

Downloading Front-ends from fecode-bd

This note outlines the procedure for establishing the ability to download an ACNet front-end from the Controls Department's operational server named fecode-bd. This procedure assumes use of the rfies development tools and the convention of front-ends having the same name as their development project. For example the millrf project produces the millrf front-end.

1) Log in to fecode-bd from your secured outland login (i.e., log in to outland using Kerberos or a CRYPTOCard, and then rlogin to fecode-bd):

```
rlogin -N -l vxworks_boot fecode-bd
```

2) Run the addhost program to add your front-end's network node name and user name to fecode-bd's .rhosts file. The network node name is the name, in dot notation form (e.g., millrf.fnal.gov), that was assigned when you registered your node for an IP address. The user name vxworks_boot is recommended, indeed required if you need read/write file access on fecode-bd.

```
addhost
Adding .rhosts file entry...
Node to add [default-> none]: xxx
User to add [default-> vxworks_boot]: vxworks_boot
```

where xxx is the network node name of your front-end. If your front-end does not require read/write file access you may log out of fecode-bd and go on to #4 below.

3) To provide read/write file access on fecode-bd you must create a directory with the same name as your project under the /fecode-bd/vxworks_write/fe directory. The vxworks_boot login has very limited functionality (e.g., you can't do a cd command) so creating these directories is accomplished by specifying path names relative to the login directory. To do this you must be logged-in to fecode-bd as vxworks_boot (see #1 above) and type:

```
mkdir ../vxworks_write/fe/xxx
```

where xxx is the name of your project. Any read/write directories required by your project must be created in this way. For example the sequence:

```
mkdir ../vxworks_write/fe/xxx
mkdir ../vxworks_write/fe/xxx/ksvxi
mkdir ../vxworks_write/fe/xxx/ksvxi/v152
mkdir ../vxworks_write/fe/xxx/ksvxi/v152/tbl
```

would create a read/write directory structure for a project xxx requiring a KineticSystems V152 resource manager tbl directory. When all read/write directories have been created you may log out of fecode-bd and continue with #4 below.

4) To provide read-only file access on fecode-bd you must create a directory with the same name as your project under the /fecode-bd/vxworks_boot/fe directory. To do this you must be logged into your development machine (e.g., nova) as yourself and type:

```
mkdir /fecode-bd/vxworks_boot/fe/xxx
chmod 775 /fecode-bd/vxworks_boot/fe/xxx
```

where xxx is the name of your project. Alternatively, you can use make to create this download directory from your project's sandbox:

```
setup xxx
make downl oaddi rectory
```

where xxx is the name of your project.

You are free to create any desired structure within this directory. Remember to use the chmod 775 command on each directory to establish proper group access permissions. Install all files that you will be reading or downloading into this directory structure. Note that the front-end will have read-only access to these directories.

5) Initialize the front-end's VxWorks boot parameters as necessary:

5.1) The **host name** is fecode-bd.

5.2) If using a Controls Department standard kernel the complete kernel **file name** is:

```
vxworks_boot/kernel/yyy/vxWorks,
```

where yyy is the name of your single board computer family (e.g., mv162, mv2300 or mv2400). This scheme is risky business however because a kernel referenced in this manner could be modified at any time by the Controls Department without being tested on your front-end. A much safer scheme would be to place a copy of a tested standard kernel and its symbol table (i.e., vxWorks and vxWorks.sym) in the directory created in #4 above and load from there. In this case the complete kernel **file name** would be:

```
vxworks_boot/fe/xxx/vxWorks,
```

where xxx is your project name.

5.3) The **inet on ethernet** should be 131. 225. xxx. yyy: ffff0000, where xxx and yyy represent your particular subnet and node designations. The subnet mask shown above (: ffff0000) is typical but may be different for your front-end's subnet. Contact Network Administration to determine the correct value for your particular case.

5.4) The **host inet** address for fecode-bd is 131. 225. 121. 145

5.5) The **gateway inet** address should be 131. 225. xxx. 200, where xxx is your node's particular subnet designation as specified in 5.3 above. Again, this value is typical but may be different for your front-end's subnet. Contact Network Administration to determine the correct value for your particular case.

5.6) The **user** should be identical to that specified while running addhost in #1 above.

5.7) The **ftp password** should be left blank as we are using rsh for downloading front-ends.

5.8) The **startup script** path should be

```
vxworks_boot/fe/xxx/yyy.
```

This path is the one set up in #4 above where xxx is your project name and yyy is your startup script's name.

5.9) The **boot device, processor number, inet on backplane, flags, target name** and **other** boot parameters are project specific and must be provided by the front-end designer.

6) If your front-end requires NFS access to the vxworks_write and vxworks_boot directories created in #3 and #4 above you must call the fecode-bd system administrator and request that your front-end be given NFS access privileges on fecode-bd. You can then use the following code in a startup script to mount fecode-bd:

```
#mount NFS files - 1217, 5143 is user vxworks_boot, group bdmi crop
nfsMount( "fecode-bd", "vxworks_boot/fe/xxx", "/home" )
nfsMount( "fecode-bd", "vxworks_write/fe/xxx", "/write" )
nfsAuthUnixSet( "fecode-bd", 1217, 5143, 0, 0 )
```

where xxx is the name of your project.

Your network node name must be in the DNS system for your front-end to be able to load its code from fecode-bd (step #2 above) or for NFS access (step #6 above) to work. In theory the DNS system gets updated between 8:00 and 9:00 AM of the morning after your network address is assigned by the Networking Group, but this does not always happen as expected.

End.

040130

DCV