MINIX on Linux/VMware

COEN 177 – Operating Systems

Silvia Figueira – Fall 2007

Tutorial – Getting Started

We will be using MINIX 3 (http://www.minix3.org/) as a model to illustrate design principles and the realities of implementing an operating system. Each student will have in his/her Linux account an image of the MINIX 3 operating system, which will be used in the class projects. The MINIX 3 system will run on a virtual machine created with VMware (http://www.vmware.com/).

In our projects, we will be changing the system to make it execute tasks. We will be basically working on the existing source code by changing and/or adding files. After each change, we will rebuild the system and boot from the new (hopefully improved) version.

The first time you run MINIX 3, you have to execute the following commands from a Linux terminal:

# bash
# setup vmware
# minix-get-image
# vmware

After that, every time you login, you should simply run the following commands to get back to your MINIX 3 system:

# bash
# setup vmware
# vmware

When you run command vmware, it will open a window, from which you will be able to select and start the MINIX operating system, which will appear as a terminal on your Linux screen. From the user perspective, MINIX behaves pretty much like UNIX, i.e., all the basic commands, library functions, and system calls are the same. The system itself is smaller and simpler.

Once you start running MINIX, the cursor will be locked to that window. To unlock, type CTL-ALT. To bring it in again, just click on the window.

To login into the system at first (and to change anything in the system), you need to login as root (login: root). To shutdown MINIX, type: sync and then shutdown. To get off from VMware, type: off. The editor is called elvis, and it behaves pretty much like vi.
In case something goes really wrong

If you ever need to get a new, clean copy of the MINIX image, you can run these commands again:

```bash
# bash
# setup vmware
# minix-get-image
```

In this case, the 'minix-get-image' script will detect if you already have a MINIX image and warn you. It will then ask for confirmation before deleting your existing image and copying down the master.

**Note that you will lose everything**, i.e., all the changes made to the system up to that point. Therefore this should only be done if there is no other way to revert the damage created.

**Source Code**

The source code of the system is located in `/usr/src`. There you will find the core of the system in the following subdirectories: kernel, drivers, and servers. The kernel directory contains all the files related to the kernel. The drivers directory contains one directory per driver. The servers directory is further divided in two directories: pm and fs. Directory pm contains files related to process management, and directory fs contains files related to the file system.

Directory `/usr/src` contains other subdirectories with the source code for the non-core files that form the system, such as libraries and commands.

Each directory contains makefiles to help you compile changed files and rebuild the system image. If you add files to a directory, you will need to change the corresponding makefile(s) accordingly.

If you change the kernel, you will need to make a new image of the system before rebooting. This is accomplished by the makefile at `/usr/src`. There will be specific instructions on how to do that in each lab.

**Getting files out of MINIX**

To backup your code (mainly the files you change and create) and/or to get files out for submission, you will need to use ftp from the MINIX to the Linux system. The VMware MINIX images have network access, but only to a private LAN connected directly to the actual physical machines on which they are located. This was done mainly for security reasons.

All VMware images will end up with a 192.168.1.0/24 address, and each physical Linux box will show up as 192.168.1.1. Therefore, on any Linux box, if you are inside of MINIX, you can simply FTP into 192.168.1.1, log in with your DC UNIX username/password, and you will have access to your normal UNIX home-directory for upload/download.
Running MINIX outside the Linux lab

Both VMware and MINIX are perfectly usable via FreeNX, so you can connect to and use MINIX remotely. FreeNX is free and available on web.