NetBSD 7.0.1 Installation in VirtualBox (32- and 64-bit)

System Description

So-called stable distribution; no attempt to provide most recent leading-edge versions of programs.

This procedure does not use a GUI desktop. Sudo users group is wheel, group enabled for sudo using visudo as root.

Latest Hercules version available in standard package repository: 3.11.

Results from `uname` on a 64-bit NedBSD:

```
uname -a:  NetBSD netbsd764.<domain>.<tld> 7.0.1 NetBSD 7.0.1
               (GENERIC.201605221355Z) amd64

uname -p   x86_64

uname -m   amd64
```

Results from `uname` on a 32-bit NedBSD:

```
uname -a:  NetBSD netbsd732.<domain>.<tld> 7.0.1 NetBSD 7.0.1
               (GENERIC.201605221355Z) i386

uname -p   i386

uname -m   i386
```

Package directories URL:

```
64-bit: http://ftp.netbsd.org/pub/pkgsrc/packages/NetBSD/amd64/7.0.1/
32-bit: http://ftp.netbsd.org/pub/pkgsrc/packages/NetBSD/i386/7.0.1/
```

Hercules-required packages that are included in the base distribution.

bzip2	1.0.6, includes libraries and headers. Required only for the legacy build process
gcc	4.8.5
grep	2.5.1, GNU
m4	BSD, no version available. This is replaced when GNU Autoconf is installed
make	BSD, no version available
awk	BSD, no version available, same functionality as gawk
sed	BSD, no version available
zlib	1.2.11, part of base NetBSD. Not installable nor removable.

Packages that must be installed.

cmake	3.9.2

> git   2.14.3
> ninja-build   1.8.2, optional substitute for make
> rexx-regina   3.0.1, which is too old. We shall build from source.

Packages that must be installed to use the GNU Autotools legacy build process.

> automake   2.69
> autoconf   1.15.1
> perl   5.26.0

VERIFY THIS: NetBSD installs perl and rexx in /usr/local, and this has two impacts.

Special Considerations for NetBSD 7.0.1

No VirtualBox Guest Additions

Because there are no VirtualBox Guest Additions for NetBSD, running a graphical desktop will be relatively inefficient and will require configuration of x11 and the desktop for the chosen screen size. Because a graphical desktop is not required to build or run Hercules, no effort is made to install any graphics support, neither x11 nor a graphical desktop.

Pre-Installation Activities

Host System Preparation

Determine the following:

- VirtualBox guest machine name
- VirtualBox group name
- IP and MAC address for the host-only network interface of the new machine.
- Name of the VirtualBox host-only network to which the guest should be connected.
- Path of the VDI file created for the guest's emulated hard drive.

Configure the host system firewall to treat the guest machine as a trusted guest. This will be required for any network access from the guest machine to the host system, for example for file sharing. Use the MAC address to identify the guest machine to the file, not the IP addresses.

VirtualBox machine preparation

Make a copy of the template VirtualBox machine creation command file. Change the following to customize it for NetBSD:

- The name, group name, OS type, OS family, and VDI path name.
- The path to the installation medium, typically an ISO image.
- The NIC type of both network adapters from virtio to 82540EM. NetBSD 7 does not seem to be able to activate more than one virtio NIC, and we need two.   (This issue is not unique to NetBSD.)

- The MAC address of the second NIC. This must correspond to the MAC address that identifies this guest to the host system firewall as a trusted machine. Trusted maching status is required for `mount_smbfs` from NetBSD to the host.

Run the script to create the machine.

NetBSD Base System Installation

Boot Installation media

Boot the network installation CD

The NetBSD installer starts. Respond to the prompts as follows:

- Pick option 1 (Install NetBSD) from the Primary Bootstrap screen to start the installation. This will install with ACPI and SMP.
- Select option a: Installation messages in English
- Select keymap z: US-Dvorak.

Note that the keyboard mapping above does not work.

- Select x: Exit Install System.
- Issue the following two commands at the command prompt:
  ```
  wsconsctl -k -w encoding=us.dvorak
  exit                # returns to text-mode installer
  ```
- Select option a: Install NetBSD to hard disk.
- You see a summary of the operations to be performed for the install. Select b: Yes to continue the installation.

Disk Partitioning

Partition the disk with no swap partition and a root partition that uses the remainder of the disk. Keep in mind as you make selections that the virtual disk is empty: it has no boot code and any Master Boot Record and partition structure are there just to make the disk recognizable.

- Select a: wd0 (12G, VBOX HARDDISK) to partition the virtual disk.
- Select a: This is the correct geometry.
- Select b: Use the entire disk.
- Select a: Yes. This will install the NetBSD bootcode.
- Select a: Set sizes of NetBSD partitions. Make the following changes to the partitions:
  - Set the size of the swap partition to zero. This removes the swap partition
  - Increase the size of the root partition (/) to the maximum, which is the value shown in parentheses on the root partition line.
  - Select x: Accept partition sizes.
  - Select x: Partition sizes ok
- Enter a disk name. I consider the host name, which in my environment includes the distro name, version, and bitness, *e.g.*, netbsd764.

- Select b: Yes, when you are asked, "Shall we continue?"

System Configuration

- Select a: Use BIOS console when asked about the console for the bootblocks,
- Select x: Exit, to leave bootblocks console selection.
- Select d: Custom installation. Ensure that the following distribution sets are selected. You can install the X11 sets if you wish. I did not.
  - Kernel (GENERIC)
  - Kernel modules
  - Base
  - Configuration files
  - Compiler tools
  - Manual pages
- When the selected items are as needed, select x: Install selected sets

You will be asked whether to install sets from CD-ROM. Select a: CD-ROM. The items needed to boot a basic system are copied from the CD-ROM. At the end of that process, you are prompted to "Hit enter to continue." Do so.

A screen listing additional items to be configured is displayed. Configure each as follows:

Configure additional items

Configure the Network

Network interface wm0 will be set for dhcp, and the second interface wm1 will be configured after system installation.

- Select a: wm0 to select the first Ethernet interface.
- Accept the default of [autoconfigure]
- Accept the default of [autoselect] for media type.
- Select a: Yes when asked, "Perform autoconfiguration?"
- Enter the host name again.
- The dhcp results are presented. Select a: Yes if they are ok.
- Select a: Yes again to have the setup for wm0 stored in /etc for future boots.

Timezone

Select America, then select America/New York, then select Exit.

Root Shell

Leave root shell unchanged. Note that the root shell specified here is used as the default shell for any users you create during installation.

Root password

Set a root password.

Enable Installation of Binary Packages

The pkgin tool is the primary package installation tool for NetBSD. The pre-seeded answers are fine; select x: Install pkgin and update package summary.

The configuration of pkgin continues with a download of the pkgin database. When complete, you see "Hit enter to continue." Do so.

Enable sshd

Press Enter to enable SSH access to this NetBSD system.

Add a User

- Add the user to group wheel? a: Yes
- User shell: a: /bin/sh   (you can change this later)
- Set a password for the user account.

Select x: Finished configuring.

The "Final Configuration" screen appears. Use this screen to revisit any of the selections you have made. Select "Hit enter to continue," when you are ready.

First Reboot After Installation

NetBSD does not eject the installation image ISO as part of shutting down prior to the first boot. This means the first boot will really be a reboot of the installation image ISO. Do the following to shut the system down cleanly1, eject the installation image, and then boot from the hard drive containing the newly-installed NetBSD system.

- o Pick option 1 (Install NetBSD) from the Primary Bootstrap screen to start the installation. This will install with ACPI and SMP.
- o Select option a: Installation messages in English
- o Select keymap z: US-Dvorak.

Note that the keyboard mapping above does not work.

- o Select e: Utility menu
- o Select g: Halt the system
- o Wait for the message "The operating system has halted…"

    Do the next two steps using the menu bar of the VirtualBox machine window

- o Devices – Optical Drives – Remove disk from virtual drive
- o Machine – Reset

This will restart the NetBSD guest from the newly-installed hard drive.

Post Installation System Setup

Login using your non-privileged user id and do the following. Note that the keyboard will be using a US QWERTY layout until you issue the `wsconsctl` command.

Change the Keyboard Layout for This Boot of NetBSD

```
su
wsconsctl -k -w encoding=us.dvorak
```

The keyboard layout has been changed for this boot only.

Install Software

```
pkgin install sudo
pkgin install nano
pkgin install bash
```

Permanently Change the Keyboard Layout

Edit /etc/wscons.conf to make the Dvorak keyboard mapping permanent.

```
nano /etc/wscons.conf
```

Change commented line:

```
#encoding sv
```

Remove the # and change "sv" to "us.dvorak" so that it looks like this:

```
encoding us.dvorak
```

Configure use of sudo

Configure /etc/sudoers to allow members of the wheel group to use `sudo`. Remove the comment ("# ") from the line that starts "# %wheel".

```
visudo
<scroll down to the line starting '# %wheel'>
<position cursor over the '#'>
xx   <deletes the # and the space>
<esc>
:wq
```

For the remaining steps, you may either continue the su session started above or you may exit and prefix commands that require root privilege with `sudo`.

Disable Root Account Warning

When one uses `sudo`, a warning is displayed that you should not log in as root, but should create a non-root account and use `su`. Since using sudo achieves the same end, disable the message. Use `nano` to comment the two lines in /root/.profile that create the message. Do the same in /root/.login.

Enable the Host-Only Network Interface

Start by connecting the host-only adapter cable to the guest machine. Connect the second VirtualBox network adapter. Right-mouse on the network icon at the bottom of the VirtualBox guest window and click on "Connect Network Adapter 2." Note that this is a permanent change; it does not need to be done each time you start the NetBSD virtual machine.
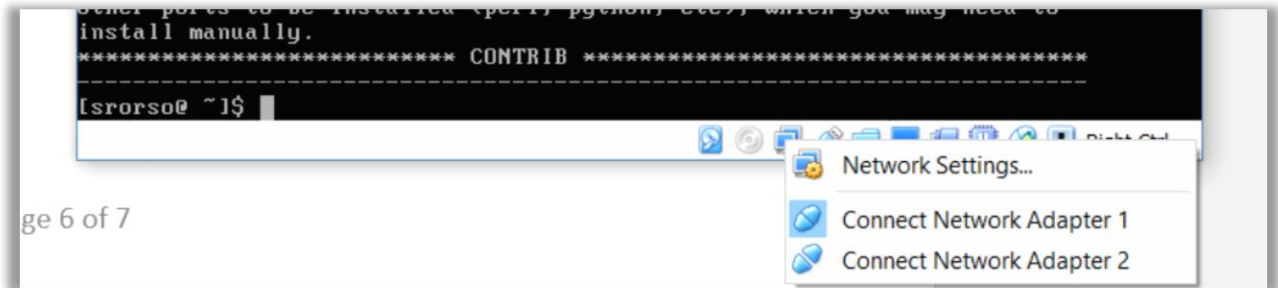
Configure the host-only adapter for just this boot of NetBSD by issuing the following command:

```
ifconfig wm1 inet <guest IP> netmask 255.255.255.0
```

Replace `<guest IP>` with the IP address that you picked in Host System Preparation, above.

Add an entry to /etc/hosts for the host system using the host address assigned by VirtualBox to its host-only adapter. Normally, this is on the same subnet as the <guest IP>, only it ends in .1.

Verify network connectivity by pinging the host system from NetBSD and pinging NetBSD from the host system. It may take a moment for the network connection to become available. The following commands may be helpful in checking that everything is set up correctly:

```
netstat -r -f inet
ifconfig wm0
ifconfig wm1
```

Once connectivity has been verified in both directions, use the following commands to configure the host-only adapter at boot:

```
cp /etc/ifconfig.wm0 /etc/ifconfig.wm1
cat >> /etc/ifconfig.wm1
inet <guest IP> netmask 255.255.255.0
<Ctrl-D>
```

Replace `<guest IP>` with the IP address that you picked in Host System Preparation, above.

Create Mount Point for SMB

FreeBSD's packaged VirtualBox Guest Additions do not support shared folders. If the host system is Windows (this author's is), then you will need to create a mount point for the github folder. Issue the following command:

```
sudo mkdir /mnt/sf_github
```

Test this by mounting the Windows github share:

```
sudo -i
mount_smbfs -I vboxhost //<hostuser>@hostname/github /mnt/sf_github
exit
```

If you include the `mount_smbfs` command on the sudo command line, you may or may not get two identical password prompts, the first for sudo (if the sudo password has timed out), the second to authenticate access to the share. Use of sudo -i avoids this confusion.

Set Up Root Certificates

Extra steps are required on NetBSD to install and enable access to root certificates to allow git and CMake to download packages using secure HTTP connections.

The following procedure is courtesy of Joachim Nilsson, posted on his blog at:

http://troglobit.com/howto/netbsd-pkgsrc/.

First, install the required software packages. OpenSSL may have been installed already as part of enabling sshd.

```
pkgin install openssl mozilla-rootcerts
```

The above does not actually install the certificates or, for that matter create an empty OpenSSL configuration file. Issue the following commands:

```
touch /etc/openssl/openssl.cnf
cd /etc/openssl/certs
mozilla-rootcerts extract
mozilla-rootcerts rehash
```

This should enable `wget`, `curl`, and applications that use those commands or functions.

Complete Post Installation System Setup

Reboot the system. When the reboot completes, sign on and verify the following:

- Correctness of the keyboard layout
- Ability to use the `sudo` command.
- Network connectivity to the Internet
- Ability to access the package repository: `sudo pkgin update`.
- Bi-directional network connectivity with the host system.

Install Packages Needed to Build Hercules on NetBSD 7.0.1

Install Required Packages

Suitable binaries exist for three of the packages that must be added to NetBSD to build Hercules. Install them. You will get a message about an unmet requirement of tk when installing git. This message may be ignored, as it is an x11 dependency and will not affect git operation in character mode. Note that the 32-bit version of NetBSD 7.0.1 does not have a pre-built CMake, and the git package has a different name.

Install the following on 64-bit systems:

```
pkgin install cmake
pkgin install git
```

Install the following on 32-bit systems:

```
pkgin install curl                 # Required to build CMake from source
pkgin install git-base
```

Install ninja, a replacement for GNU make or BSD make, if you wish.

```
pkgin install ninja-build          # Optional substitute for make
```

Build CMake from Source (32-bit systems only)

The binary package repository for NetBSD i386 systems does not include CMake. Building CMake from source is not difficult, but it does take a little time. You will need the curl package, installed as part of this procedure.

Start by visiting the cmake.org downloads page at https://cmake.org/download/ to pick the version you will build from source. A good choice would be one version lower than the most recent.   Replace the URL in the wget command below with the URL of the source you choose to build. See

   https://hercules-390.github.io/html/CMake/upgrcmake.html

for detailed information about building CMake from source for use with Hercules, including the need to install curl. Change the URL, file name, and directory name as needed to suit the version you will build. Version 3.9.2 is suggested because that is the version included in the amd64 repository.

```
sudo pkgin install curl wget
wget https://cmake.org/files/v3.9/cmake-3.9.2.tar.gz
tar xzf cmake-*
cd cmake-*
./bootstrap --system-curl
make
sudo make install
```

Install Regina-Rexx From Sources

The Regina-REXX installation materials include build targets to create RPM files (and Debian DEB files too). The created RPMs are then installed using yum. (CentOS documentation suggests that installing from source is a bad thing and that making an RPM out of the source is much better; this does exactly that.)

Download the current Regina-REXX package from https://regina-rexx.sourceforge.io/. Then use the following commands to extract, configure, create rpm files, and then install Regina Rexx. Do not attempt a parallel build using make -j. Errors result.

```
tar xzf Regina-REXX-3.9.1.tar.gz
cd Regina-REXX-3.9.1
./configure --prefix=/usr/local
make
make install
```

Install Packages Needed to Use the Legacy Build Procedure

Material that follows requires validation.

NetBSD installs perl and rexx in /usr/local, and this has two impacts:

1. bldlvlck fails because it expects perl to be installed in /usr/bin
2. configure fails to find the rexx and bzip2 headers and library because /usr/local is not included in Hercules builds on other than AIX.

The application flex, identified as required by bldlvlck, does not seem to be necessary and need not be installed.

To use the legacy procedure that uses GNU Autotools, the following additional packages must be installed.

```
pkgin install autoconf automake
```

If you do not install the compression package BZip2, Herules will be built without support for BZip2-compressed emulated tape and disk volumes. No warning of this omission will be issued during the build.

```
pkgin install bzip2
```