

MINIX initial experiment

Paul Ashton

July, 1996

1 Objectives

1. To introduce you to smx by getting you to use smx.
2. To acquaint you with the smx sources.

2 Getting started with smx

This lab introduces you to the Solaris MINIX operating system—a version of MINIX that runs as a SunOS process.

1. Change to the `src/tools` directory of the smx distribution and have a look at the `.minix` file it contains. If you want to run smx from any directory, then you will need to copy `.minix` to your home directory and to give full pathnames for the `hdx` and `image` options. This is not necessary if your current directory is `tools`.
2. Make sure the directory containing the smx SunOS binaries and scripts is on your search path.
3. You can now start up `minix` simply by entering `minix`. A couple of lines of boot messages come up followed by the `login:` prompt. Three warnings appear from networking software—you can ignore them because networking is disabled by default.
Login as `root` (that's right—you are a MINIX superuser). There is no password. The first thing to do after logging in is to set a password using `passwd`.
4. Logout, and see what happens. To actually close smx down properly, login again and enter the `shutdown now` command.
5. Now that you have changed the `root` password, it is (reasonably) safe to make MINIX multiuser (you should give the `bin` usercode a password as well). Edit `src/tools/.minix` again. Remove the `#` from the beginning of the `host` line, and replace the string `myminix` with a host name for your copy of smx. Using your SunOS usercode as a hostname will mean that you arrive at a name that differs from everyone else's.
6. Startup smx and login as before. MINIX is a multi-user Unix system, and has many commands in common with SunOS. List the `/usr/bin` directory to see a full list of commands available.

7. In another `xterm` on the same Sun as the one on which you are running `smx`, enter the command `mlogin hostname`, where `hostname` is the name you selected in step 5. Login as `root`. You now have two sessions logged into MINIX! Convince yourself that the two sessions are logged into the same `smx` instance, using tools like `ps`, `who` and `users`. Notice that within a single SunOS process running `smx` are many `smx` processes!
8. You can also setup networking between multiple instance of `smx`, as is explained in “Smx—the Solaris port of MINIX”.
9. Use the `df` command to discover what filesystems are currently mounted. Each `smx` filesystem is stored as a SunOS file, with the correspondence between `smx` disk device files (`/dev/hdx`) and SunOS files specified in the `.minix` file. What is the relationship between the total blocks reported by `df` in `smx` and the file size reported by `ls` in SunOS?

Your root file system should have over a megabyte free. You can always create extra `smx` file systems if need be—see “Smx—the Solaris port of MINIX” for more details. The other two file systems are mounted read-only, and are shared by the whole class.

3 Creating a new user

Now that you are a system administrator, you can add new users to your system. You can do this using the `adduser` script, or do it by hand in the following steps:

1. Create an entry for the new usercode in `/etc/passwd`. Entries have the form:

```
username:passwd:uid:gid:description:login-dir:login-shell
```

For example (`passwd` should be left empty):

```
paul::9:3:Paul Ashton:/usr/paul:/usr/bin/ash
```

2. Create the home directory specified in the `/etc/passwd` entry, and change the owner of the directory:

```
mkdir /usr/paul
chown paul /usr/paul
chgrp other /usr/paul
```

3. Copy `.profile` and `.ashrc` into the home directory and change the ownership of these files as well.
4. Log into the new usercode, then put a password on it.

Note that a new group was really required (say `users`), as `other` doesn't seem a particularly appropriate group. You can add a new entry to `/etc/group`, and change the `/etc/passwd` entry for the new user (and the group of their home directory and files) if you wish.

4 Compiling programs for smx

Solaris MINIX user programs are compiled under SunOS using the `mcc` command. They are transferred to smx using `sunread`. See “Smx—the Solaris port of MINIX” for more information on `mcc` and `sunread` (and other SunOS and smx commands specific to Solaris MINIX).

Do steps 1 and 2 in SunOS, and steps 3 and 4 under smx.

1. Make sure that your `MX_LIB` and `MX_INCL` are set to the correct values.
2. Write a hello world C program and compile it using `mcc` (which takes the standard “C compiler” command line options).
3. Read the smx executable into smx using `sunread` from within smx. `sunread` takes one command line argument (the SunOS pathname of the file to be read in), and writes the file specified to standard output.
4. Change permissions on the file to make it executable, then execute it within smx.