



Datum Systems, Inc.  
23 Las Colinas Lane, #112  
San Jose, CA 95119

January 20, 2008

## *SnIP Technical Note*

*DSN08-015*

Initial issue January 20, 2008

### **Subject: How to Update the SnIP Kernel and Root Filesystem Images**

This Tech Note is intended to provide SnIP users and stocking resellers with a proven method of updating a SnIP's entire default Linux kernel and Root Filesystem, bringing it up to the latest version of all files, procedures and structure. Although there is a lot of detail intended to help with the process initially, what is being done is fairly straightforward. We are going to:

- ❑ Make a TFTP server that the SnIP can access and load it with the new images we want to put into the SnIP.
- ❑ Cold boot the SnIP and stop it in the bootloader.
- ❑ Set up the bootloader's environment correctly to access the TFTP server.
- ❑ Download a script from the TFTP server that will handle downloading the actual kernel and the Root Filesystem.
- ❑ Run the script. It does most of the work.

Once you have the setup and learn the procedure it becomes very simple to update many SnIP devices.

#### ***Basic SnIP Software Storage Description:***

The SnIP has a 16 Mbyte Flash memory device which serves as the non-volatile memory and is the equivalent of the hard disk drive in a larger computer system. The flash memory is currently comprised of 128 sectors, each 128 kbytes. It is partitioned into 4 functional parts:

1. Bootloader – Starts the power up process and load the Linux kernel. 768 kbytes allocated.
2. Bootloader configuration storage – 128 kbytes allocated.
3. Linux kernel image. "ulmage" – 1 Mbyte allocated
4. Root Filesystem image – "jffs2.img". Approx 14 Mbytes allocated while the image is approximately 8.5 Mbytes.

In this Tech Note we are going to update the entire Linux kernel image and the entire Root Filesystem. This process will overwrite anything that you may have put into the root filesystem, which includes any customized configuration files you may have created. For that reason we will also discuss a method for reloading that information separately.

**What You Need to Accomplish This Update:**

In order to follow the steps below you should have the following items ready before starting:

1. A Windows or Linux computer with Internet access configured on a local LAN. This computer will become the TFTP server. Internet access is only required to download the tftp server program and the latest kernel and filesystem images. Once you have those files Internet access is no longer needed.
2. SnIP or SnIPs to be updated.
3. An Ethernet Switch or Hub device to interconnect the SnIPs to the LAN.
4. Standard Ethernet RJ45 to RJ45 cable(s)
5. Standard DB9 Male to Female serial cable. Used to make a console connection to the SnIP

The step by step procedure below assumes a Windows computer is used. Several notes are given where a Linux computer differs.

**Step 1 - Create a TFTP Server:**

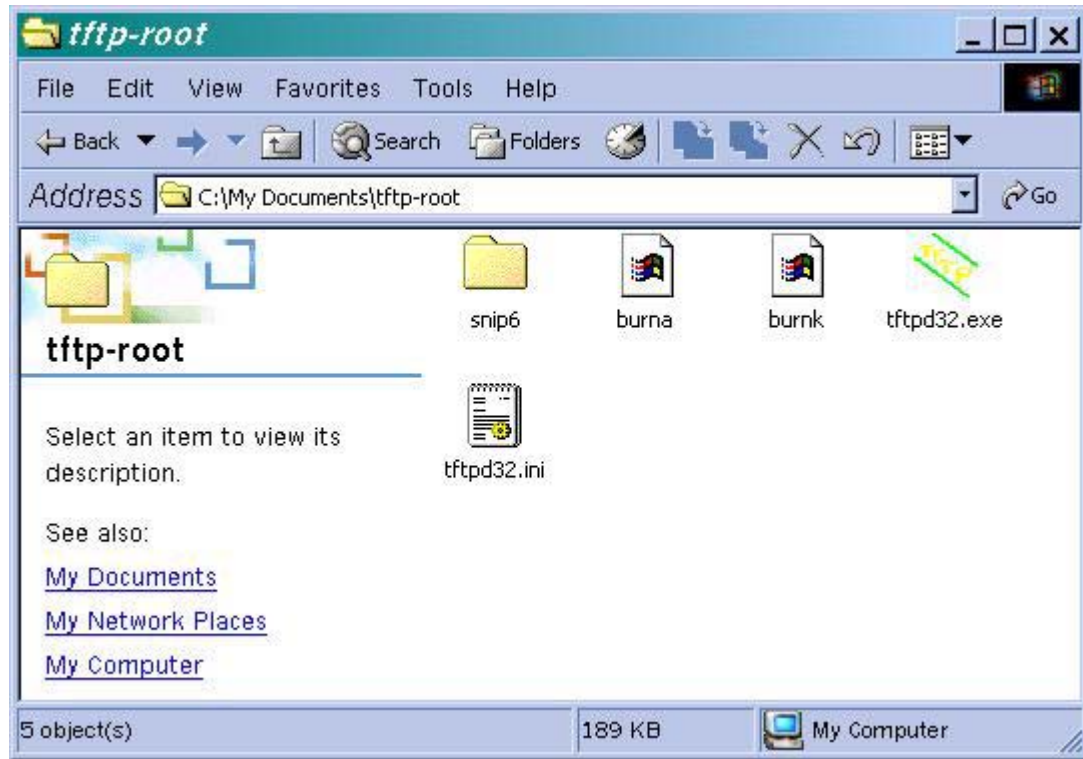
This setup installs a TFTP server program and file structure on the computer. If you are running Linux it is assumed that you can start the TFTP service and know where the base directory is (usually "/tftpboot").

The recommended Windows server program named "tftpd32" does not actually "install" itself into the Windows registry and can be easily removed if required. It is small and very safe. It comes in two versions, an install exe file and a zip file. The zip contains the files, which can be placed anywhere while the install exe file puts the files into the "Program Files" directory by default. If you use the zip version you must decide where to place this program, and it can be almost anywhere, for example in "My Documents" you could create a folder named "tftp-root" or "snip" and place it there.

**Get the TFTP server program** - Go to <http://tftpd32.jounin.net/> and download the latest version of the program. Note, we also supply this program on our CD that comes with the modem. Install the program in the selected folder.

**Set up the file structure** - Next you will need to designate a folder where you put files for TFTP access. The base folder chosen will be selected during tftpd32 setup. Create the folder if it does not already exist and then create one additional folder under it with the name "snip6" all lower case. This last folder will hold the images. For example let's assume you created a folder named "My Documents\tftp-root" and under it you have created the required "snip6" folder. We can also assume that you downloaded the

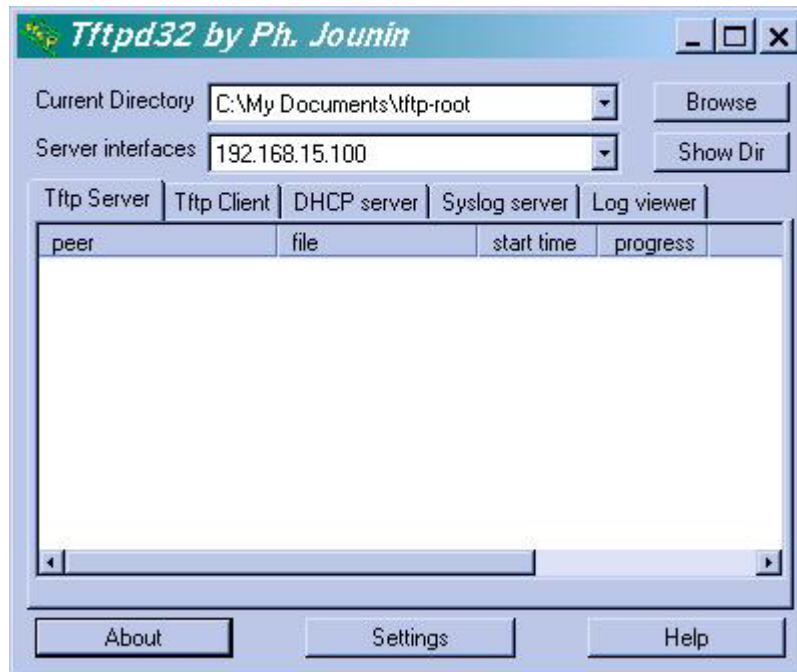
zip version of the tftpd32 program and simply extracted the program and the .ini file into the tftp-root folder. It would then look like the following.



One advantage of this type of setup is that the tftpd32 program initially will assume that it's current directory is the one it will offer to a tftp client calling in (the SnIP is the client).

Go to [http://www.datumsystems.com/snip\\_downloads.htm](http://www.datumsystems.com/snip_downloads.htm) and download the "burna" and "burnk" files into the base tftp folder chosen or created above. Then download the "ulmage" and "jffs2.img" files into the snip6 directory created above.

**Start the TFTP server** - Start the tftpd32 program by double clicking on its icon. When it is running the server is active. When you end the program you no longer have a TFTP server. You need to set up tftp32d by telling it what directory folder is the tftp base directory if it is not correctly set on initial startup.



Note the “Server interfaces” IP address shown in the program main window. This should be the IP address of the computer you are running on and will be needed in Step 4 below.

### **Step 2 - Connect the SnIP to the LAN and the Console Device:**

Connect the SnIP to the same network that the TFTP server is on using an Ethernet cable to the switch.

Connect the PC with Hyperterminal (\*\* see Note at end on using PuTTY instead) or another dumb terminal emulator to the SnIP console connector using the DB9 male to female cable. This must be a physical console connection as the following steps cannot be performed via Telnet. Hyperterminal should be in a direct connect mode emulating a ANSI or VT100 Terminal running at 38400, N, 8, 1. This is a fairly standard mode. You can also use a USB to Serial adaptor after it is installed and working as long as you know what its COM Port number is.

Now get ready to turn on the modem with the SnIP installed if it is not already on. If it is on you should be able to log into the console and type “reboot” to restart the SnIP in the next steps. We need to start from cold so that we can stop the bootloader before it loads and starts running Linux.

If the SnIP is already running in Linux then you can verify that Hyperterminal is properly connected to the SnIP. If you hit the return key when Linux is running you should be presented with a login prompt. If not

then something in Hyperterminal configuration is probably not set up correctly, for example the COM Port number.

### **Step 3 - Start the SnIP and Stop in the Bootloader:**

Now power on or reboot the SnIP. If the console is set up correctly then you should see information and messages streaming by. If you do not then something in Hyperterminal is probably not set up correctly, for example the port number.

Within approximately 6 seconds of power-on the SnIP will hold for 5 seconds with a prompt that says:

=>Hit any key to stop autoboot: 1 (Older versions of bootloader)

Newer bootloader versions will show instead.

=>autoboot in 4 seconds (stop with 'qt')...

Press the "q" then the "t" keys on your keyboard.

The bootloader should halt and present the bootloader's prompt "=>".

To check that we are operating correctly go to the next step.

### **Step 4 - Setup and Verify that the SnIP can Access the TFTP Server:**

We need to tell the bootloader what addresses to use to access the TFTP server. Type the following at the prompt:

=> **printenv** /\* short for print environment \*/.

Look at our IP Address and the Server IP Address. You will probably need to set both since they are usually at the factory default values. Your SnIP IP Address **must** be in the same network mask range as the server. Type the following commands:

=> **setenv ipaddr <desired ip address>**

=> **setenv serverip <address of the server from step 1>**

=> **printenv** /\* again to insure they are properly set \*/

=> **saveenv** /\* to save the environment settings in the configuration file \*/

By saving the environment settings you can perform an update in another session without having to set the IP Addresses again. It is not essential for the current session however.

To verify that we are set up and can access the server type the following command at the console: The response from the process is shown below.

=> **ping <server ip address>** (for example 192.168.15.101)

And the response should be:

Host 192.168.15.101 is alive

If the ping worked you might as well really test the server connection by actually downloading a small file (in this case it is the first command of Step 5). The response from the process is shown below.

```
=> tftp 100000 burna  
Using FEC ETHERNET device  
TFTP from server 192.168.15.100; our IP address is 192.168.15.83  
Filename 'burna'  
Load address: 0x100000  
Loading #  
done  
Bytes transferred = 727 (hex 2d7)  
=>
```

There are 5 zeros after the 1 in the tftp command.

If for some reason this command was not successful, you need to recheck everything and try it again. Note that if the tftp command had trouble it will likely try a few times and then change the interface port to the modem's hdlc port. You may need to use the "ethact" command to set the active port to the RJ45 Ethernet port like this (or do a reset to start the bootloader again).

```
=> setenv ethact FEC ETHERNET
```

Now we are ready to perform the update.

### **Step 5 - Perform the Actual Update:**

Once we know that everything works, the actual image updates are fairly simple. Note that the bootloader is locked in the flash and even if something catastrophic happens you can always come back to this point and reload the kernel and root filesystem.

At the boot load command prompt (that is the "=>" shown below to indicate you are at the prompt) type the following. You can skip this step if you already did this at the end of step 4. Do not type the "=>".

```
=> tftp 100000 burna  
There are 5 zeros in the number which represent the address that you will load to. The bootloader should do this and respond with the messages:  
Using FEC ETHERNET device  
TFTP from server 192.168.15.100; our IP address is 192.168.15.83  
Filename 'burna'  
Load address: 0x100000  
Loading #  
done  
Bytes transferred = 727 (hex 2d7)  
=>
```

If it does not then something is wrong. Go back and check your setup again.

Now the “burna” script is loaded into memory (at address 100000 hex) and we only have to execute the script by typing:

```
=>autoscr
```

The script will tell you what it is doing as it progresses through each step of the process. It will first download the kernel and burn it to flash, then download the Root Filesystem and burn it to flash. The kernel update only takes about a minute, while the following Root Filesystem image update takes about 6 minutes. Be patient and do not interrupt the process. When finished it will say:

```
=> done
```

And that is it. The update reload is complete!

### **Step 6 – Reboot the SnIP:**

The bootloader uses the “**reset**” command to perform a cold boot. Type that now at the terminal prompt and this time allow the normal process to proceed through the bootloader to load and start running Linux.

The first thing you should do is run the “configwiz” program and set up your LAN, WAN and Gateway Addresses plus LinkMode as desired. Since this is a newer version you may notice new entries. After running configwiz you should reboot once more so that the new addresses can be used to set up the ports correctly.

### **What about your configuration files?**

When you reloaded the Root Filesystem any changes you had made to the configuration was overwritten. You may not need to preserve anything, and the changes can be made again if you remember what they were. But if you have extensive static-routes or iptables then you may want to back them up for recovery in the new Root Filesystem. Following is a simple procedure to save the appropriate files to the TFTP server created and recover them. Let us assume for this example that the following files need to be saved:

```
/etc/config/system (these are the settings from configwiz)
```

```
/etc/config/static-routes
```

```
/etc/config/iptables
```

```
/etc/config/eatables-config
```

```
/etc/config/daemons (Quagga daemons)
```

```
/etc/quagga/*.conf (Quagga configuration files)
```

Saving and reloading the /etc/config/system file is always of dubious value since you have to run the configwiz program after the reload anyway to set the IP Address to allow using other Ethernet based programs.

**Method 1 – Using WinSCP with a Windows PC**

WinSCP use is described in the SnIP Appendix or Quick Start Guide. You can use this program to save your unique configuration files on the computer running WinSCP. After reloading a new Root Filesystem you would use WinSCP again to put those same files back into the newly reloaded SnIP. The /etc/config/system file will probably be of the wrong version number and configwiz should be run again, saving the settings to update to the new configuration version.

**Method 2 – Using SCP or Rsync with a Linux PC**

Both SCP and Rsync use is described in the SnIP Appendix or Quick Start Guide. You can use either of these programs to save your unique configuration files on the Linux PC. After reloading a new Root Filesystem you would use the same program again to put those same files back into the newly reloaded SnIP. The /etc/config/system file will probably be of the wrong version number and configwiz should be run again, saving the settings to update to the new configuration version.

***Resolving Problems:***

The tftpd32 server program is possibly easier to use than the Linux equivalents. It still must be set up properly.

***Notes:***

Hyperterminal does not work very well for me and many other people I know. One main problem seems to be that the scroll-back is garbled on over half the PCs I've seen. So I often use the very good open source PuTTY program in "Serial" mode. You can create a serial port emulator by setting the session type to "Serial" and the Serial category's values to the proper COM Port, 38400, N, 8, 1 and use none or XON/XOFF flow control. PuTTY web site: <http://www.chiark.greenend.org.uk/~sgtatham/putty/>

For questions concerning this technical note please contact Datum Systems, Inc via email at [support@datumsystems.com](mailto:support@datumsystems.com)