How to Install Uucp in One Easy Lesson

This is a step-by-step procedure for installing uucp on a system. Follow these steps in the sequence shown, in super-user mode.

(1) Create the file /etc/systemid. This file must have a UNIQUE system id in it (i.e., 'trs16' is NOT a good idea for a system id). If it does not, two systems with the same id that attempt to communicate will get confused as to who owns what, and chaos will be the result.

The name should be 6 or 7 characters followed by <ENTER>, with no uppercase characters and no characters that would be illegal in filenames (such as '/').

As an example installation, let's say that we have two software development divisions of the TSD corporation, which need to exchange files on an as-needed basis. They have an rs-232 cable running between their machines and wish to use uucp to have one machine call the other so that the machines may exchange files, mail, and so on.

We'll call the machines 'tsd1' and 'tsd2'. On tsd1, the systemid file is created with the name 'tsd1'. The same is done on tsd2, using 'tsd2' instead of 'tsd1'.

(2) Cd to /usr/lib/uucp. This directory is the 'program' directory for uucp. It contains all the files you will need to edit in order to install uucp, except for /etc/systemid.

(3) Edit the file 'L-devices'. This file has a line in it for each device that uucp will be receiving or making calls on. The format is as follows:

```
device_line  call_unit  line_speed
```

device_line is the line on which uucp will make calls on. In our example, it is decided to make the connection on channel A on each machine. Thus for device_line, each shows 'tty01'.

call_unit is the device that uucp will use to dial numbers with. If the line is a hardwire connection, then the call_unit is listed as 0. This tells XENIX that this is a hardwired connection. In the example, it is 0 for both machines since they will be communicating over the rs-232 cable.

line_speed is the baud rate at which uucp is to send or receive data. For a hardwired connection, you usually want to go as fast as the line will support, which in this case is 9600 baud. A typical hardwired device entry would be 'tty01 0 9600', and this is what is put into each 'L-devices' file in the example.

(4) Edit the file 'L-dialcodes' if you wish to have dialing abbrieviations in your calling numbers. In the example, since there is no one to call, this file is empty and unedited. It should exist, even if no abbrevations are to be used.

(5) Next, edit the file called 'USERFILE'. This file tells uucp which systems are permitted to call in, what directories they are permitted to access, and which users are permitted to use uucp locally. It consists of a list of lines which have the following format:

```
login_name,systemid  path_name  [path_name]...
```

Where a path_name is an acceptable place for that particular login name / system-id pair to do business (request files from that area, put files there, etc). In the example, the standard 'uucp' login is used, and the USERFILE on tsd1 contains the following two lines:

```
uucp,tsd2  /usr/spool/uucppublic  /usr/project_x/src
,  /
```

The first line tells uucp that any system logging in with the name uucp with the system name of 'tsd2' is only to be allowed access in '/usr/spool/uucppublic' and in '/usr/project_x/src'. Note that on the second line, there is just a ','. This is permissible, as both the login_name and the system-id are optional. Since there is no login name, this allows any local user on tsd1 to use uucp, sending and receiving files anywhere in the filesystem where they would normally be able to read or write.

Note that uucp itself needs world read/write permission to be able to transfer files, and that the path_names give permission for the indicated directory and all sub-directories.

The USERFILE on tsd2 contains more than two lines:

```
uucp,tsd1  /usr
uucp,   /usr/spool/uucppublic
root,  /
frank,  /usr/frank  /usr/spool/uucppublic
john,   /
penny,  /usr
```

First, the 'uucp,tsd1' entry tells uucp that when tsd1 calls in, it may send/receive files anywhere under /usr. Next, the 'uucp,' line tells uucp that any system that it doesn't know, i.e., one that is not specifically listed, is allowed access only in /usr/spool/uucppublic. Lastly, since there is no ', /' line in the file, only the users 'frank', 'john', and 'penny' will be allowed to use uucp locally.

(6) Edit the file called 'L.sys'. This file contains a list of the systems that uucp 'knows', i.e. can call out to or queue work for. It is configured like the previous files, with one line per entry, as follows:

```
system_name when_to_call call_dev speed phone_# login_info
```

system_name — is the same as used in the USERFILE, that is, it is the name of a system that it is okay to call or queue work for.

when_to_call — is an entry telling the system the time(s) it is okay to call this particular system (see vol 4, pg. 3-18 for more information).

call_dev — is the device that uucp is going to call out over, or, in the case of a system which cannot make calls, the device that it receives calls on (this becomes a don't-care field when the when_to_call field is 'Never').

speed — is the baud rate that should be used on call_dev.

phone_# — is only used where uucp has an Automatic Call Unit that it can talk to. For a hardwired connection, the same device entered for call_dev is used here.

login_info — is a field which consists of a list of prompts that uucp should expect and what to answer the prompts with when it gets them. A typical entry would be

Notice that the word 'password' was not spelled out fully. This is permissible, as uucp does pattern matching on the prompt string, and is standard procedure (in case a couple of characters are dropped after the <ENTER>).

A sample entry in L.sys for a hardwired setup would be:

```
my_system Any tty02 9600 tty02  login: uucp ssword: password
```

In the example, it was decided that tsd1 would be the 'originating' machine, in other words, the machine who does the calling, thus leaving tsd2 to receive the calls. Therefore, the L.sys file on tsd1 appears as:

```
tsd2 Any tty01 9600 tty01  "" uucp ssword: password
```

Notice that he used "" in lieu of a prompt string. This tells uucp to not look for anything - just go ahead and send out the response (in this case, 'uucp'), then wait for the string 'ssword:'. When it receives this, then it will send 'password'.

The L.sys file on tsd2 has a similar entry:

```
tsd1 Never tty01 9600 tty01  login: uucp password: password
```

Notice that 'Never' is listed as the time to call. This means that tsd2 will never try to call tsd1 (tsd2 is a receive only system in our example). When uucp sees 'Never', then it will not attempt to call - it will just queue the work for that system when it calls.

General Installation Notes

On a machine that is going to be calling OUT over a line, you must make sure that the line has been disabled. This may be accomplished by performing a 'disable ttyOn', where 'n' is the tty number you put into the L.sys and the L-devices files. Otherwise, the XENIX login program will fight uucp for all of the I/O over that line and neither one will get anything intelligible. For example, the super-user on tsd1 (the originating system) enters 'disable tty01'.

On a machine that is going to be RECEIVING calls, you must make sure that logins are ENABLED on the line. This may be done by typing 'enable ttyOn', where 'n' is the tty line that uucp will be calling on. Otherwise, the remote uucp will be unable to login, as the receiving system will be ignoring the line. The super-user on tsd2 types 'enable tty01' to make sure that logins are enabled.

On each machine, you must make sure that the idea of the other machine's passwords and systemids are the SAME. If they are not, this will cause grief and trouble to no end.

At this point, uucp should be installed on your machine. If no mistakes have been made, it is now configured for use.