This document contains modified instructions for “Setting up the VMS Software Community License Package on VirtualBox” as originally supplied by VSI and modified by Scott Snadow, 15-Apr-2024. In a nutshell, the supplied instructions are more appropriate for the *prior* Community License arrangement that involved installing VMS from an ISO image, not the *new* configuration that supplies a VMDK image. My comments are preceded with [Scott Snadow] and are also highlighted; ~~strikethrough text~~ is used as needed on VSI’s original instructions. *My own needs are currently limited to using a Windows-based host, so* ***I have not reviewed the section discussing hosting on Linux***.

Setting up the VMS Software Community License Package on VirtualBox

1. Fill out the license request form. If you provide a correct email address, you should get an email back with a download link.
2. Download and unzip the archive. You will see 2 files: community.vmdk and community-flat.vmdk. Copy both to your VM host but use community.vmdk for attaching to your VM.
3. Install VirtualBox. Follow their User Manual.
   [Scott Snadow] Supplemental information: VSI did not mention the VirtualBox Extension Pack which is also downloadable from the page that includes the download links for VirtualBox itself. **You do not need the Extension Pack** and I recommend that you do not download it: Its licensing terms are more restrictive than the terms for VirtualBox, and since you don’t need it for VMS, why bother? You may be asking yourself why I am discussing it at all, since VSI is silent about it. It’s a long story not worth going into here. Just take my word for it.
4. Depending on your host architecture, follow the instructions below to get started with OpenVMS.

Linux Shell

1. Copy the vmdk files to your host system.
2. Edit the following shell script and save it on your host system:

```
#!/bin/bash
#
# This script creates an OpenVMS VM and mounts a vmdk
VM_NAME=Test #VM name
VMDK_NAME=community.vmdk #path to the VMDK
#DISK_NAME=DATA #add an extra disk
$PORT_NUMBER=1 #extra disk adds as DKA100
CONTROLLER=SATA
VBoxManage createvm --ostype=Other_64 --name=$VM_NAME --register
VBoxManage storagectl $VM_NAME --name=$CONTROLLER --add=SATA --bootable=on \
--portcount=4 --controller=IntelAhci --hostiocache=on
#uncomment the following line to create a new disk
#VBoxManage createmedium disk --filename $DISK_NAME --size 8000 --format VDI --variant Fixed
###
VBoxManage modifyvm $VM_NAME --ostype=Other_64
VBoxManage modifyvm $VM_NAME --cpus 2
VBoxManage modifyvm $VM_NAME --pae on
VBoxManage modifyvm $VM_NAME --memory 8000 # add more memory
VBoxManage modifyvm $VM_NAME --firmware efi64
VBoxManage modifyvm $VM_NAME --chipset ich9
VBoxManage modifyvm $VM_NAME --boot1 disk
VBoxManage modifyvm $VM_NAME --ioapic on
VBoxManage modifyvm $VM_NAME --uart1 0x3F8 4 --uartmode1=tcpserver 2026 #telnet 127.0.0.1 2026
VBoxManage modifyvm $VM_NAME --nic1 nat
VBoxManage modifyvm $VM_NAME --nictype1 82540EM
VBoxManage modifyvm $VM_NAME --cableconnected1 on
VBoxManage modifyvm $VM_NAME --audio=null
VBoxManage modifyvm $VM_NAME --audio=none
VBoxManage storageattach $VM_NAME --storagectl $CONTROLLER --port 0 \
--type hdd --medium $VMDK_NAME
# uncomment the following line to add your newly created disk
#VBoxManage storageattach $VM_NAME --storagectl $CONTROLLER --port $PORT_NUMBER \
--type hdd --medium $DISK_NAME
echo "VM setup complete. Run with VBoxManage startvm $VM_NAME --type=headless"
```

3. Fix permissions on the shell script if necessary:

```
$ chmod +x createvm.sh
```

4. Run the script:

```
$ ./createvm.sh
```

5. Run the VM:

```
$ VBoxManage startvm $VM_NAME --type=headless
```

6. Connect to the console:

```
$ telnet 127.0.0.1 2026
```

7. Change Telnet settings: after you connect over Telnet, press <Ctrl+]> to invoke the `telnet>` prompt. At the prompt, type two commands: “`mode character`”, invoke the prompt again, and “`unset echo`”. This will enable you to enter passwords and make the terminal output look cleaner.
8. At the `BOOT` prompt, enter `BOOT DKA0`. If for some reason the device is not found, enter “`DEV`” to view the list of devices, and if there is no bootable device there, check the previous steps.
9. The boot process should start. After you see the accounting information, press <Enter>, and you will see the welcome message and the Username prompt:

   ```
   Welcome to OpenVMS (TM) x86_64 Operating System, Version V9.2-2
   ```

10. The Username is **SYSTEM** and the password is **alphabits000000** (not case-sensitive).

Windows GUI

1. Launch the VirtualBox Manager.
2. Set up the VM as described in this guide.
   [Scott Snadow] "This guide" is a link to a page at VSI that is more relevant for installation from an ISO image than a VMDK image; the link starts at section 2.3 of their web page. I have copied/pasted the page at the end of this document and modified it as needed; when you have finished doing the steps in that document, return to this document and continue with the next step (3.)
3. Connect to the serial port console.
4. At the BOOT prompt, enter BOOT DKA0. If for some reason the device is not found, enter "DEV" to view the list of devices, and if there is no bootable device there, check the previous steps.
5. The boot process should start. After you see the accounting information, press <Enter>, and you will see the welcome message and the Username prompt:

   ```
   Welcome to OpenVMS (TM) x86_64 Operating System, Version V9.2-2
   ```

6. The Username is **SYSTEM** and the password is **alphabits000000** (not case sensitive).

Note on KVM

To use the vmdk image on KVM, you may need to first convert it to the raw format with the following command:

```
$ qemu-img convert -O raw community-flat.vmdk community-flat.im
```

#2.3. Creating a VirtualBox Virtual Machine

Note

The following instructions have been written for VirtualBox 7.0.

To create a virtual machine in Oracle VM VirtualBox, follow these steps:

1. Run Oracle VM VirtualBox and select **Machine** > **New** from the main menu. The **Create Virtual Machine** wizard opens.
2. On the **Virtual machine Name and Operating System** page of the wizard, do the following:
   a. Set the **Name** and **Folder** for your VM.
   [Scott Snadow] The folder should be where you placed the VMDK files that you created when you unzipped the community zip file.
   b. ~~In the **ISO Image** field, specify the path to your OpenVMS ISO file.~~
   [Scott Snadow] Leave this box empty.
   c. In the **Type** dropdown, select **Other**.
   d. In the **Version** dropdown, select **Other/Unknown (64-bit)**.
   e. Click **Next**.
3. On the **Hardware** page of the wizard, set the **Base Memory** for your virtual machine (for the values recommended by VSI, refer to Section 1.2.1). Click **Next**.
   [Scott Snadow] You will need at least 8,000 MB of memory. On this page you can also set the number of processors, if desired. **Be sure to check the box for enabling EFI**, although if you don't do it now, you'll do it when you get to Section 2.3.1.
4. ~~On the **Virtual Hard** disk (for the values recommended by VSI, refer to Section 1.2.1) set the size of your virtual hard (sic). Make sure the preallocate full size option is enabled. Click **Next**.~~
   [Scott Snadow] Do not create a virtual hard disk. Instead, choose the option for using an existing virtual hard disk; this is where you will specify the community.vmdk file. You may have noticed that this file is very small; that's OK. It's actually a simple text file, but if you look inside of it, it references the other vmdk file (community-flat.vmdk) that is much larger in size. This is why *both* files are needed, and they both must be in the same directory. If the community.vmdk file does not appear in the dropdown list, you must add it:
   a. Click on the folder at the right of the dropdown list

b. This opens a Hard Disk Selector window
c. Click Add
d. Browse to the folder containing the community.vmdk file
e. Select the community.vmdk file, and click Open
f. Select the community.vmdk file, and click Choose

Now you should be able to pick the file in the dropdown list. Click Next.

5. Click **Finish**.

You have created your VirtualBox virtual machine. Now, you must properly configure it before you can ~~install~~ [Scott Snadow] boot VSI OpenVMS x86-64 V9.2-2.

#2.3.1. Completing Your VirtualBox Virtual Machine Configuration

To prepare your virtual machine for VSI OpenVMS x86-64 ~~installation~~, follow these steps:

1. Once you have created your VM, right-click it and select **Settings** from the menu.
2. In the **Settings** window, do the following:
   a. Go to **System** > **Motherboard** and specify the following settings:
      i. From the **Chipset** dropdown, select **ICH9**. If you select any other chipset, OpenVMS will not ~~install~~ [Scott Snadow] boot.
      ii. Make sure the **Enable I/O APIC** and **Enable EFI** options are checked.

      Note

      Your virtual machine *must* boot from UEFI, not BIOS.

   b. Go to **Storage**.

   In the Storage Devices area of the Settings window, you will see the default IDE controller and ~~two devices~~ [Scott Snadow] one device: the hard disk that you ~~created~~ [Scott Snadow] linked to in the previous topic ~~and the optical drive containing the VSI OpenVMS V9.2-2 ISO file~~. IDE disk controllers are currently not supported, so you *must* change the controller type to either AHCI (SATA) or LsiLogic. To do so, perform the following steps:

   i. Click the IDE controller to select it.
   ii. In the Attributes area on the right side of the Settings window, select **AHCI** or **LsiLogic** from the **Type** dropdown list.
   iii. To avoid confusion in the future, enter an appropriate name in the **Name** field.
   iv. Click the hard disk to display its attributes and make sure that it is now marked appropriately as SATA or LsiLogic.
   ~~v. Click the optical drive to display its attributes and make sure that it is now marked appropriately as SATA or LsiLogic.~~

c. Go to **Network**.
   i. Click the **Attached to** dropdown and select **Host-only Adapter**.
   ii. Click **Advanced**.
   iii. From the **Adapter type** dropdown, select **Intel PRO/1000 MT Server**.

d. Go to **Serial Ports**. Your VM must have an active serial port, so that you can connect to it later using a terminal emulator.
   i. Check **Enable Serial Port**.
   ii. Set **Port Mode** to **TCP**.
   iii. Uncheck **Connect to existing pipe/socket**.
   iv. In the **Path/Address** field, type the port number that you want to use. In this example, we will use port 2023.

e. Click **OK** to apply the changes and close the **Settings** window.

Your virtual machine is now ready to ~~have~~ [Scott Snadow] boot VSI OpenVMS x86-64 ~~installed on it. See Chapter 3~~. [Scott Snadow] Power it on and return to the previous document.