

17 June 2001

LOCAL FLOPPY ACCESS ON A THIN CLIENT RUNNING REMOTE APPLICATION

by Robert Stanford (rob@rotapile.com)

This is a result of needing a more transparent style of floppy access on a terminal running remote applications. Once set up properly you can have local floppy access on your terminal. This works well enough so that those using gnome-desktop can use gmc or nautilus to access their local floppies and I have even had limited success with win4lin accessing the local floppy through a virtual dos volume symlinked to one of the network block device nodes. Cat and DD should work as well.

The crux of it are things called Network Block Devices, these allow a machine to export its block devices (check your kernel source for what block devices are!) to another machine so the remote machine sees them as local devices. There's alot you can do with them regarding logical volumes and raid etc but here I will just concentrate on accessing floppy drives. What we are doing is running a small server program on the terminal that is bound to the floppy drive and is listening on a port for a client program to connect. We then run a client program on the LTSP server which contacts the terminal and maps the terminals device through its own network block device.

First thing you will need to do is add support of the LTSP servers kernel for network block devices, you can add a module or make it part of the kernel. Just look in the BLOCK DEVICES section of config menuconfig or xconfig.

Or:

```
insmod nbd
```

The nbd.o module is in the block devices directory of your kernel modules
/lib/modules/<kernel-version>/block/nbd.o

Next you will need my tarball which contains the nbd-server, nbd-client and ltsd_nbd script. Copy the nbd-client to /usr/local/bin

Copy the nbd-server and ltsd_nbd script to /tftpboot/lts/ltsroot/ltsbin/

Alternatively:

The source for the nbd-server and client are available from :

<http://atrey.karlin.mff.cuni.cz/~pavel/nbd/nbd.html>

as at the writing of this document the latest version for download is at:

<http://atrey.karlin.mff.cuni.cz/~pavel/nbd/nbd.14.tar.gz>

```
./configure ; make ; make install
```

For the Workstation.

Make sure you have a kernel with floppy support.

In /tftpboot/lts/ltsroot/etc/rc.local add the following line to the end of the file.

```
/ltsbin/ltsp_nbd &
```

If you want to be real flash you could use the following script at the end of your rc.local file

```
-----  
LOCAL_FLOPPY='get_cfg LOCAL_FLOPPY N'  
if [ "${LOCAL_FLOPPY}" = "Y" ]; then  
    echo 'Starting ltsp_nbd'  
    /ltsbin/ltsp_nbd &  
fi  
-----
```

and for any Workstations that you want floppy access on add the option

```
LOCAL_FLOPPY = Y
```

These will start the network block device server in the workstation ready to accept connection on port 1025 and mapping to the floppy.

When the Workstation boots you should see

Waiting for connections.. bind, listen, accept,

If you are running an Xserver you won't have much of a chance to see this so you may want to set the UI_MODE = CHAR or SHELL in the /tftpboot/lts/ltsroot/etc/lts.conf file if you have troubles.

The reason for the ltsp_nbd script is if the nbd-server bombs out you can't just restart it immediatly, atleast not on my machines, so i added in a 30 second sleep before it restarts.

On the server

Make a .dev directory in each users home directory

Create a device node with major 43 and minors 0,1,2,3,4.....foo for all the users who need local floppy.

```
mknod /home/fred/.dev/fd0 b 43 0; chmod 666 /home/fred/.dev/fd0
```

```
mknod /home/mary/.dev/fd0 b 43 1 ; chmod 666 /home/mary/.dev/fd0
```

```
mknod /home/spot/.dev/fd0 b 43 2 ; chmod 666 /home/spot/.dev/fd0
```

Starting the nbd-client program on the LTSP server

From here you have a few options, personally I like gdm and all I did was add the following line to the /etc/X11/gdm/PreSession/Default file

```
/usr/local/bin/nbd-client $USER 1025 /home/$USER/.dev/fd0 &
```

The nbd-client must be executed as root hence my running it here. This will not work running it as a regular user! This also means that each user is bound to each terminal as you will have to make the hostnames in the /etc/hosts file the same as the user on the terminal. Anyone want to volunteer to do a decent script here!

Or if you just want to test type in

```
nbd-client <ws-ip-address> 1025 /home/fred/.dev/fd0
```

e.g.

```
gc-infolink:/home/rob# ./nbd-client 192.168.0.100 1025 /home/rob/.dev/fd0
```

```
Negotiation: ..size = 1474560
```

```
gc-infolink:/home/rob#
```

The Future

There is little documentation on nbd's and I am keen to implement an extension of Network Block Devices called Enhanced Network Block Devices, the server for these has the option to launched from inetd which would be neat as I have inetd running on my workstations at the moment for local scanner access (that to come soon).

Other possibilities of nbd's seems to be local cdrom access, and exporting almost any block device you can think of including usb thumb drives and possibly even burning cd's on a terminal. NBD's are supposedly faster than nfs, you can even set up parallel links over multiple nics and ports.

Also the scripting I have implemented is very rudimentary (as are my scripting skills) and I use the LTSP enhancements package from PC Xperience Inc. so please contact me with any updates or bugs you may find in these instructions.

Have fun and remember

IF YOU ENJOY YOUR JOB YOU WILL NEVER WORK A DAY IN YOUR LIFE!

Robert Stanford

rob@rotapile.com