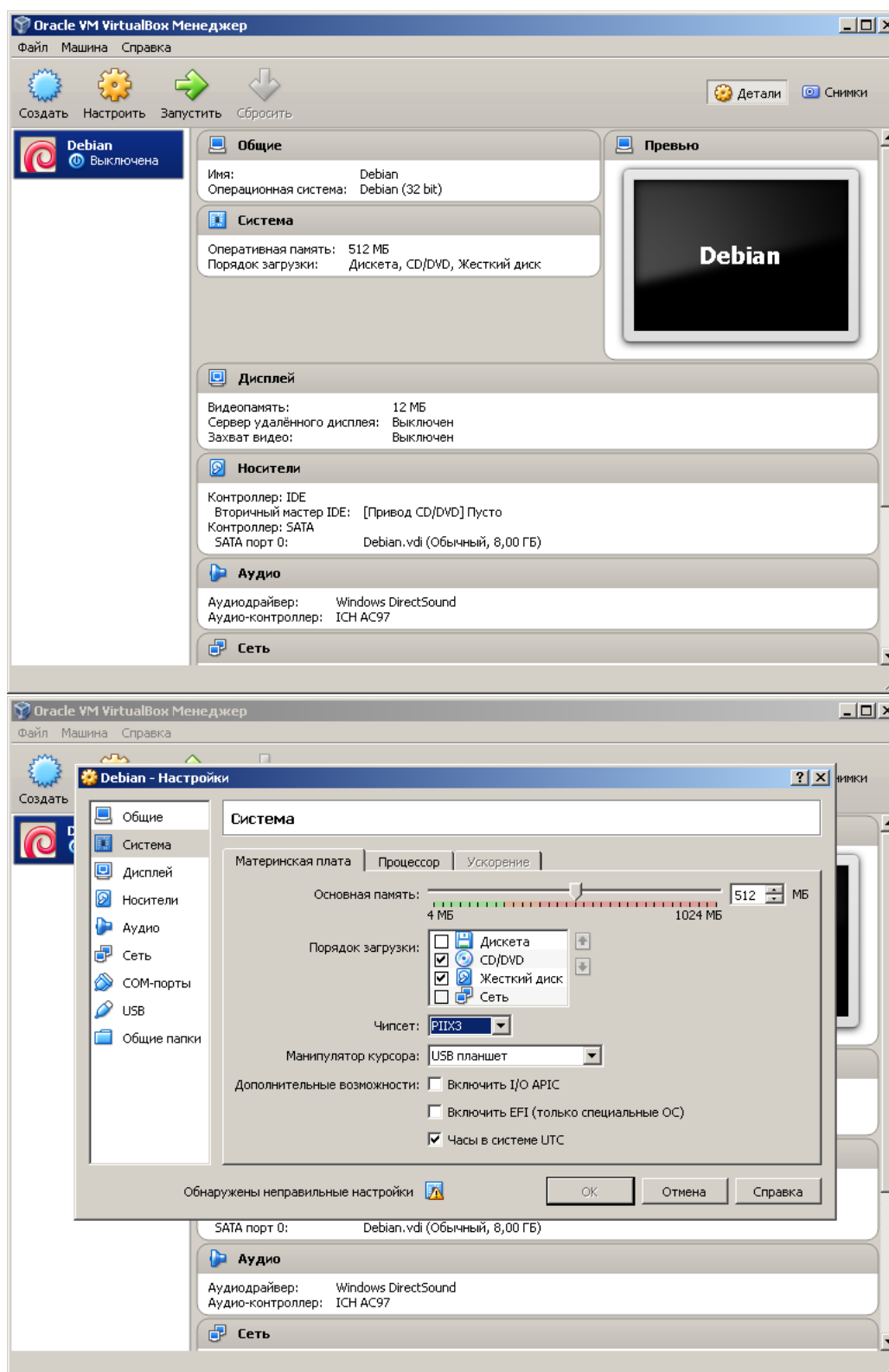
## 16 А. Пример создания виртуальной машины в Oracle VM VirtualBox



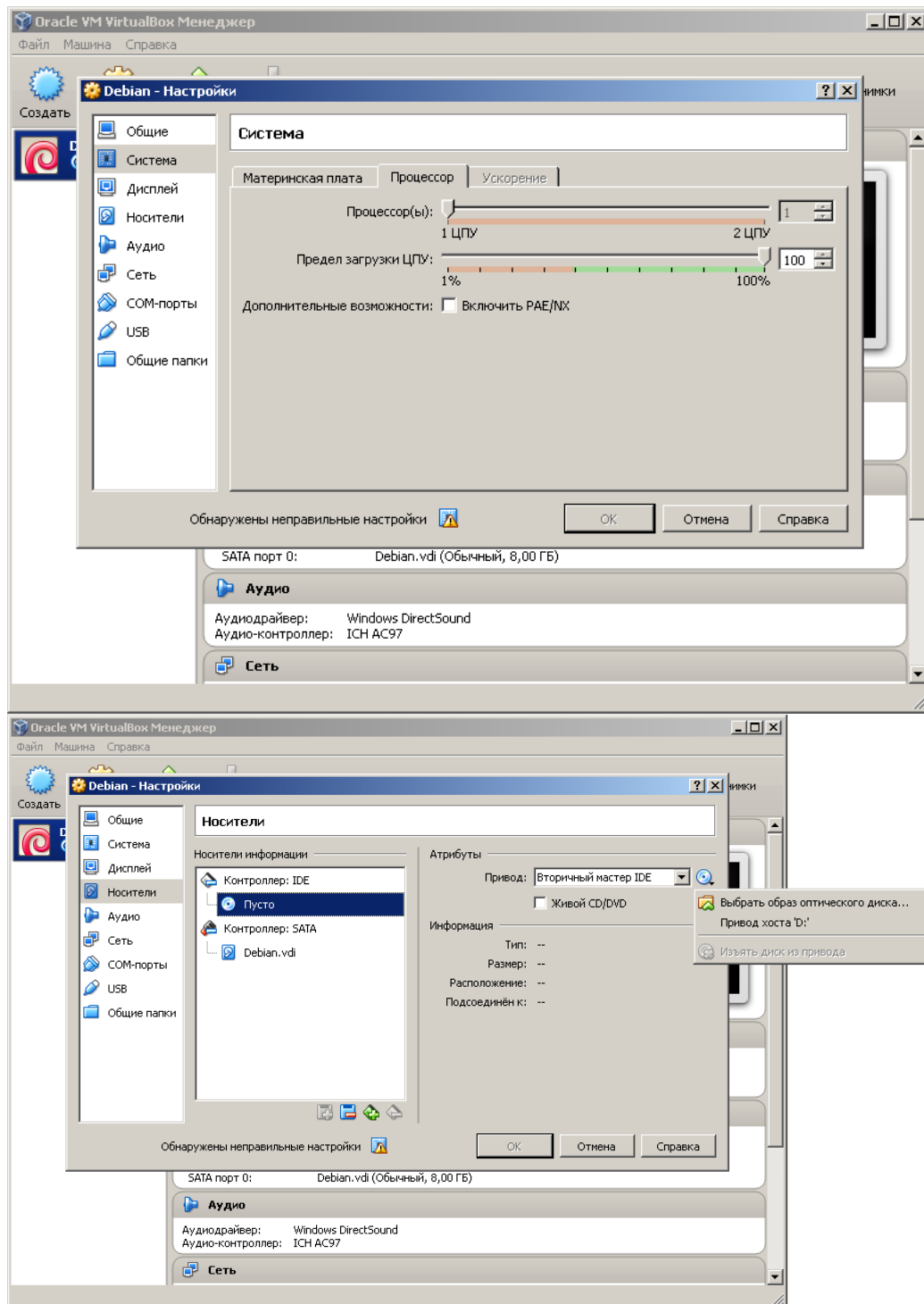


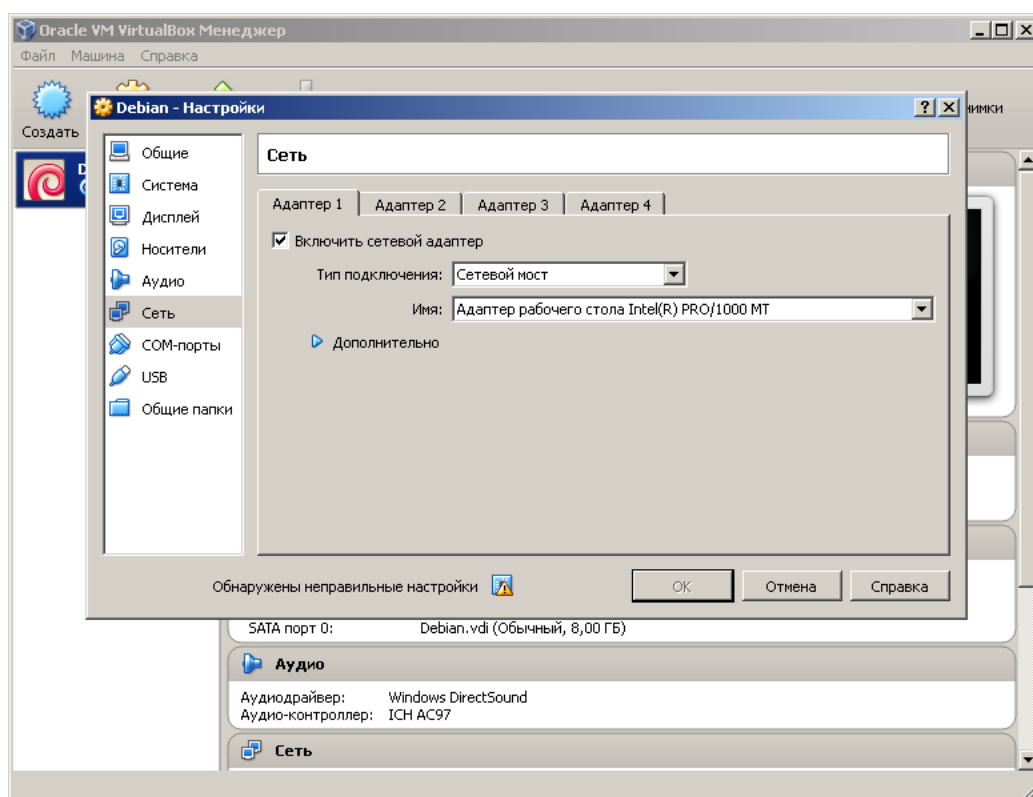
## 17 А. Пример создания виртуальной машины в Oracle VM VirtualBox





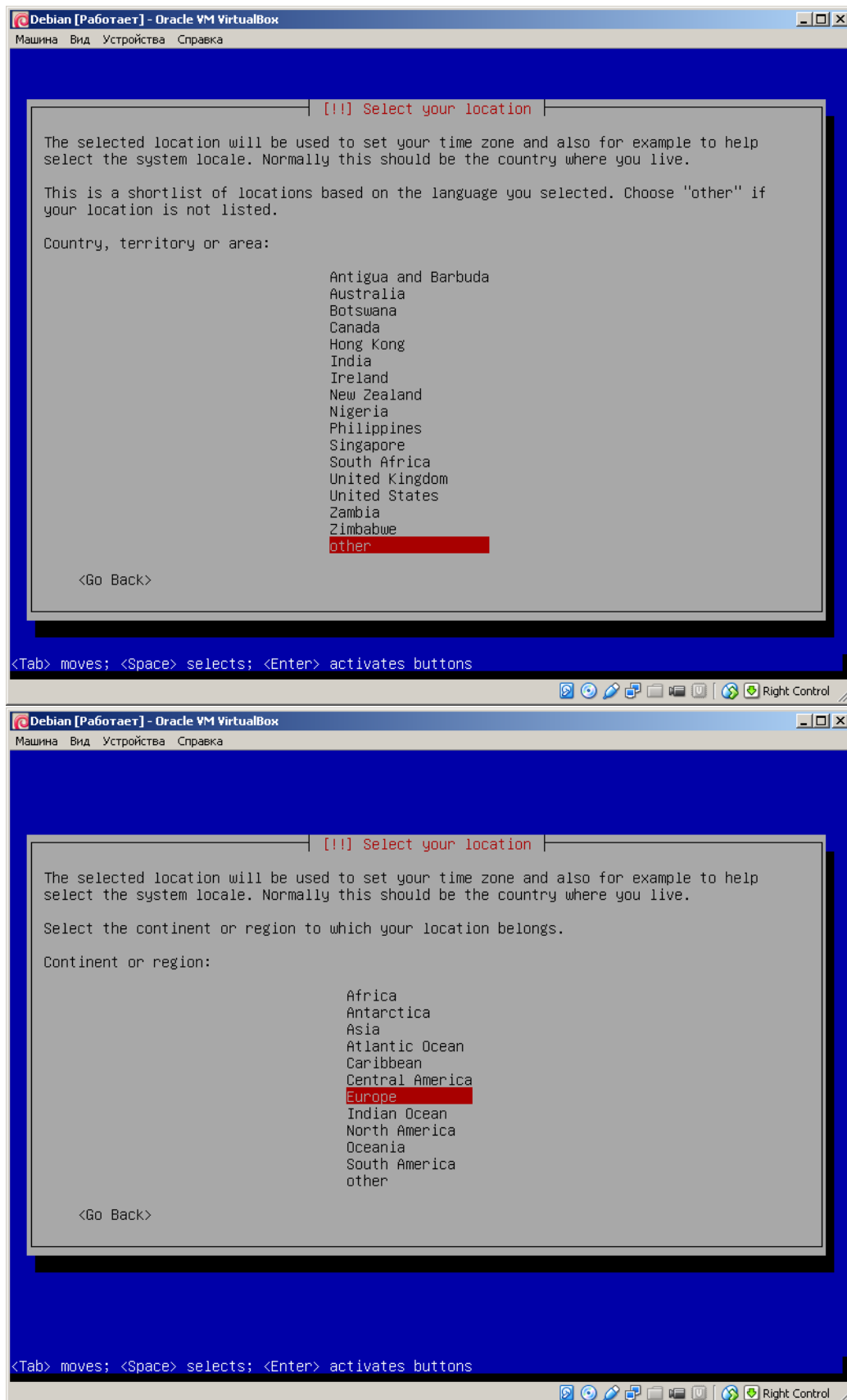
## 19 А. Пример создания виртуальной машины в Oracle VM VirtualBox

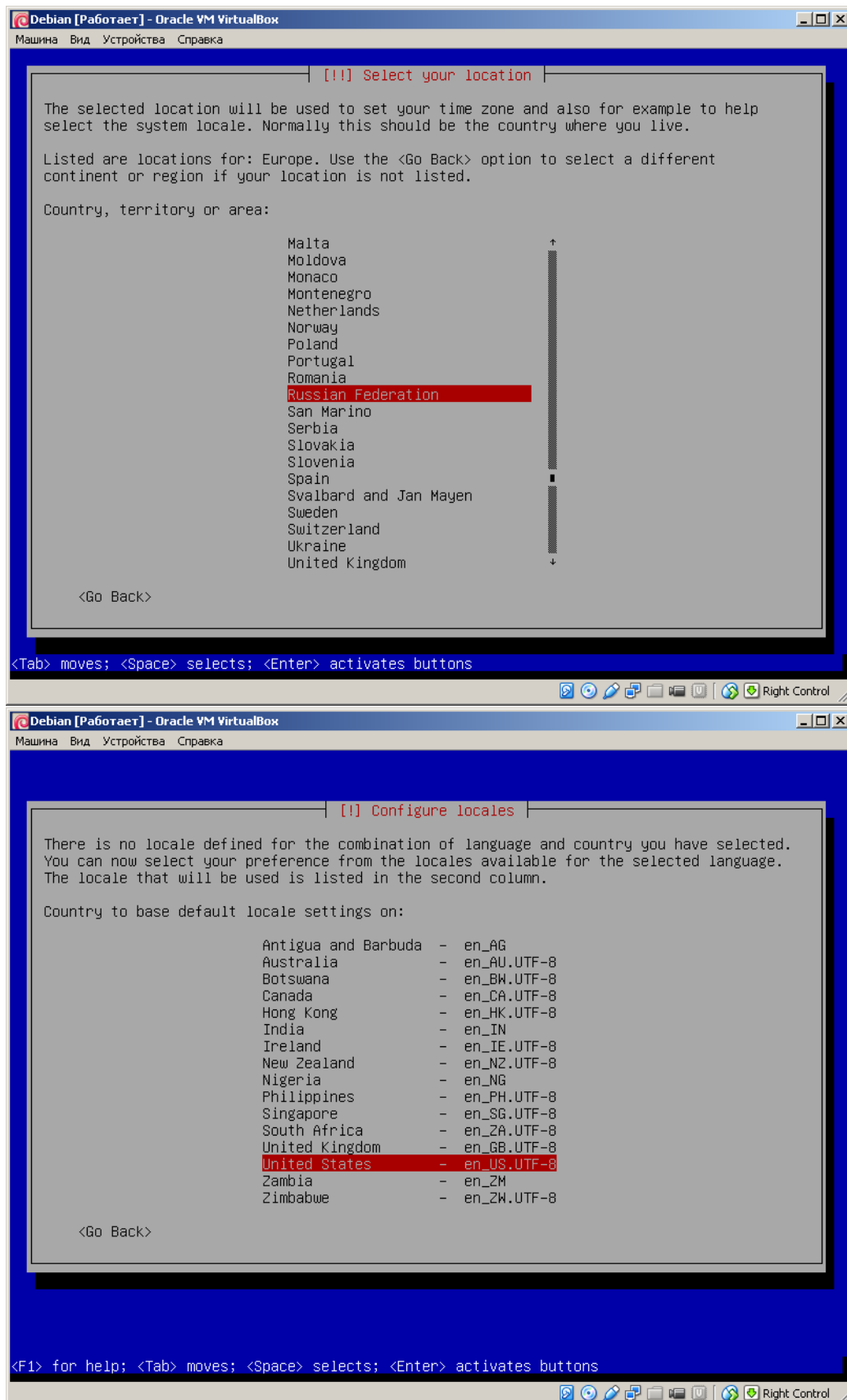


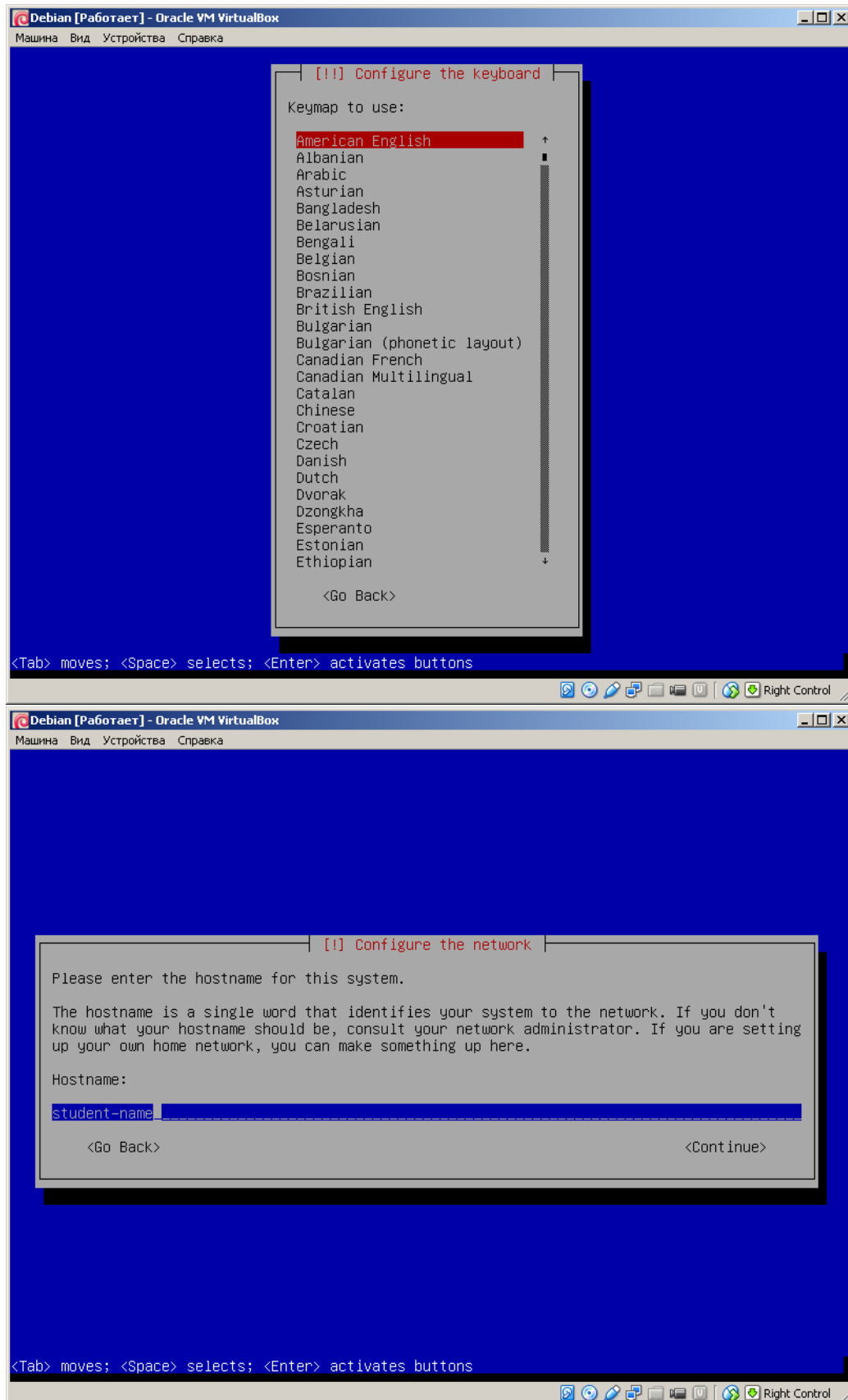


## В Пример установки Debian 7 GNU/Linux в VirtualBox

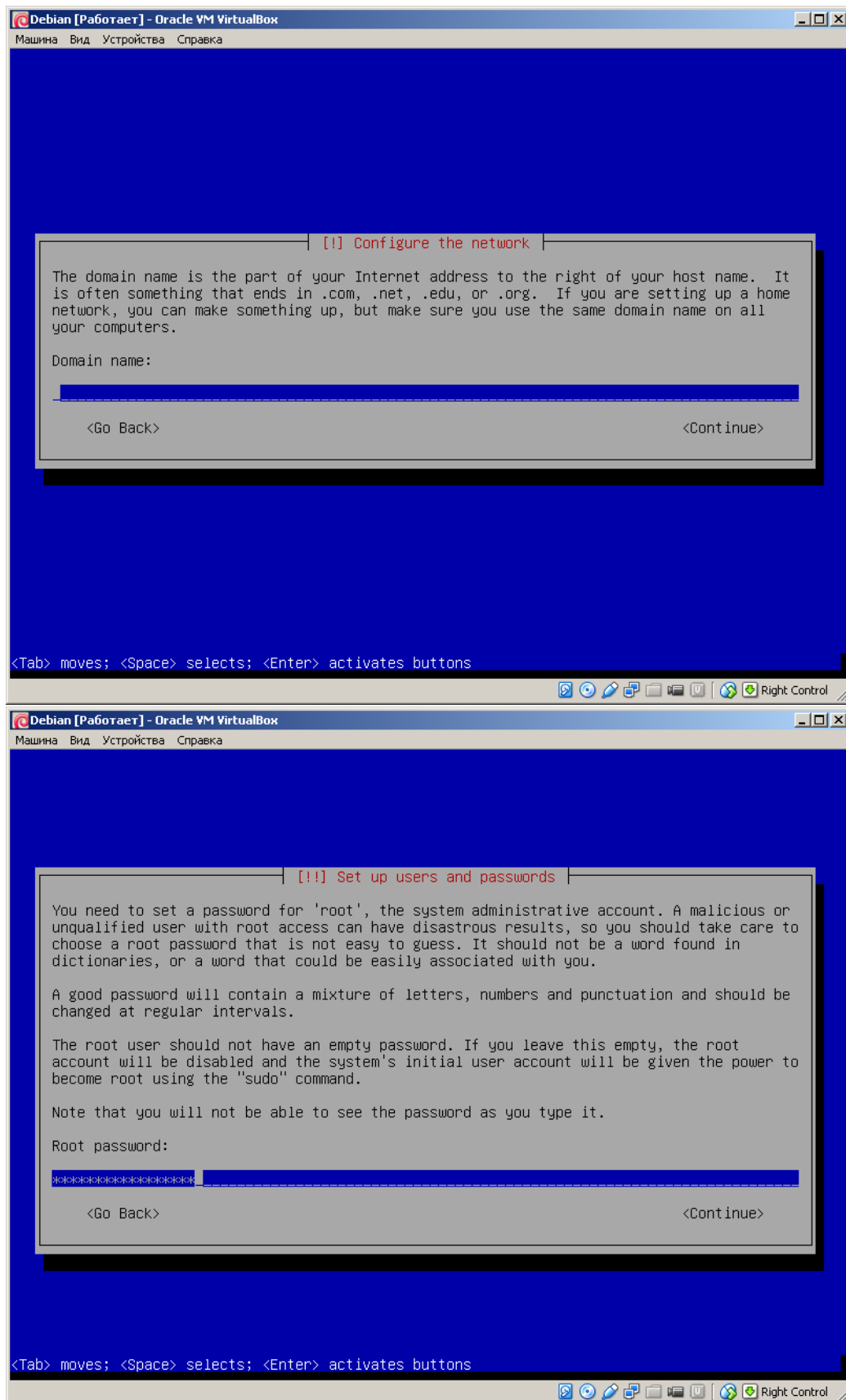


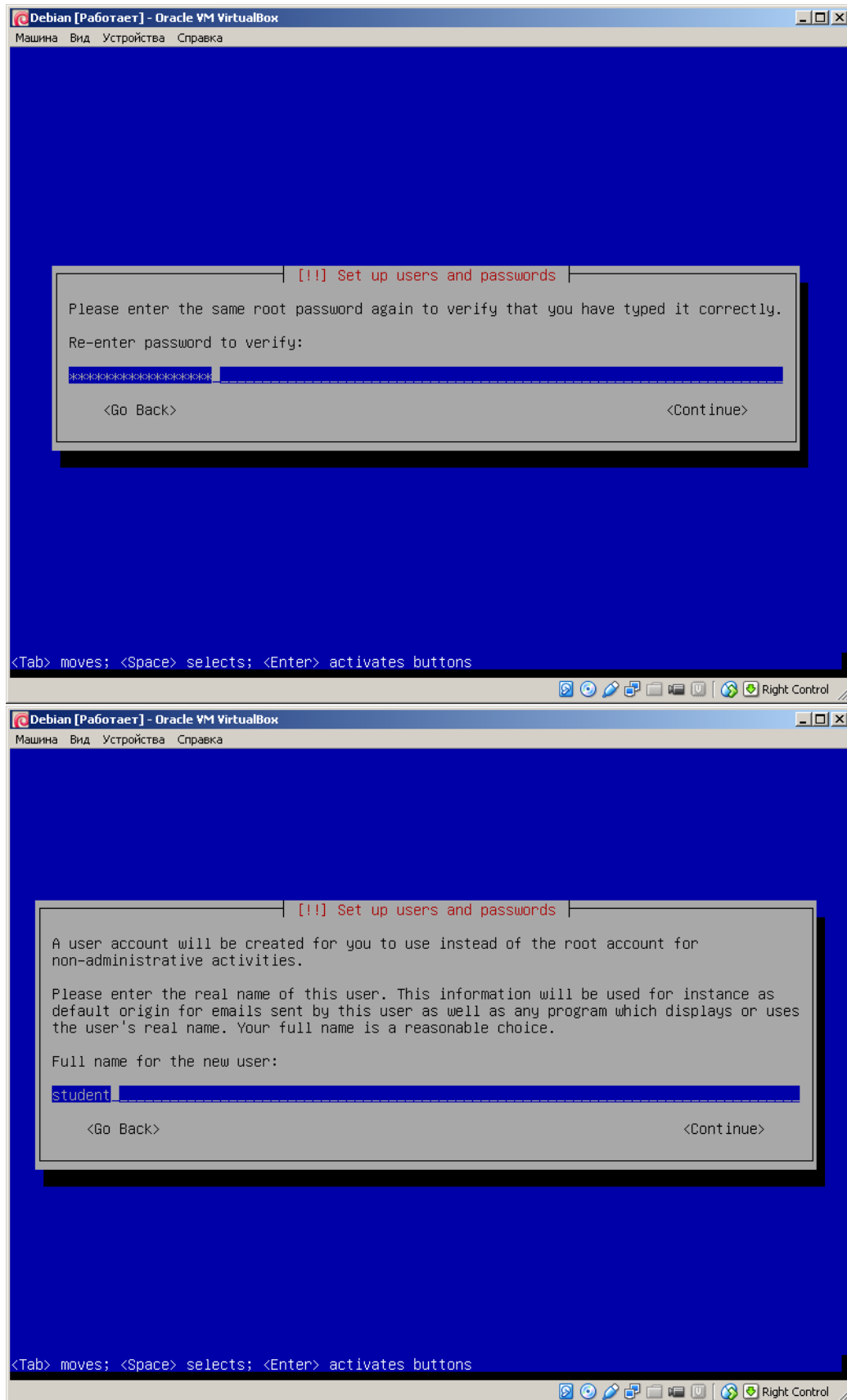


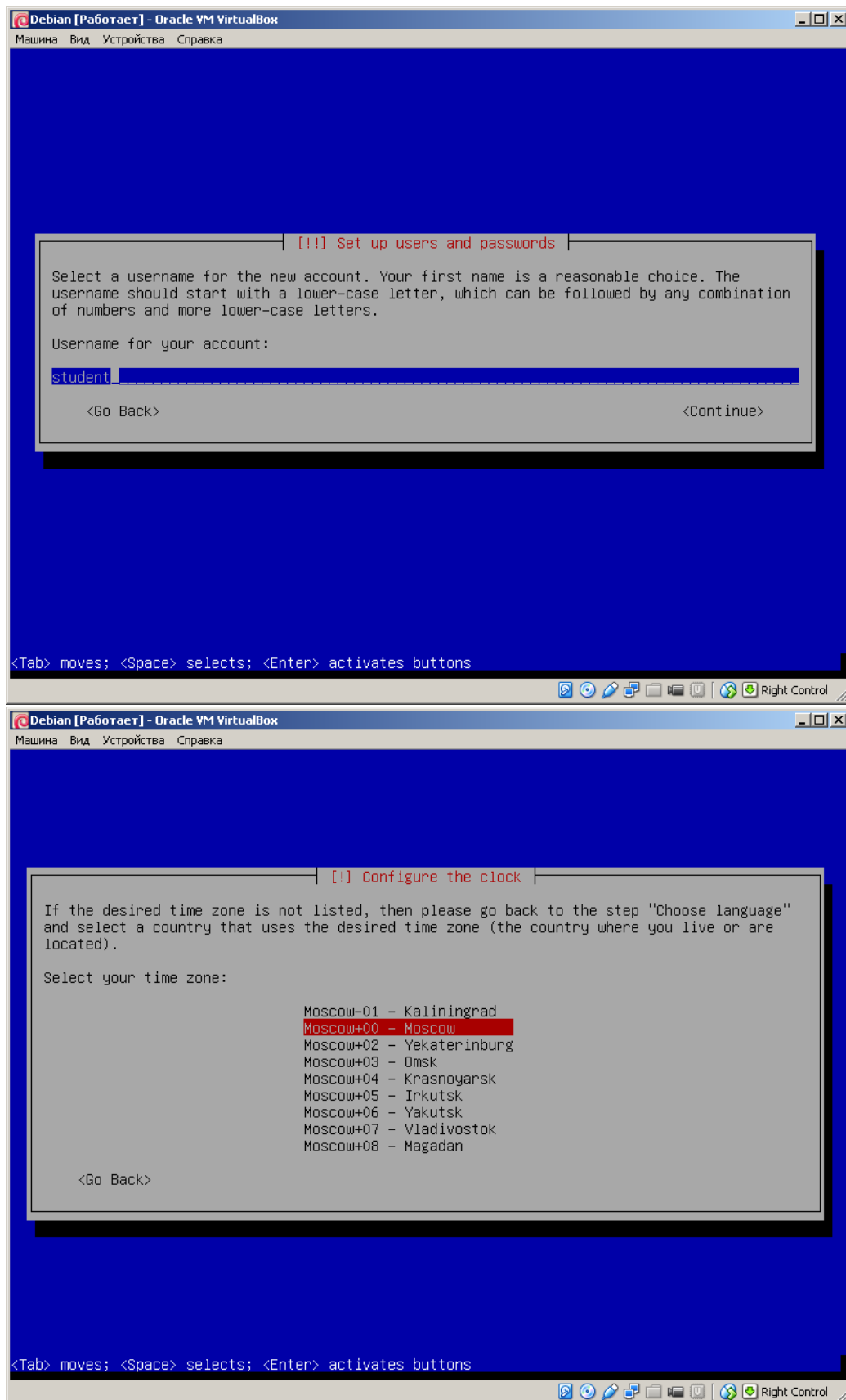


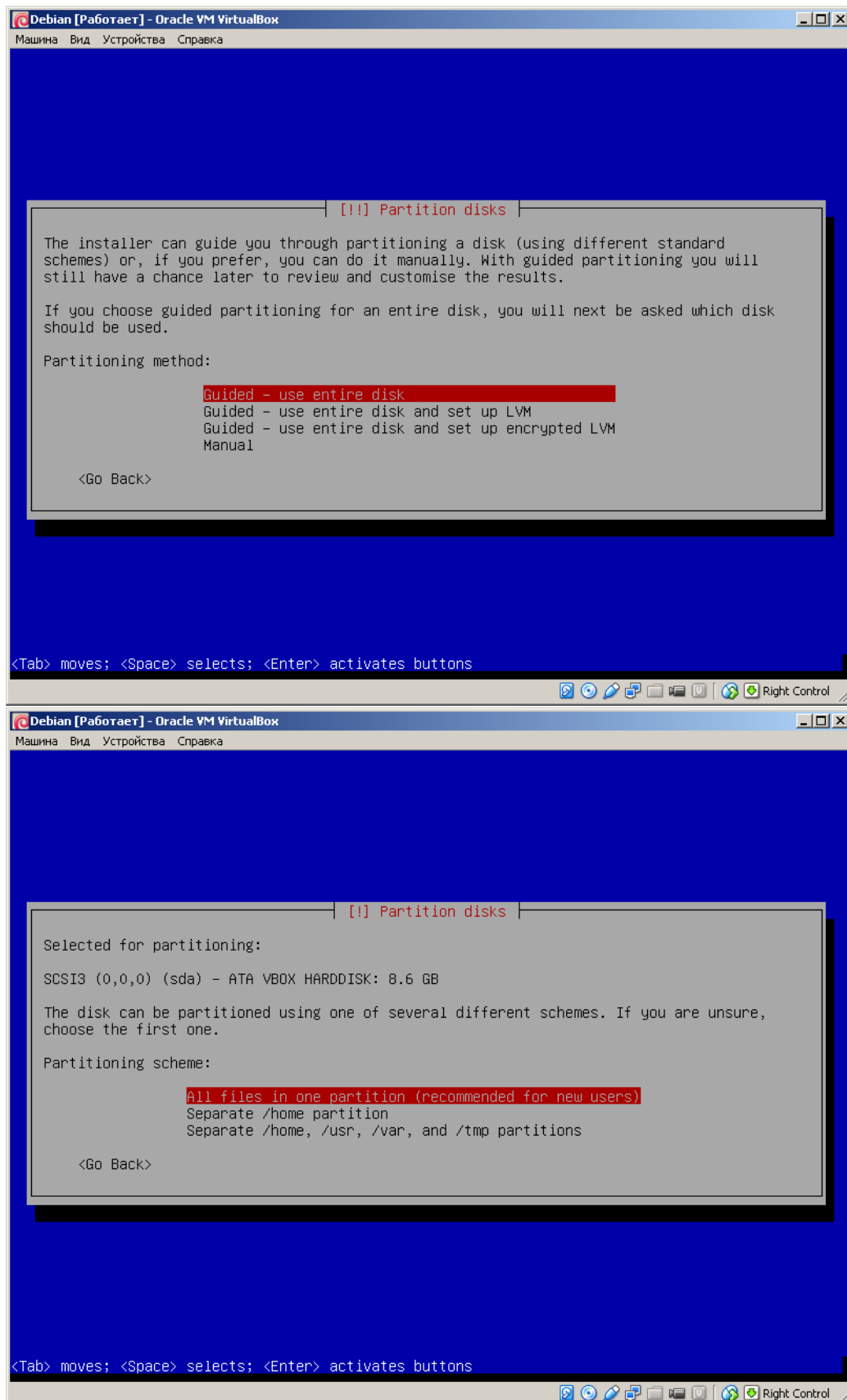


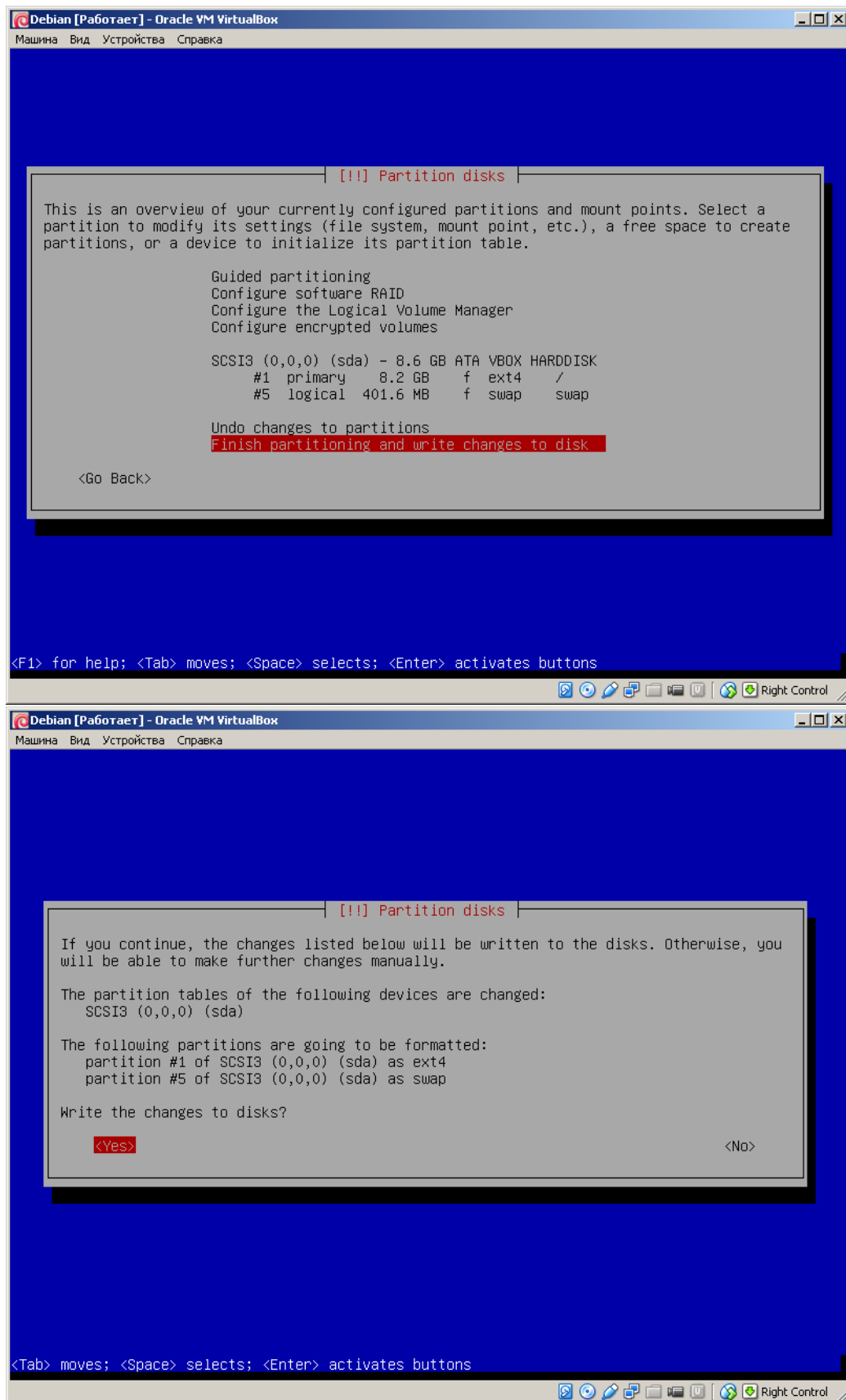


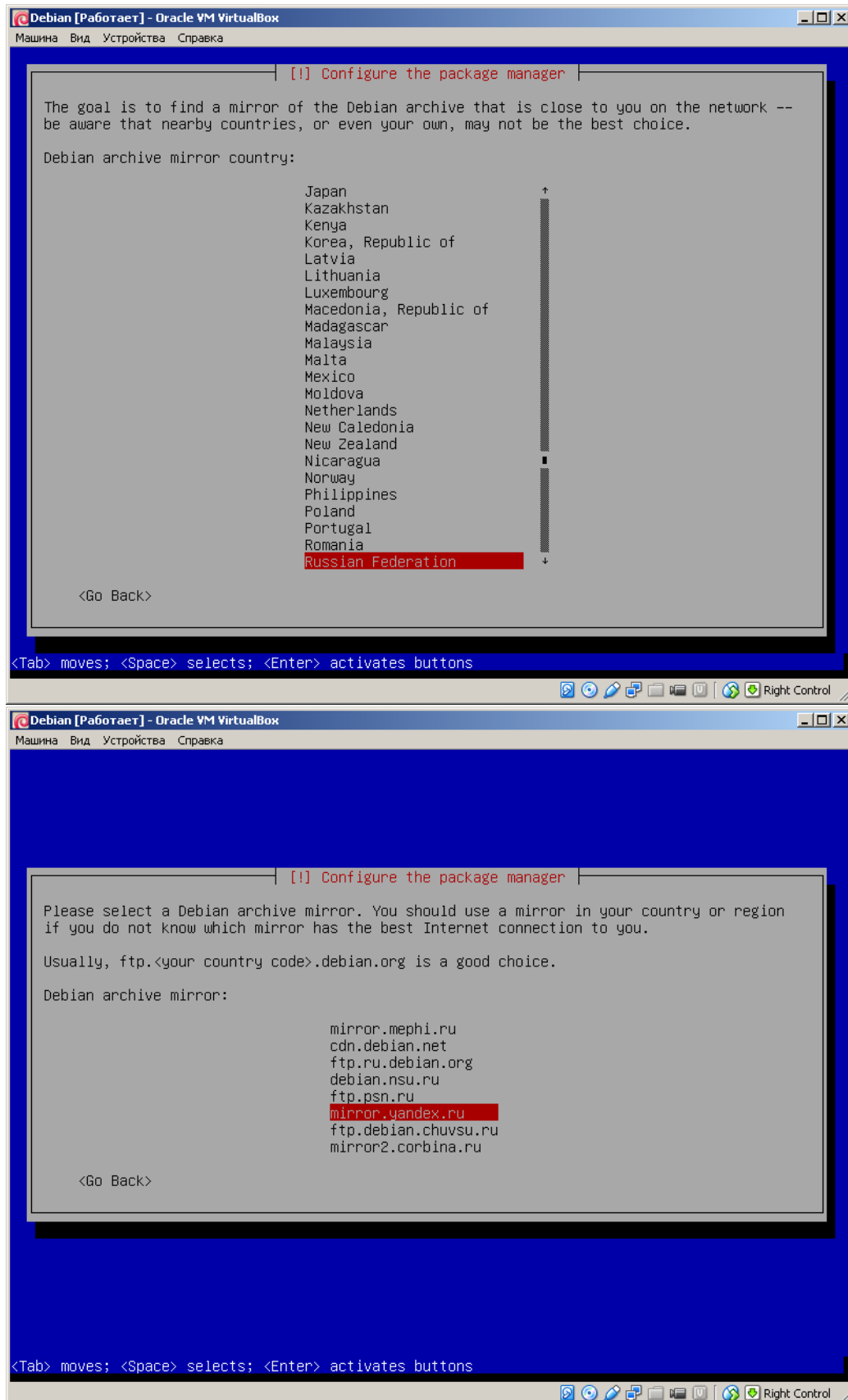


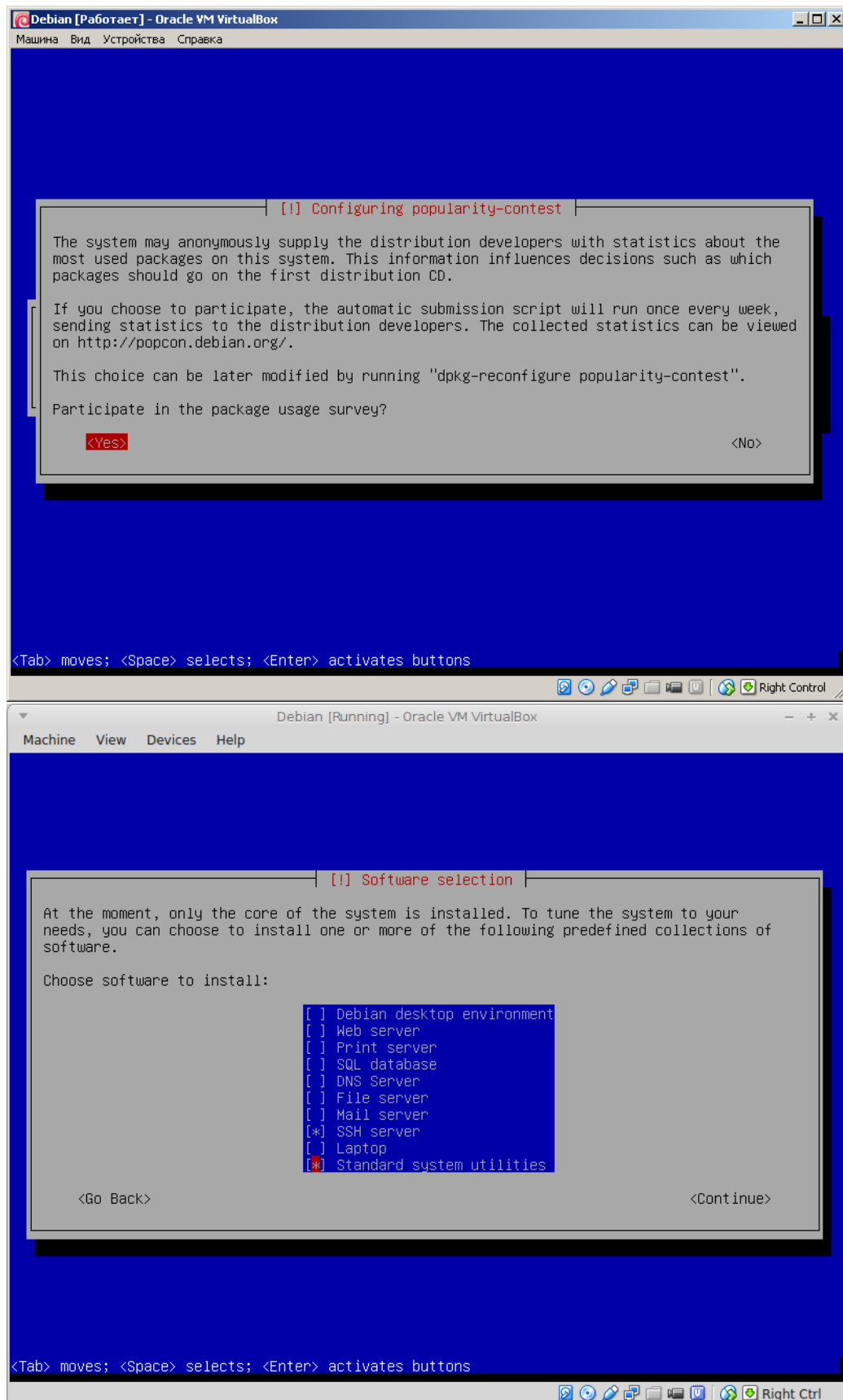


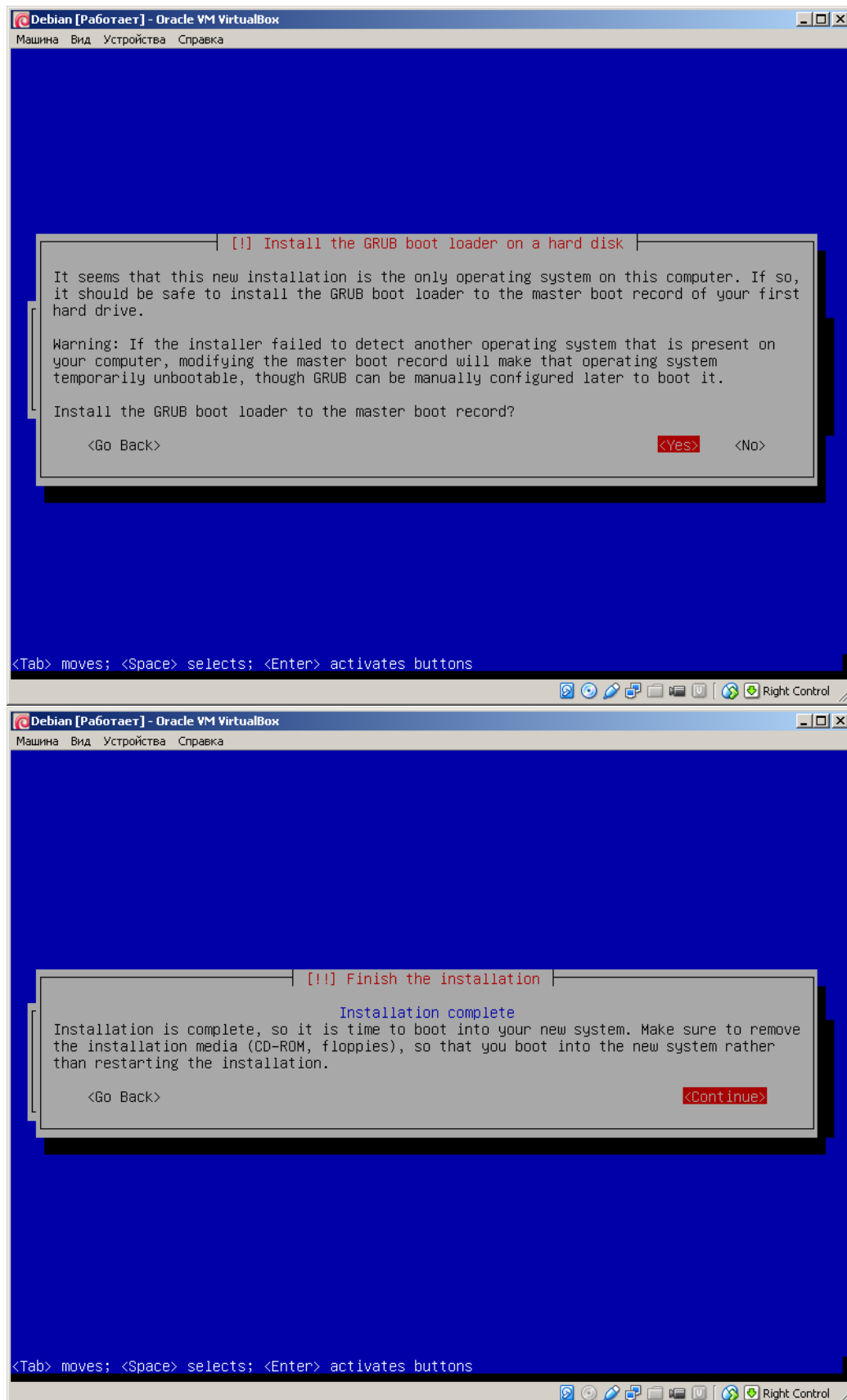














## С Пример установки ownCloud Server в Debian 7 GNU/Linux

После успешного входа в систему, в первую очередь необходимо получить права суперпользователя:

```
student@debian:~$ su -  
Password: toor
```

Добавление репозитория и ключа для ownCloud Server:

```
root@debian:~# echo "deb http://download.opensuse.org/  
    repositories/isv:/ownCloud:/community/Debian_7.0/ /" >>  
    /etc/apt/sources.list.d/owncloud.list  
root@debian:~# wget http://download.opensuse.org/  
    repositories/isv:ownCloud:community/Debian_7.0/Release.  
    key  
...  
2015-02-10 16:38:24 (51.0 MB/s) - 'Release.key' saved  
    [1003/1003]  
  
root@debian:~# apt-key add - < Release.key  
OK
```

После добавления репозитория необходимо обновить список доступного в репозиториях ПО и запустить установку ownCloud Server (во время установки MySQL, установщик запросит пароль суперпользователя MySQL, он не обязательно должен совпадать с паролем пользователя root):

```
root@debian:~# apt-get update  
root@debian:~# apt-get install owncloud  
...  
New password for the MySQL "root" user: toor-mysql  
Repeat password for the MySQL "root" user: toor-mysql
```

После того, как установщик скачал и установил все необходимые пакеты, можно проверить корректность установки (рис. 7), зайдя по адресу <http://192.168.0.102/owncloud/>, где 192.168.0.102 — IP-адрес сервера (виртуальной машины).

Приложение предлагает использовать базу данных SQLite по умолчанию, мы же будем использовать MySQL:

```
root@debian:~# mysql -u root -h localhost -p  
Enter password: toor-mysql
```

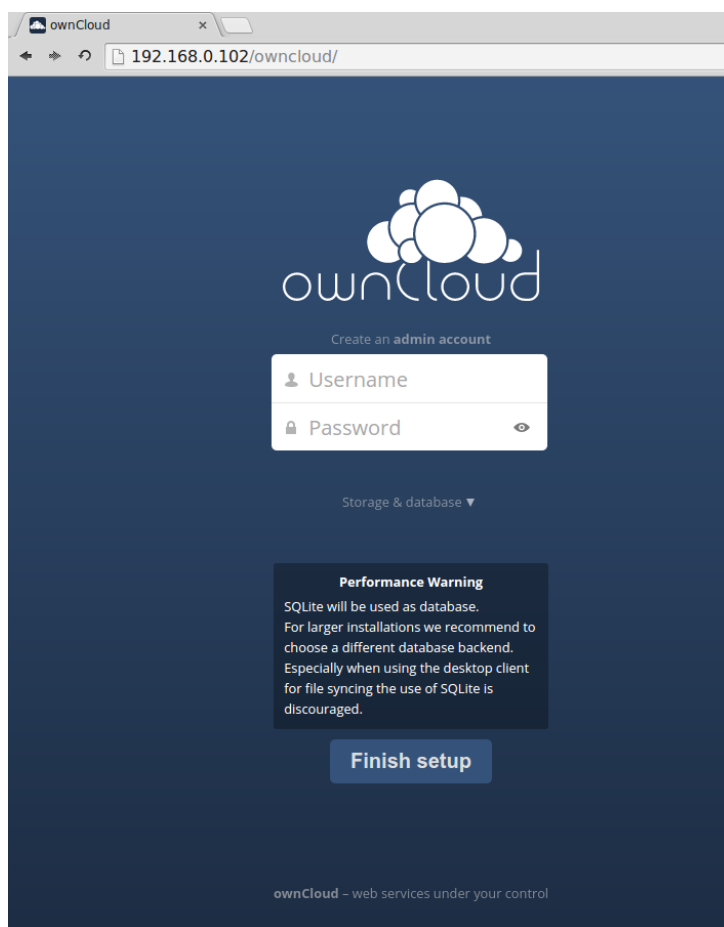



Рис. 7: Первый запуск приложения

```
mysql> CREATE DATABASE owncloud_DB;  
mysql> CREATE USER "owncloud-web"@"localhost" IDENTIFIED BY  
"owncloud-passwd";  
mysql> GRANT ALL PRIVILEGES ON owncloud_DB.* TO "owncloud-  
web"@"localhost";  
mysql> FLUSH PRIVILEGES;  
mysql> quit
```

После создания базы данных, необходимо обновить страницу приложения, настроить параметры базы данных и установить данные аккаунта администратора ownCloud (рис. 8).

После нажатия клавиши «Finish Setup» база данных и пользовательские настройки успешно подключаются к ownCloud (рис. 9).



Create an admin account

student

student-kvt-

Storage & database ▼

Data folder

/var/www/owncloud/data

Configure the database

SQLite **MySQL/MariaDB** PostgreSQL

owncloud-web

owncloud-passwd

owncloud-DB

localhost

**Finish setup**

ownCloud – web services under your control

Рис. 8: Параметры БД и аккаунта администратора

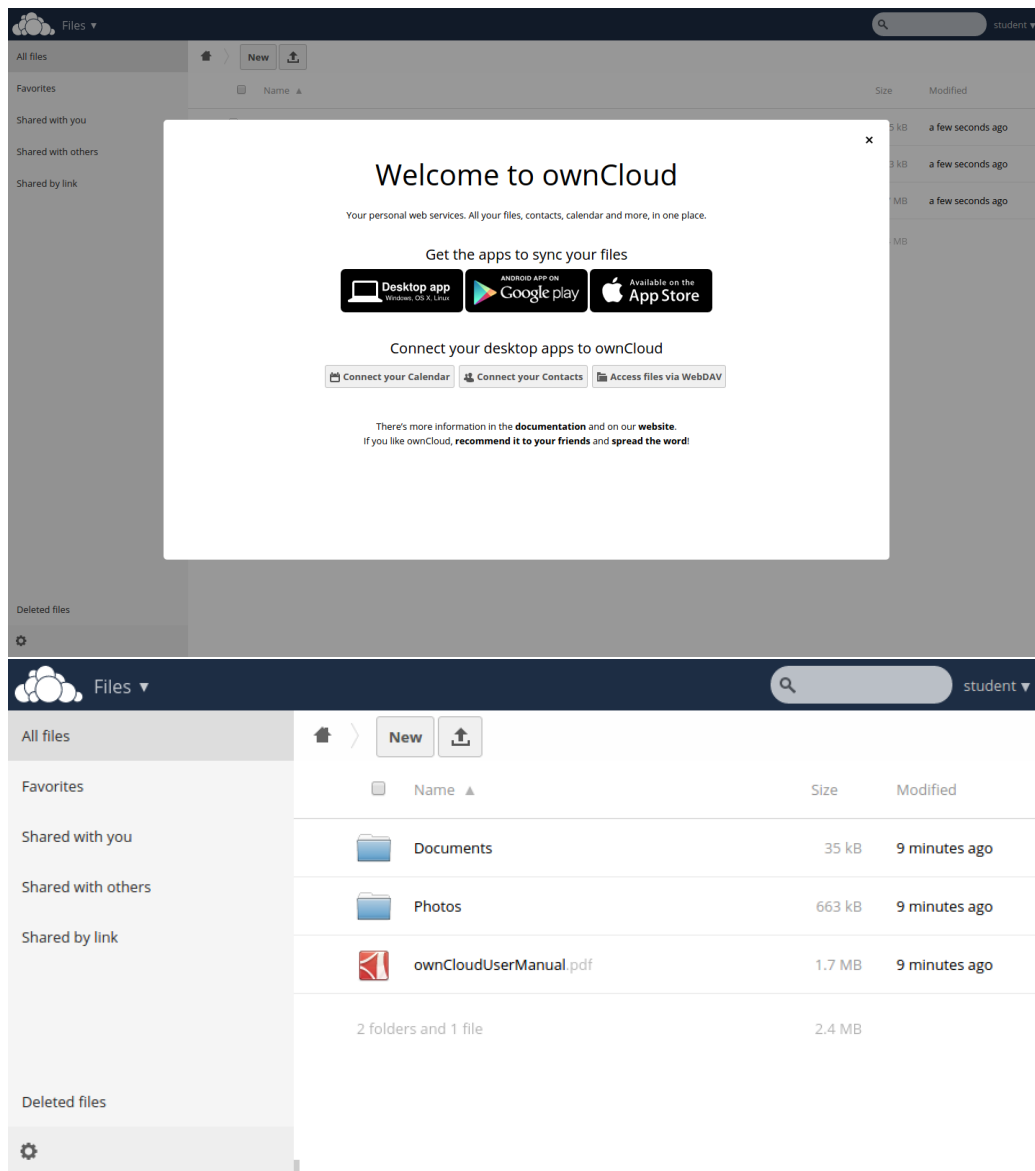
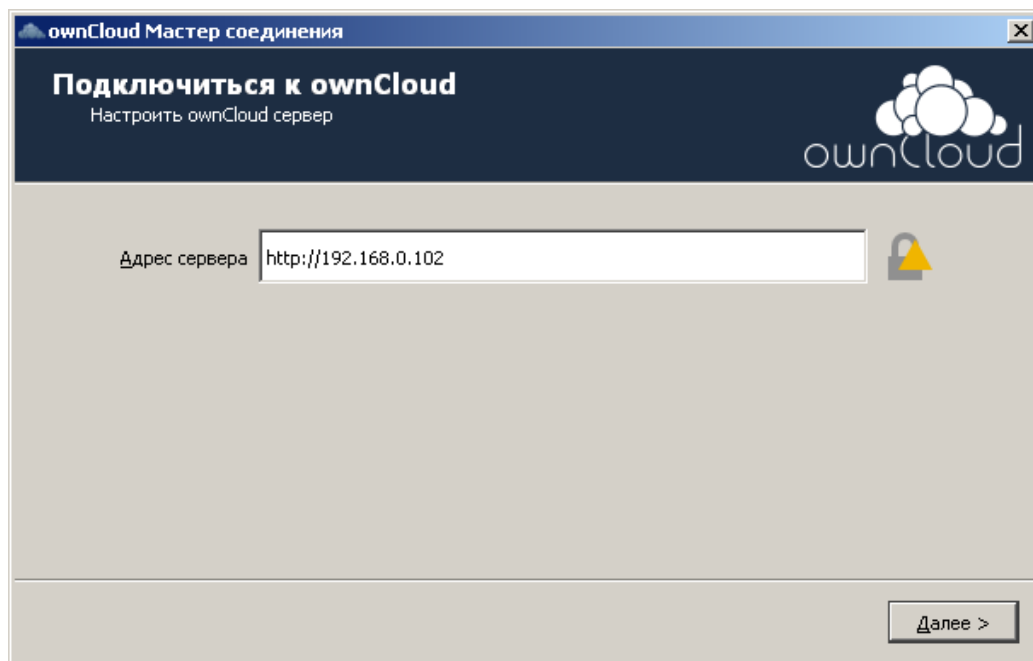


Рис. 9: Первый вход в ownCloud и интерфейс приложения

## D Настройка подключения ownCloud Client к серверу



The image displays two sequential screenshots of the "ownCloud Мастер соединения" (ownCloud Master Connection Wizard) window.

**Top Screenshot:** The window title is "ownCloud Мастер соединения". The main heading is "Подключиться к ownCloud" (Connect to ownCloud), with the subtitle "Ввести учётные данные" (Enter credentials). The ownCloud logo is in the top right. Below the heading, there are two input fields: "Имя пользователя" (Username) containing "student" and "Пароль" (Password) with masked characters. At the bottom right, there are two buttons: "< Назад" (Back) and "Далее >" (Next).

**Bottom Screenshot:** The window title is "ownCloud Мастер соединения". The main heading is "Подключиться к ownCloud" (Connect to ownCloud), with the subtitle "Настроить локальные папки" (Configure local folders). The ownCloud logo is in the top right. Below the heading, there are two sections. The first section, labeled "Сервер" (Server) with a cloud icon, has two radio button options: "Синхронизировать всё с сервера" (Sync everything from the server) which is selected, and "Выберите то, что хотите синхронизировать" (Select what you want to sync). The second section, labeled "Локальная папка" (Local folder) with a folder icon, has a text box containing the path "C:\Documents and Settings\Amet Umerov\Рабочий стол\owncloud\". At the bottom, there are three buttons: "Пропустить настройку папок" (Skip folder configuration), "< Назад" (Back), and "Соединение..." (Connect).

