Upload via TFTP Server

Introduction

Cisco TelePresence Non-Immersive endpoints contain two software images. These can be found by typing the command selectsw when logged in as root. They are usually shown in the following format:

```
image1
image2 [ACTIVE] [SELECTED]
```

We use this method in those scenarios where there is no bootable image present which can be used to boot up the device.

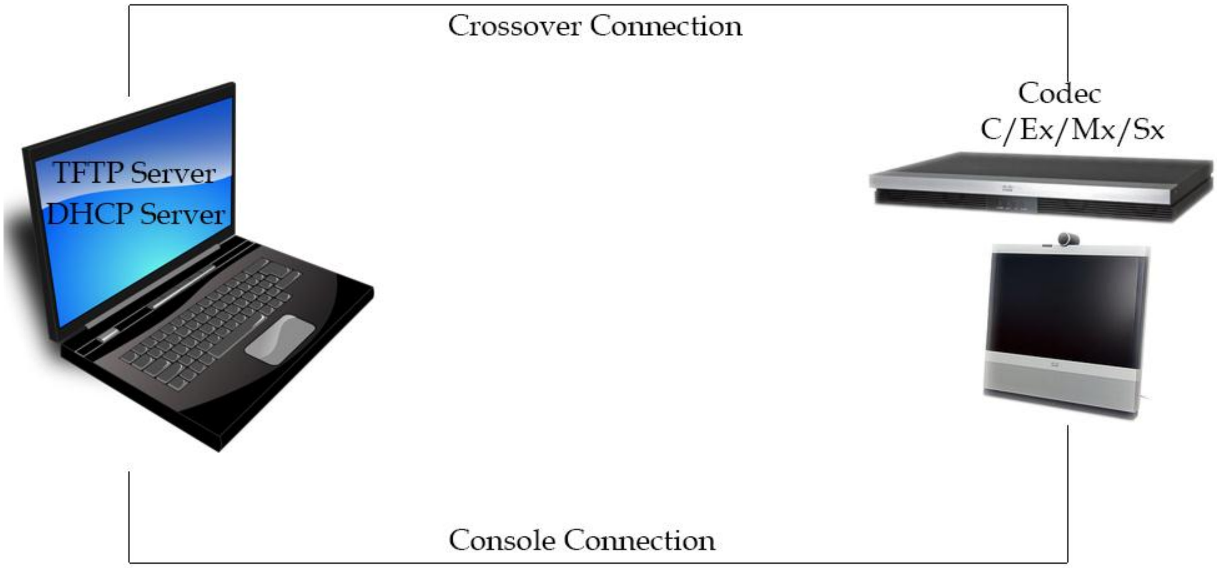
Here’s a list of the things you will need for this procedure:
TFTP Server: To store the package file from which the codec can download the file.
DHCP Server: To assign an IP address to the codec so that it can communicate on LAN.
Crossover Cable: To have IP connectivity to the codec
Laptop: To give commands and input to the codec.
Console cable: To have a means to be able to give commands to the codec.

There are two ways the following process can be successfully done:

1. If you have an external DHCP & TFTP Server and a console session on your codec.
2. If you use application simulated DHCP & TFTP Server on the laptop and also use the laptop to console into codec.

For this particular document, I will be using the second example since that is more common and complicated.

Connections Diagram:



U-Boot Explained

This section will explain how the process works and what is the theory applied behind this procedure. For all those who are only interested in the steps and not looking for any more information, can skip this section.

Normally when the codec is not booting up, it means that the software image is corrupted. Using the following procedure, we would be able to upload a fresh software image to the codec. The software is uploaded to the hard disk and then it is installed. Once the process is started, the hard disk is first erased which would delete any previous software on the device. When I tried to do it once on my system, even my release and option keys were wiped out and so was the configuration. So it is always recommended that whenever you have a working system, you should always take a backup.

The process is done using the U boot which is a special interface with some useful commands. U boot is either automatically activated when there is no software image present in the system, or it can be manually triggered by a user. During the task, there are many instances where the process might fail. In this document I will be highlighting all such instances where there are chances of process failure.

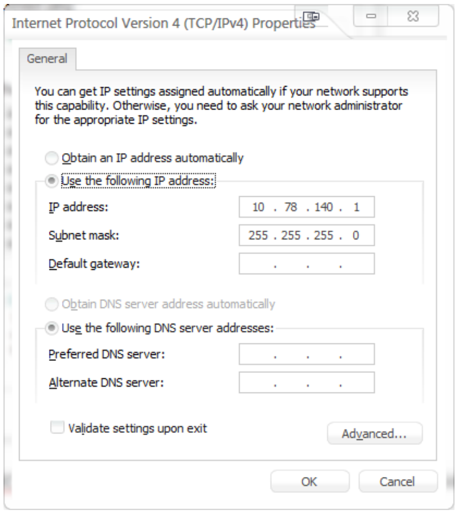
Procedure

Step 1: MAKE PHYSICAL CONNECTIONS AS SHOWN IN THE DIAGRAM ON PAGE 1

We need to have one console connection and one Ethernet connection between the codec & laptop. Ethernet connection should be made using the Crossover cable.

Step 2: ASSIGN A STATIC IP TO THE LAPTOP

Once the connections are made, choose the interface connected to your codec (in case if you have multiple interfaces) & assign a static IP with the following details:



Internet Protocol Version 4 (TCP/IPv4) Properties

General

You can get IP settings assigned automatically if your network supports this capability. Otherwise, you need to ask your network administrator for the appropriate IP settings.

○ Obtain an IP address automatically

● Use the following IP address:

IP address: 10 . 78 . 140 . 1

Subnet mask: 255 . 255 . 255 . 0

Default gateway: . . .

○ Obtain DNS server address automatically

● Use the following DNS server addresses:

Preferred DNS server: . . .

Alternate DNS server: . . .

□ Validate settings upon exit

Step 3: BREAK THE BOOT SEQUENCE

The first process is to reach the Uboot prompt. Either you will be automatically sent to the U boot prompt or you can activate the U boot when the codec is booting up. You will require a console session with the codec in order to see the prompt. When you are on the console session, you would see this output when the codec is booting up:

```
CPU:   e300c1, MPC8347_TBGA_EA, Rev: 3.0 at 600 MHz, CSB: 200 MHz
I2C:   ready
DRAM:  DDR2 RAM: 512 MiB
RAM data bus test  [PASS]
RAM address bus test          [PASS]
FLASH: j¥ 5
NAND:  2048 MiB
RTC:   2012-10-05 (Friday) - Time: 13:33:37
PCI Video Processor Detection            [PASS]
Falcon
      SN: A1AR44D00273
      Object level: 05
Falcon main board
      Rev: I
      SN: F1004485
      Object level: 68
      MAC: 00:50:60:06:4B:BD
Net:   TSEC0, TSEC1
Hwmon: 45
Press 'b' to enter u-boot prompt
Press 'c' to stop autoboot:
```

The following text “Press ‘b’ to enter u-boot prompt” & “Press ‘c’ to stop autoboot” will show up on the screen. As soon as you see it, you need to hold ‘B’ key and press ‘C’ key.

There is a short window open for the process, hence it should be very quick.

Once the boot sequence is broken, you would see a prompt come up on the console session. The

prompt would differ as per the device you're using. Here are the different prompts:
FALCON> For Ex90
AMERICA> For C40
CASPER> For C60
SATURN> For C90
VENETO> For MX300

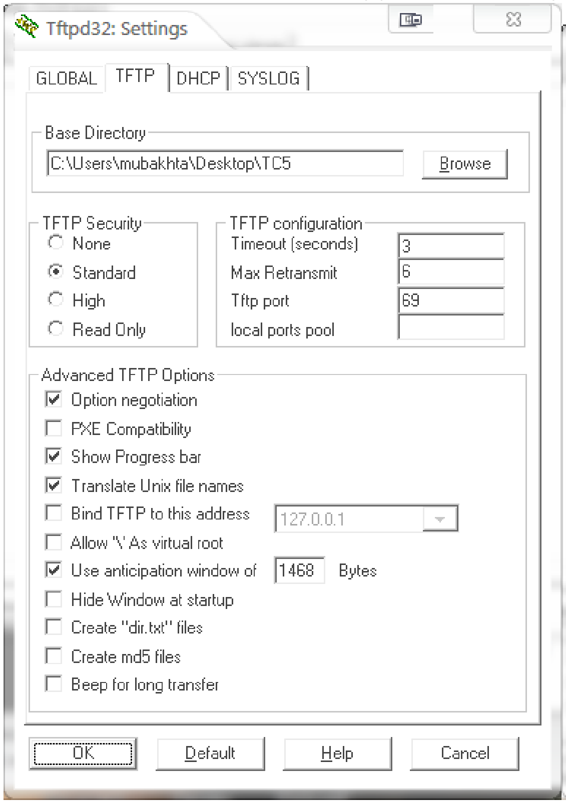
Once we have reached this prompt, the boot sequence is broken.

STEP 4: ASSIGN IP ADDRESS TO CODEC VIA DHCP SERVER

We now have to assign an IP address to the codec so that it is able to download the file from the TFTP server. There is an open source application available called TFTPd32. Download link: http://tftpd32.jounin.net/tftpd32_download.html

This application can be used for both the purpose; TFTP Server & DHCP Server.

Here's the screenshot of the application on my laptop:

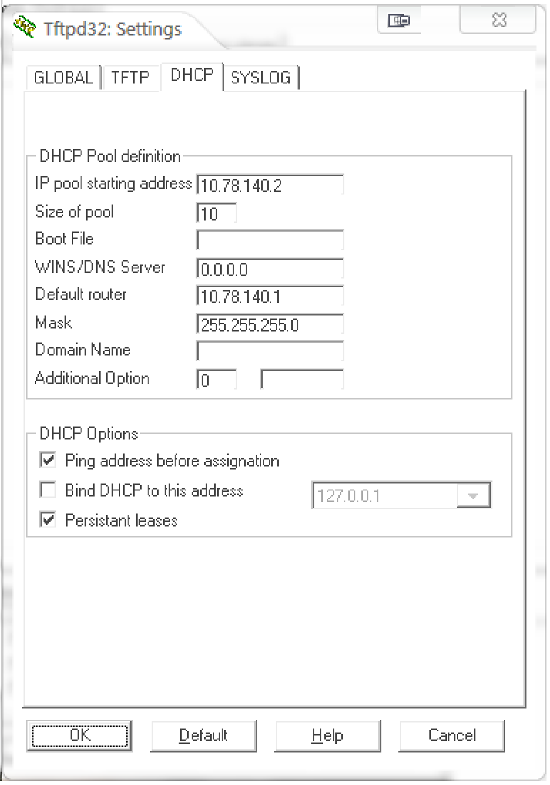


Base Directory
C:\Users\mubakhta\Desktop\TC5

TFTP Security
○ None
● Standard
○ High
○ Read Only

TFTP configuration
Timeout (seconds) 3
Max Retransmit 6
Tftp port 69
local ports pool

Advanced TFTP Options
☑ Option negotiation
☐ PXE Compatibility
☑ Show Progress bar
☑ Translate Unix file names
☐ Bind TFTP to this address 127.0.0.1
☐ Allow '\' As virtual root
☑ Use anticipation window of 1468 Bytes
☐ Hide Window at startup
☐ Create "dir.txt" files
☐ Create md5 files
☐ Beep for long transfer



DHCP Pool definition
IP pool starting address 10.78.140.2
Size of pool 10
Boot File
WINS/DNS Server 0.0.0.0
Default router 10.78.140.1
Mask 255.255.255.0
Domain Name
Additional Option 0

DHCP Options
☑ Ping address before assignation
☐ Bind DHCP to this address 127.0.0.1
☑ Persistant leases

However, most of the times we have seen that using this application as a DHCP as well as TFTP create issues. The DHCP server does not assign IP addresses and if you open the Log viewer tab on the application, you will see these errors:

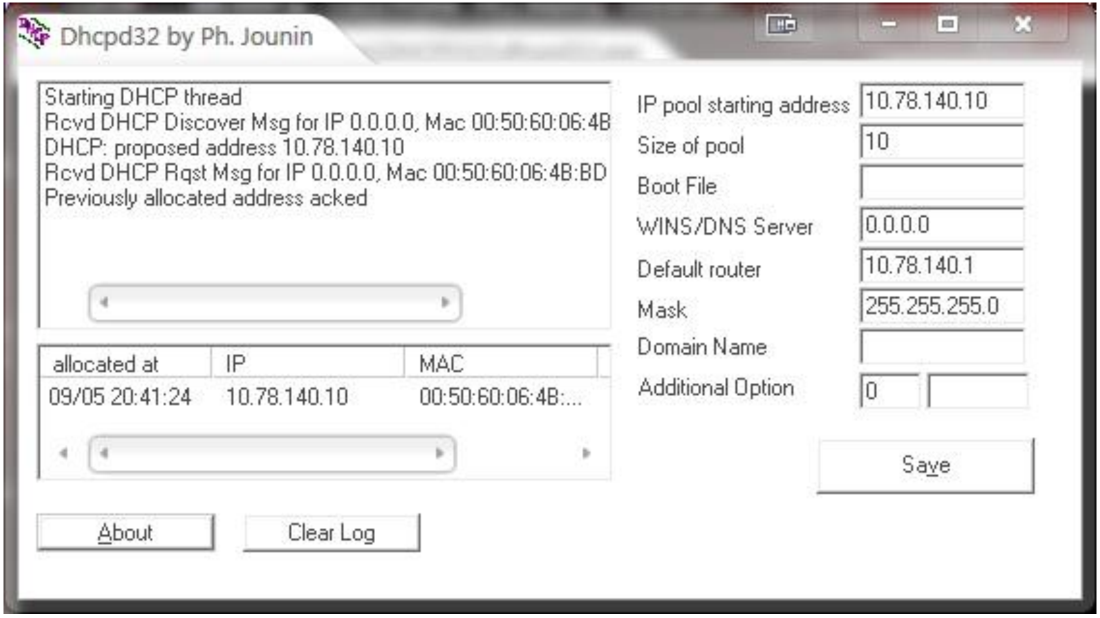
```
Rcvd DHCP Discover Msg for IP 0.0.0.0, Mac 00:50:60:06:4B:BD [05/10 20:20:07.458]
Client requested address 0.0.0.109 [05/10 20:20:07.475]
Suppress pingable address 10.78.140.2 [05/10 20:20:16.217]
```

Another indication is when you type these commands on the Uboot prompt, you will see the following results:

```
FALCON> dhcp
BOOTP broadcast 1
BOOTP broadcast 2
BOOTP broadcast 3
BOOTP broadcast 4
BOOTP broadcast 5
Retry count exceeded; starting again
FALCON> printenv ipaddr
## Error: "ipaddr" not defined
```

If you see these errors, this means that the DHCP server is not working well. In that case I would recommend having DHCP & TFTP run on different applications rather than running on the same. DHCPd32 is one open source application which can be used for this purpose. Download link: http://www.softpedia.com/progDownload/DHCPD32-Download-77229.html
FIRST close the TFTPd32 application window and THEN run DHCPd32 (running DHCPd32 first is required so that bootp agent which is used to assign IP addresses is used for DHCPd32 rather TFTPd32)

Here's a screenshot of the DHCPd32 application with the settings:



```
Starting DHCP thread
Rcvd DHCP Discover Msg for IP 0.0.0.0, Mac 00:50:60:06:4B
DHCP: proposed address 10.78.140.10
Rcvd DHCP Rqst Msg for IP 0.0.0.0, Mac 00:50:60:06:4B:BD
Previously allocated address acked
```

allocated at	IP	MAC
09/05 20:41:24	10.78.140.10	00:50:60:06:4B:...

IP pool starting address	10.78.140.10
Size of pool	10
Boot File	
WINS/DNS Server	0.0.0.0
Default router	10.78.140.1
Mask	255.255.255.0
Domain Name	
Additional Option	0

You will notice the logs on the left hand side top corner.
Go to the codec U boot prompt and type this command:

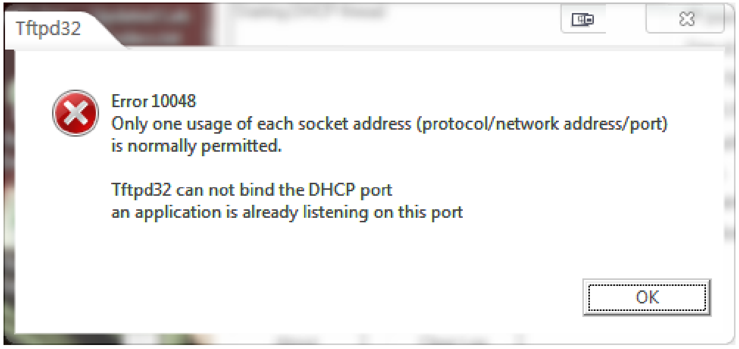
```
FALCON> dhcp
BOOTP broadcast 1
DHCP client bound to address 10.78.141.252
FALCON> printenv ipaddr
ipaddr=10.78.140.10
```

Also logs can be seen on the DHCPd32 window top left corner. This shows the IP address 10.78.140.10 was successfully assigned to the codec.

Step 5: START TFTP SERVER WITH ACCESS TO FILES REQUIRED

Mainly the package files are only required however, in the case of C series codec, you would also require two files namely; pkgextract.elf & u-boot.ini. These files can be downloaded from the following link: https://supportforums.cisco.com/servlet/JiveServlet/download/23161-1-123930/pkgfiles_uboot.rar.zip

Create a TC5 folder on your desktop so that they contain all the files required for the process. Run the TFTPd32 in order to configure it. If you followed the steps and ran DHCPd32 first, you will see this error upon running TFTPd32:



Tftpd32

Error 10048
Only one usage of each socket address (protocol/network address/port) is normally permitted.

Tftpd32 can not bind the DHCP port
an application is already listening on this port

Ignore it, it is just telling you that DHCP can only be used by one application and is currently being used by DHCPd32 hence TFTPd32 will not be able to use it. Now point the TFTP server to the correct directory TC5 on your desktop. Also if you have multiple network interfaces on the laptop, make sure the Server Interface is showing the IP address assigned to the interface connected with the codec.

Step 6: CONFIGURE THE CODEC FOR DOWNLOADING FILES FROM TFTP

```
FALCON> setenv serverip 10.78.140.1
FALCON> setenv tftp_path (blank since files in the root directory of TFTP)
FALCON> setenv saturnpkg s52000tc5_1_4.pkg
```

Step 7: START UPLOAD OF FILES FROM THE TFTP

We will now start the upload process:

```
FALCON> run pkgex
BOOTP broadcast 1
*** Unhandled DHCP Option in OFFER/ACK: 7
*** Unhandled DHCP Option in OFFER/ACK: 44
*** Unhandled DHCP Option in OFFER/ACK: 7
*** Unhandled DHCP Option in OFFER/ACK: 44
DHCP client bound to address 10.78.140.10
Unmounting yaffs2 mount point /flash

NAND erase: device 0 whole chip
Erasing at 0x0 --   0% complete.Erasing at 0x1440000 --   1% complete.Erasing at 0x28c0000 --   2% complete.Erasing at
0x3d40000 --   3% complete.Erasing at 0x51c0000 --   4% complete.Erasing at 0x6640000 --   5% complete.Erasing at 0x7ac0000
--   6% complete.CQ…Í¥¹⍰ at 0x8f40000 --   7% complete.Erasing at 0xa3c0000 --   8% complete.Erasing at 0xb840000 --   9%
complete.Erasing at 0x18500000 --  19% complete.Erasing at 0x19980000 --  20% complete.Erasing at 0x1ae00000 --  21%
complete.Erasing at 0x1c280000 --  22% complete.Erasing at 0x1d700000 --  23% complete.Erasing at 0x1eb80000 --  24%
complete.Erasing at 0x1ffc0000 --  25% complete.Erasing at 0x21440000 --  26% complete.Skipping bad block at  0x21dc0000
```

```
Erasing at 0x228c0000 --  27% complete.Skipping bad block at  0x23a80000
complete.CQ…Í¥¹⍰ at 0x5c280000 --  72% complete.Erasing at 0x5d700000 --  73% complete.Erasing at 0x5eb80000 --  74%
complete.Erasing at 0x5ffc0000 --  75% complete.Erasing at 0x61440000 --  76% complete.Erasing at 0x628c0000 --  77%
complete.Erasing at 0x63d40000 --  78% complete.Erasing at 0x651c0000 --  79% complete.Erasing at 0x66640000 --  80%
complete.Erasing at 0x67ac0000 --  81% complete.Skipping bad block at  0x68780000
Erasing at 0x68f40000 --  82% complete.Erasing at 0x6a3c0000 --  83% complete.Erasing at 0x6b840000 --  84%
complete.Skipping bad block at  0x6bfc0000
Erasing at 0x6ccc0000 --  85% complete.Erasing at 0x6e140000 --  86% complete.Erasing at 0x6f5c0000 --  87%
complete.Skipping bad block at  0x6fc40000
Erasing at 0x70a00000 --  88% complete.Skipping bad block at  0x71100000
Erasing at 0x71e80000 --  89% complete.Erasing at 0x73300000 --  90% complete.Erasing at 0x74780000 --  91%
complete.Erasing at 0x75c00000 --  92% complete.Erasing at 0x77080000 --  93% complete.Skipping bad block at  0x78480000
Erasing at 0x78500000 --  94% complete.Erasing at 0x79980000 --  95% complete.Erasing at 0x7ae00000 --  96%
complete.Skipping bad block at  0x7bec0000
Erasing at 0x7c280000 --  97% complete.Erasing at 0x7d700000 --  98% complete.Erasing at 0x7eb80000 --  99%
complete.Skipping bad block at  0x7fa40000
Erasing at 0x7ffc0000 -- 100% complete.
```

Till this part, we can see that the IP address was identified from the DHCP & hard disk erased in order to download and install the new package. After this, you should see the following :

```
Mounting yaffs2 mount point /flash
Mount point name already used
Using TSEC0 device
TFTP from server 10.78.140.1; our IP address is 10.78.140.10
Filename 'TC5/s52000tc5_1_4.pkg'.
Load address: 0x2000000
```

Filename shows the exact location of the file which the codec is looking to download. If the path is incorrect, you will get the following error:

```
Filename 'TC5/test.pkg'.
Load address: 0x2000000
Loading: *
TFTP error: 'File not found' (1)
Not retrying...
```

The last hiccup to download the file from the TFTP would be the MTU size. You might encounter the following error:

```
TFTP from server 10.78.140.1; our IP address is 10.78.140.10
Filename '/s52000tc5_1_4.pkg'.
Load address: 0x2000000
Loading: *Ethernet ERROR: IP fragmentation
  Check ethernet MTU on server
Ethernet ERROR: IP fragmentation
  Check ethernet MTU on server
```

This error shows up because it is required for the TFTP server to have the MTU of 1500. Hence, on your PC the MTU should be configured as 1500 which is not of this value by default. You can follow these steps to set it to 1500.

On Windows 7 PC:
Open a command line window as an Administrator (ie. right click on All Programs > Accessories > Command Prompt and select Run as administrator) ...
Type the command netsh and wait for prompt
Type the command interface and wait for prompt
Type the command ipv4 and wait for prompt

Type the command set subinterface "Local Area Connection" mtu=xxxx store=persistent
exit [Enter]
Eg:

```
C:\Users\mubakhta>netsh
netsh>interface
netsh interface>ipv4
netsh interface ipv4>set subinterface "Local Area Connection" mtu=1500 store=per
sistent
Ok.
```

Finally if everything goes well, you will see the file getting downloaded:

```
TFTP from server 10.78.140.1; our IP address is 10.78.140.10
Filename '/s52000tc5_1_4.pkg'.
Load address: 0x2000000
Loading: *#############################################ÿä##################ä
          ##########ÿä######################################################
          #######ÿä##########ÿä#####ÿä#######################ä##ÿä############
          ###############ÿä#################################################
          #################################################################
          ######ä########################################ÿä#################
          #################################################ä###############
          ##################################ä#########
done
Bytes transferred = 215957872 (cdf4170 hex)
```

Once all the processing is complete, you will return to the same prompt. Type boot to start booting the codec:

```
FALCON> boot
Configure FPGAs
\|/-
Setting up Davinci HD (00.10.00)... 675Mhz          [DONE]
Setting up Davinci HD (00.11.00)... 675Mhz          [DONE]
Setting up Davinci HD (00.12.00)... 675MhzJ [DONE]
Setting up Davinci HD (00.13.00)... 675MhzJ [DONE]
Setting up Davinci HD (00.14.00)... 675Mhz          [DONE]
Copy /flash/image1/arm/rootfs to 0x03000000...             [DONE]
Copy /flash/image1/arm/kernel to 0x02000000...             [DONE]
Copy /flash/image1/arm/u-boot.bin to 0x01000000...         [DONE]
Copy /flash/image1/kernel to 0x01000000...       [DONE]
Copy /flash/image1/rootfs to 0x02000000...       [DONE]
Copy /flash/image1/dtb3 to 0x03000000...         [DONE]
Unmounting yaffs2 mount point /flash
```

You should now see the login prompt and when you login as admin, the API should now be accessible:

```
login: admin
Welcome to
TANDBERG Codec Release TC5.1.4.295090
SW Release Date: 2012-08-10
OK
```

If the codec still does not boot up, break the boot sequence again as shown in Step 3 and restore the codec to factory default settings using the following command:

```
FALCON> setenv othbootargs factorydefault
```

Once done boot the codec again and you should now be able to login.