

# PC Deployment over network using PXE environment

**Workshop**

---

September 2005

## Table of Contents

1.0 Introduction.....	4
1.1. Terminology.....	4
1.2 Overview.....	4
2.0 Background.....	5
2.1 The Issue.....	5
2.2 The Solution.....	5
2.3 Overview.....	5
2.4 How does PXE work?.....	5
3. Workshops.....	7
3.1. Workshop №1 – using Microsoft Software (DHCP and TFTP).....	7
3.1.1 What we need to carry out this workshop:.....	7
3.1.2 Step by step instruction:.....	7
Step 1. Downloading.....	7
Step 2. PXE folder creating.....	8
Step 3. Installing DHCP and TFTP.....	8
Step 4. Setting up TFTP Server.....	9
Step 5. Setting up DHCP Server.....	10
Step 6. Additional settings for the DHCP Server.....	12
Step 7. Boot disk creating.....	14
Step 8. Syslinux files preparing.....	15
Step 9. pxelinux.cfg creating.....	15
Step 10. Boot Image creating.....	16
Step 11. Registry editing.....	17
Step 12. Finishing.....	18
3.2. Workshop №2 – using third-party software under Windows OS.....	18
3.2.1 What we need to carry out this workshop:.....	18
3.2.2 Step by step instruction:.....	19
Step 1. Downloading.....	19
Step 2. PXE folder creating.....	19
Step 3. Boot disk creating.....	19
Step 4. Syslinux files preparing.....	19
Step 5. pxelinux.cfg creating.....	19
Step 6. Boot Image creating.....	19
Step 7. Installing Tftpd32.....	19
Step 8. Starting up Tftpd32.....	20
Step 9. Setting up DHCP.....	20
Step 10. Finishing.....	22
3.3. Workshop №3 – similar to the workshop №2 but using alternative third-party software.....	22
3.3.1 What we need to carry out this workshop:.....	22
3.3.2 Step by step instruction:.....	23
Step 1. Downloading.....	23
Step 2. PXE folder creating.....	23

Step 3. Boot disk creating.....	23
Step 4. Boot Image creating.....	23
Step 5. Boot menu creating.....	24
Step 6. Boot menu and Boot Image copying.....	25
Step 7. Installing Tftpd32.....	25
Step 8. Starting up Tftpd32.....	25
Step 9. Setting up DHCP.....	25
Step 10. Finishing.....	26
3.4. Workshop №4 – using Linux software only.....	26
3.4.1 What we need to carry out this workshop:.....	27
3.4.2 Step by step instruction:.....	27
Step 1. Downloading.....	27
Step 2. PXE folder creating.....	27
Step 3. Installing and setting up DHCP Server.....	27
Step 4. Installing TFTP Server.....	29
Step 5. Syslinux files preparing.....	29
Step 6. Starting up all services.....	31
Step 7. Finishing.....	32

## 1.0 Introduction

### 1.1. Terminology

Term	Description
PXE	Preboot Execution Environment
LAN	Local Area Network
NIC	Network Interface Card
DHCP	Dynamic Host Configuration Protocol
TFTP	Trivial File Transfer Protocol
HDM	Hard Disk Manager
DB	Drive Backup
RIS	Remote Installation Services

### 1.2 Overview

PXE is an open industry standard developed by a number of software and hardware vendors. It was initially designed by Intel, with input from several other vendors including 3Com, HP, Dell, Compaq, and Phoenix Technologies. PXE works with a network interface card (NIC) in the PC, and makes the NIC a boot device. The PXE vision is to "Make the network interface a standard, industry-accepted PC boot device." This means adding the NIC to the traditional list of standard boot devices, such as floppy drives, hard disks, and CD-ROMs, that load the operating system or set up programs on the PC. It allows the client PC to "network boot." Booting from the network opens up a vast array of management and support features.

PXE boots the client PC from the network by transferring a "boot image file" from a server. This file can be the operating system for the client PC or a small operating environment to perform a client management task. Since PXE is not operating system-specific, the image file can load any OS. It provides support for network booting, of embedded and other operating systems.

Because PXE works with the NIC, it requires a PXE-enabled NIC. Most currently available NICs do support PXE, including those from 3Com, Intel, Digital, RealTek, and SMC. PXE is available either as a boot ROM chip that you add to the NIC, or as part of the system BIOS if the network interface is on the motherboard. PXE is specific to a type of NIC; a boot ROM for one type (for example, a 3C905C) will not work on another type of NIC.

## 2.0 Background

### 2.1 The Issue

In case there are no internal CD-ROMs or floppy disk drives in most PC systems and it is not possible to connect them, restoring or backuping for these PCs must be performed through the LAN even if all backup images are located on another PC in the LAN. In order to use a backup software, namely HDM or DB, we must first get the system booted into DOS or Linux and then start up Hard Disk Manager or Drive Backup for DOS or Linux. This can be done using a bootable Floppy/CD disks or by booting over the network. In order to reduce the hassle of keeping up removable media, we have set our PXE server up to act as a complete restore solution for the DOS or Linux client. The restore process begins by connecting our PXE client and PXE server together through the LAN. We can then PXE boot our PXE client, load a DOS or Linux boot disk image with network support and start up HDM or DB. Our backup files can be stores on any PC in the LAN so restoring over the network is very easy at that point.

### 2.2 The Solution

Restoring and backuping can be managed centrally from the server using the PXE scheme, which is particularly useful in managing PSs that needs frequent backup or restore. The PXE solution is recommended in the event you don't have external CD-ROMs or other storage media for restoring or backuping.

### 2.3 Overview

This document provides step-by-step instructions on setting up the PXE environment using Windows and Linux operating systems as the PXE servers with DOS and Linux clients as PXE clients.

To do PXE boot on the PC, press F12 during the BIOS POST, this will skip the boot sequence setting in BIOS and go directly to network boot. You can also go to the BIOS and set LAN as **First Boot Device**.

### 2.4 How does PXE work?

There are five steps involved in remote booting:

1. Gathering information about the client's configuration. In this phase, the client computers establish communication with a server using the DHCP protocol to get the basic information needed to proceed to the next stages.

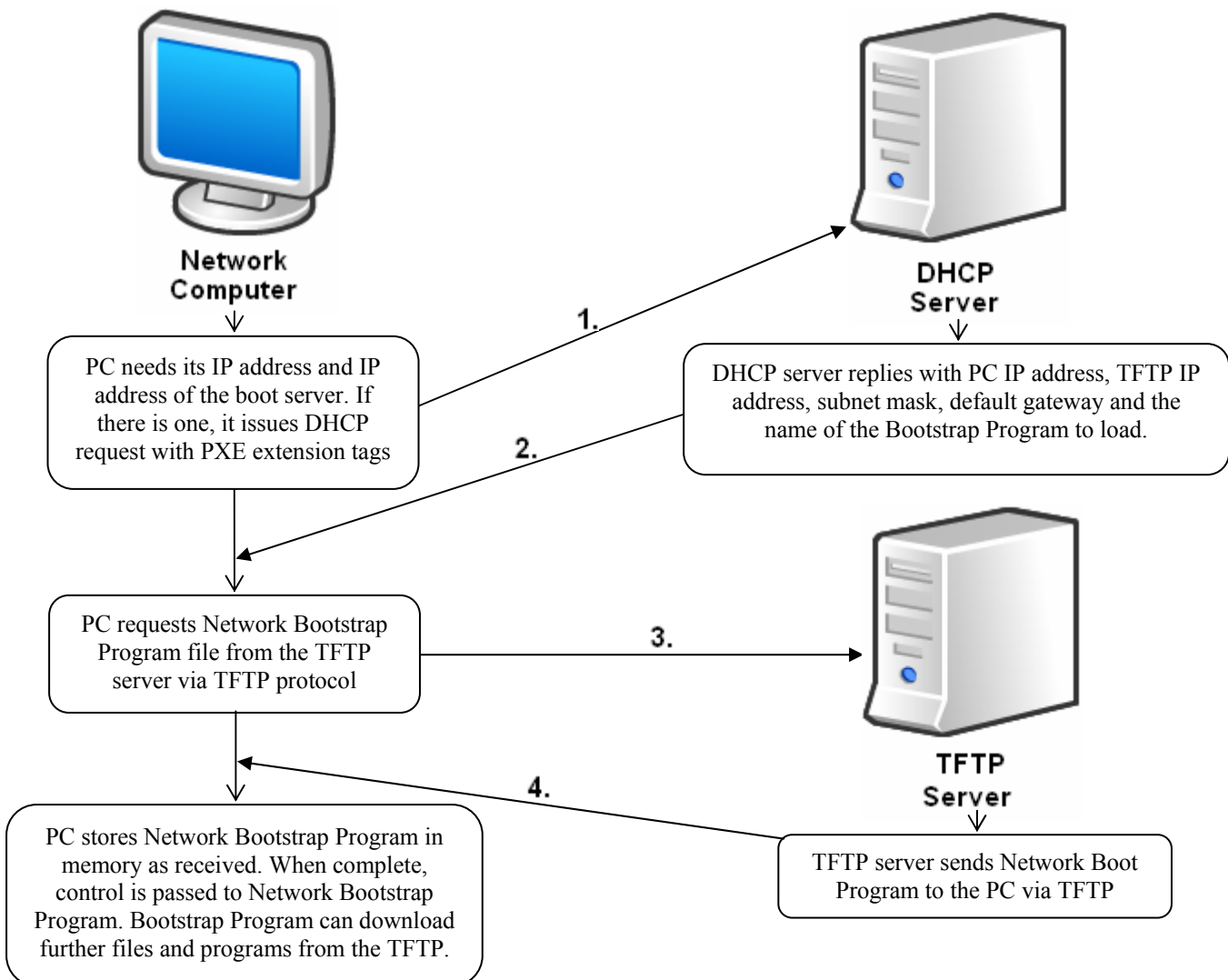
2. DHCP Server replies with this critical information that includes the IP address, TFTP IP address, subnet mask, default gateway and the name of the Network Bootstrap Program.

3. PC requests the Network Bootstrap Program file from the TFTP server via TFTP protocol.

4. Downloading the Network Bootstrap Program. The bootstrap program is the core of the remote-booting process. It is permanently stored on the server's hard disk and transferred to the client computer on demand using the TFTP protocol. The task of the bootstrap program is to prepare the client to run the specified program or operating system as configured by the system administrator.

5. Executing the bootstrap program, which typically leads to download and launching the operating system.

The diagram below shows a synopsis of the PXE protocol:



### **3. Workshops.**

This document describes four different workshops where different OSs and software are used.

The first workshop describes how to prepare PXE environment using Microsoft software under Windows Server OS. The second and third workshops describe how to prepare PXE environment using third-party software under Windows OS. The fourth workshop describes how to prepare PXE environment under Linux OS.

#### **3.1. Workshop №1 – using Microsoft Software (DHCP and TFTP).**

This workshop provides step-by-step instructions on setting up the PXE environment using Windows operating system with Microsoft DHCP Server that already is installed or not and DOS or Linux as clients with network support.

##### **3.1.1 What we need to carry out this workshop:**

1. PC with a PXE-enabled NIC.
2. Second computer running Windows with Microsoft DHCP Server to act as the PXE server.
  1. Wired network connecting the PC to the PXE server.
  2. TFTP server from Windows 2002, 2003 Server as the part of RIS
3. DOS boot disk with or without network support. You can also download Paragon PXE Network Deployment Package where you can find universal DOS boot image file with network support for all PXE-enabled NIC. The Paragon PXE Network Deployment Package contains also Paragon Recovery PXE files to boot Paragon Recovery CD with network support (Linux part only).
4. Syslinux-3.07.zip or higher from [kernel.org](http://kernel.org).
5. WinImage software for creating the boot disk image. Download winimaXX.exe from [WinImage](#)

##### **3.1.2 Step by step instruction:**

###### **Step 1. Downloading.**

Begin by downloading the necessary software and files into a temporary folder on the computer that will act as the PXE Server.

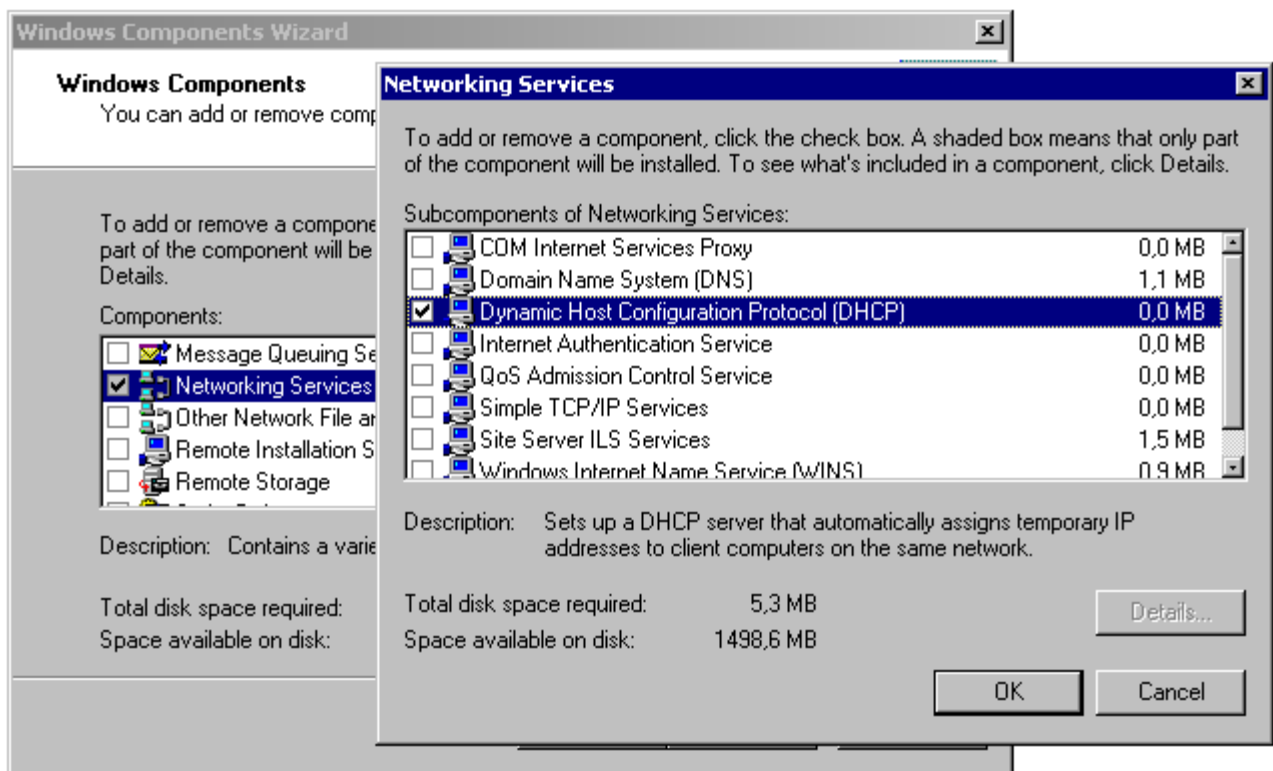
## Step 2. PXE folder creating.

Create a folder on the server to use for hosting a boot image file, for example **C:\tftpboot**. We call this folder as PXE folder.

## Step 3. Installing DHCP and TFTP

- **DHCP installing** ( skip this step if you have already Microsoft DHCP Server installed)

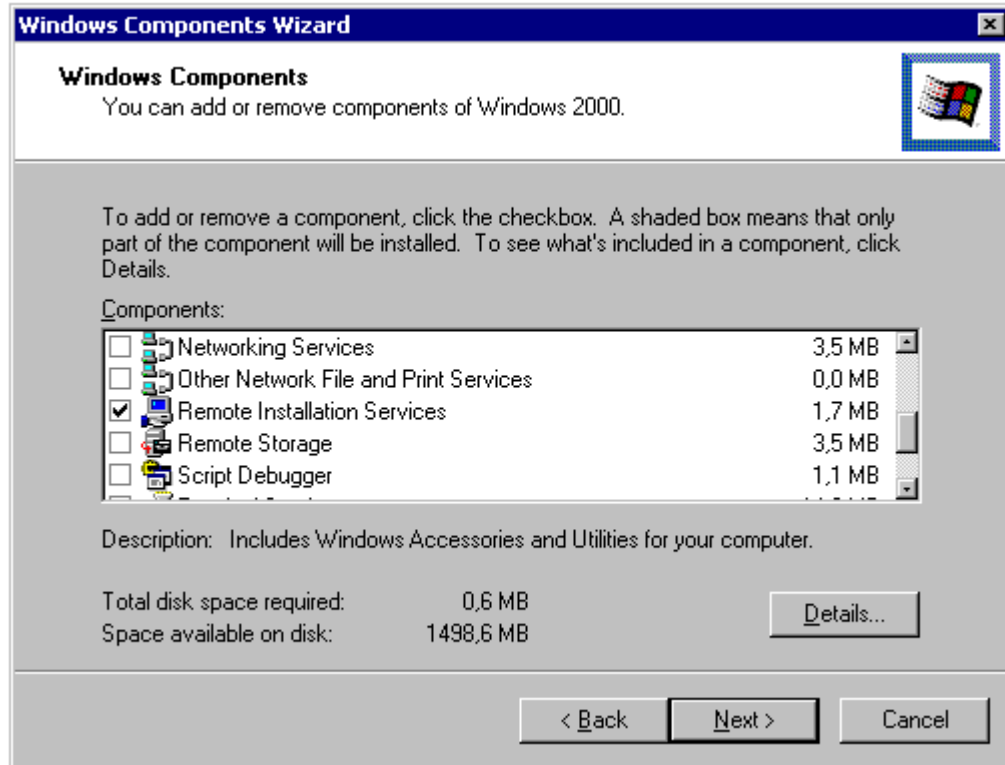
Go to **Start -> Settings -> Control panel -> Add/Remove Programs**, select **Add/Remove Windows components**. In the appeared dialog press **Components** if any, click on **Network Services**, in the appeared dialog check **Dynamic Host Configuration Protocol (DHCP)**, press **OK** and then **Next**. Follow the instructions on the screen to finish the installation. After installing restart the PC.



- **TFTP installing**

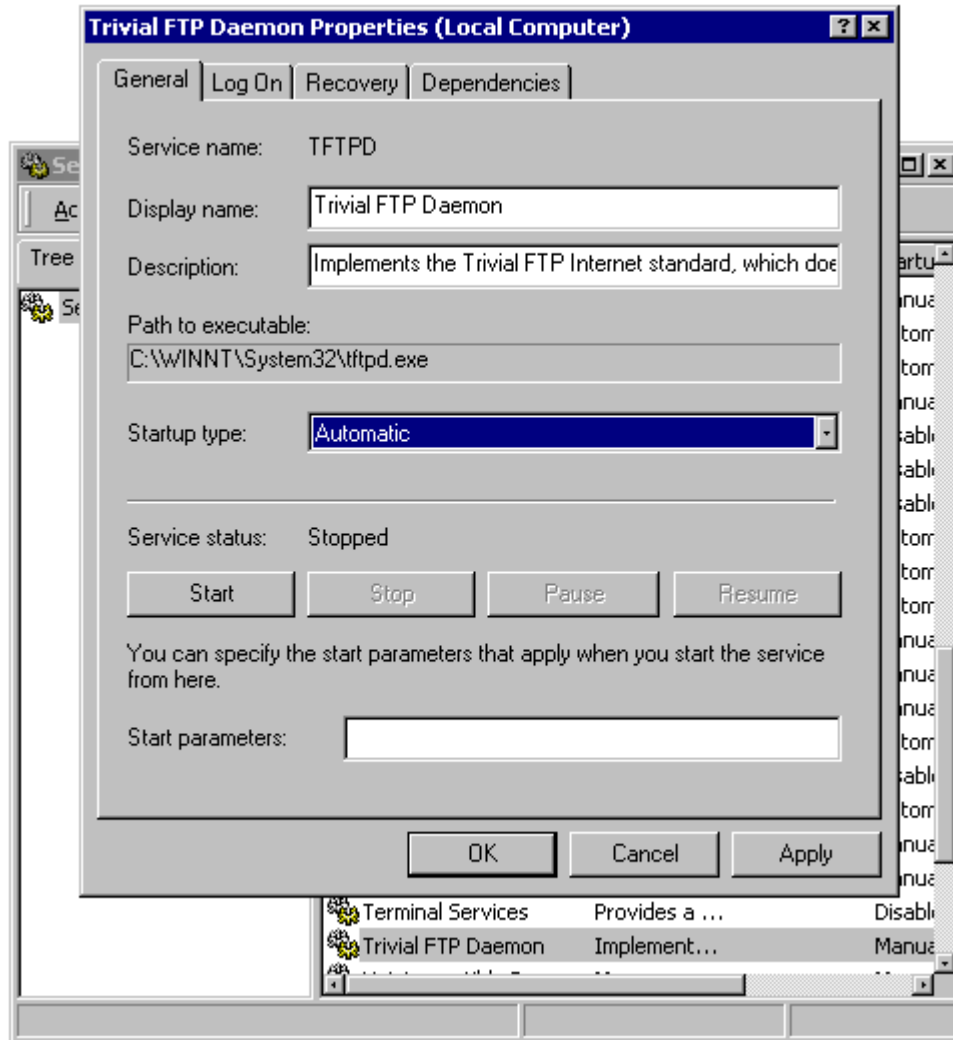
Go to **Start -> Settings -> Control panel -> Add/Remove Programs**, select **Add/Remove Windows components**. In the appeared dialog press **Components** if any, check **Remote Installation**

**Services** – TFTP Server is a part of RIS. Press **Next**. Follow the instructions on the screen to finish the installation. After installing restart the PC.



#### **Step 4. Setting up TFTP Server**

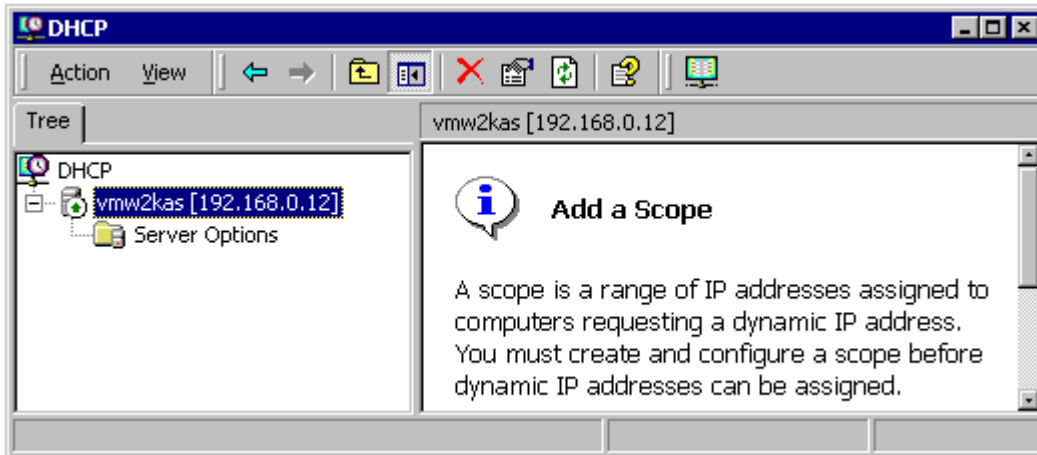
After restating your PC, go to **Start -> Programs -> Administrative Tools -> Services**. Double click on **Trivial FTP Daemon**. In the appeared dialog set **Startup type** to **Automatic**.



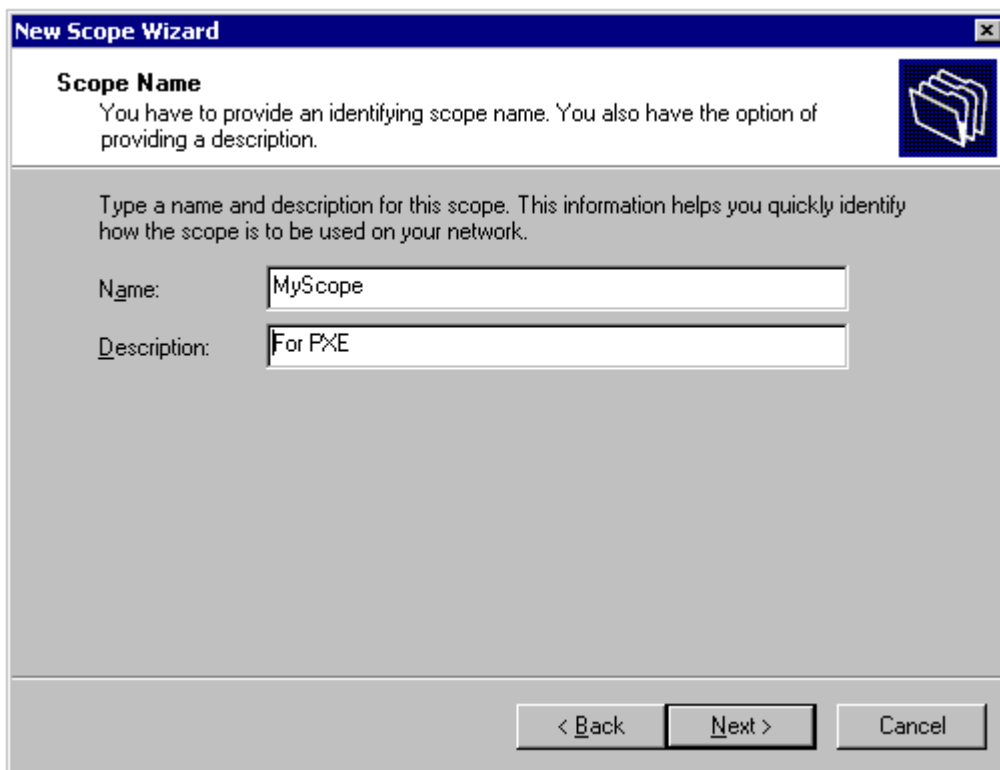
**Step 5. Setting up DHCP Server** (skip this step if you have already Microsoft DHCP Server configured)

Make sure that DHCP Server is started (**Administrative Tools -> Services**).

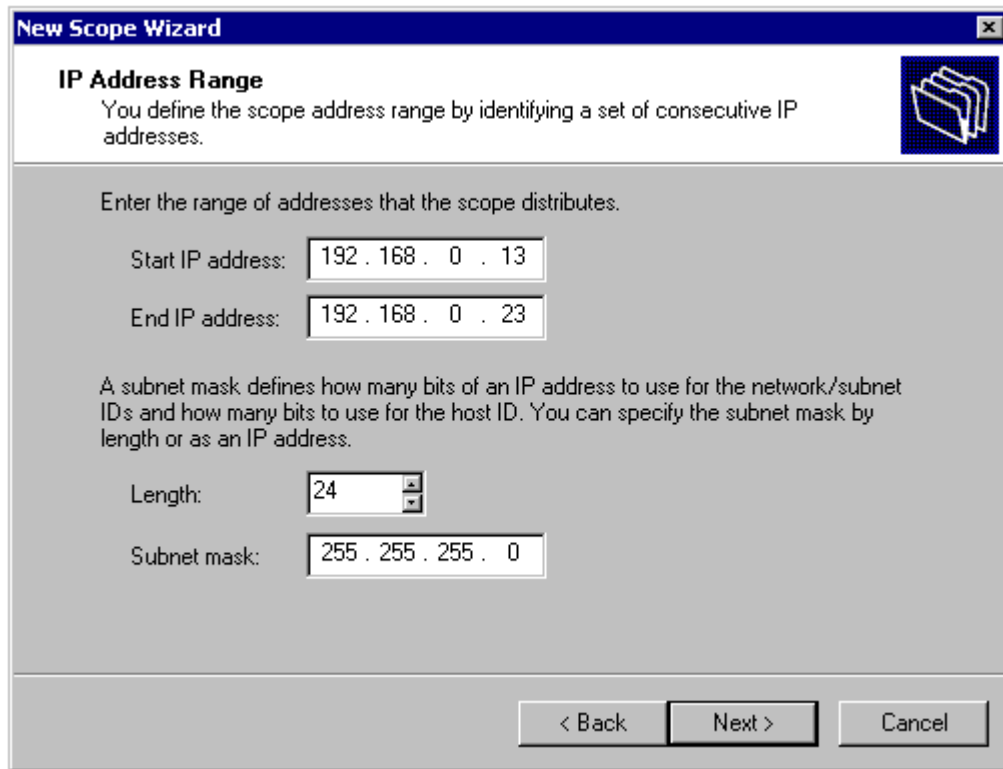
Go to **Start -> Programs -> Administrative Tools -> DHCP**



Right click on the icon with the green arrow and select **New Scope**, press **Next**. Name it as you wish.



Press **Next**. Enter **Start IP address**, **End IP address** and change **Subnet mask** if needed.



**New Scope Wizard**

**IP Address Range**  
You define the scope address range by identifying a set of consecutive IP addresses.

Enter the range of addresses that the scope distributes.

Start IP address: 192 . 168 . 0 . 13

End IP address: 192 . 168 . 0 . 23

A subnet mask defines how many bits of an IP address to use for the network/subnet IDs and how many bits to use for the host ID. You can specify the subnet mask by length or as an IP address.

Length: 24

Subnet mask: 255 . 255 . 255 . 0

< Back    Next >    Cancel

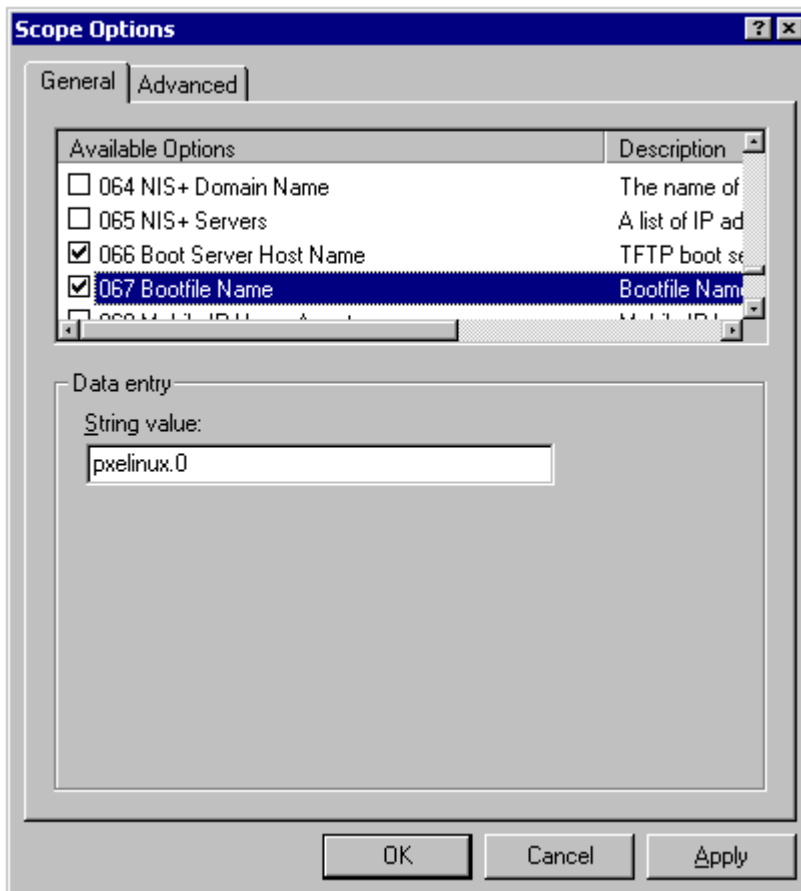
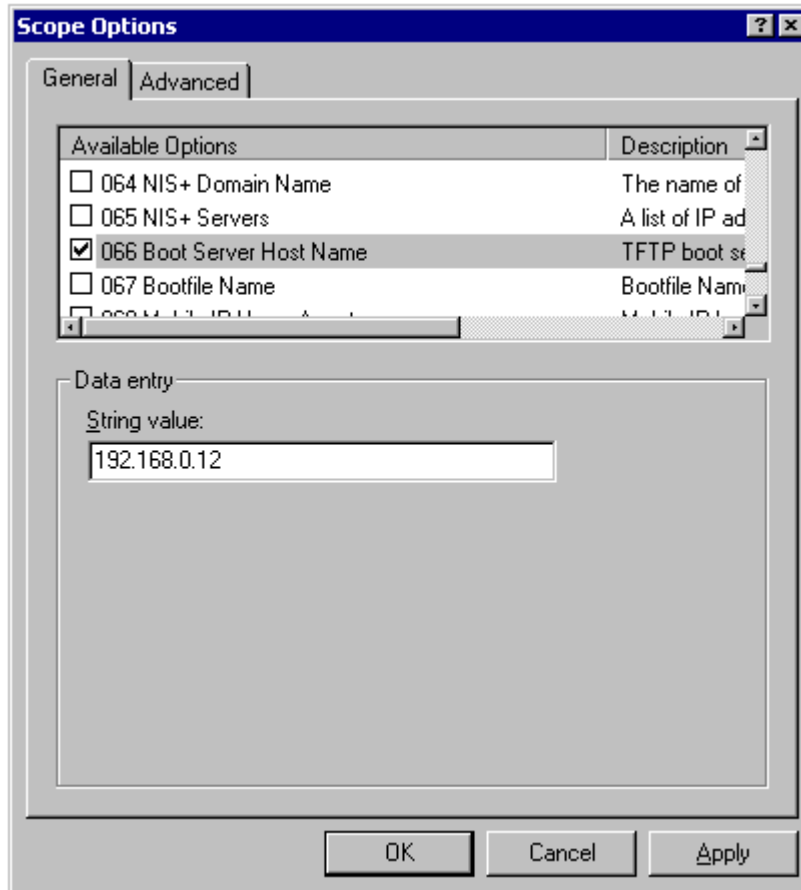
If you don't need to:

- exclude IP addresses from this scope;
- change the lease time;
- add an IP address for a router used by client;
- use Domain and DNS server;
- use WINS Servers;

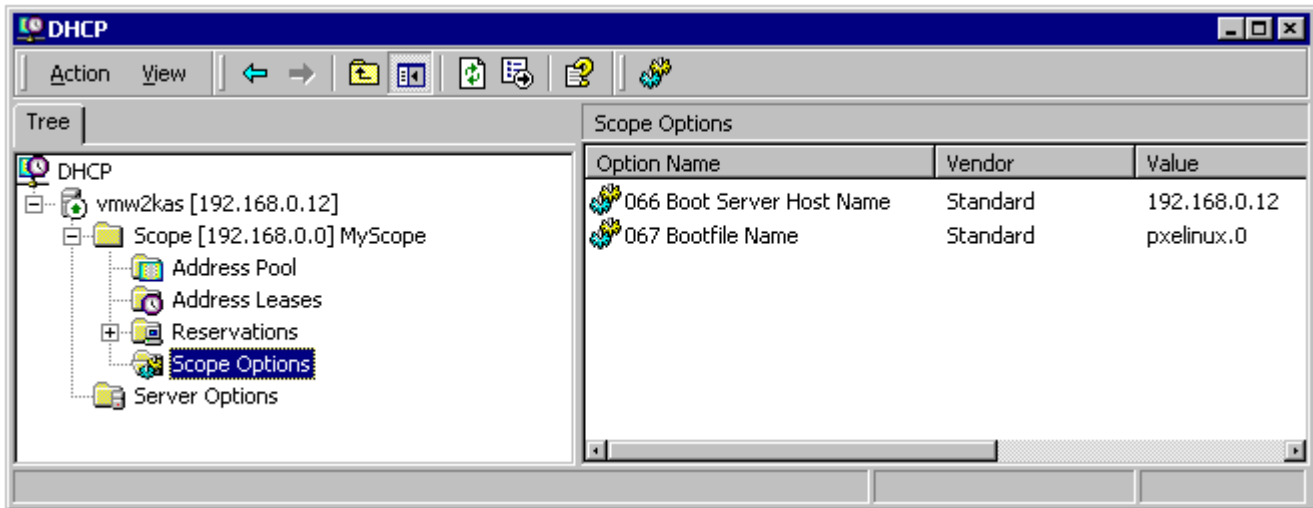
then press **Next** until you finish the configuration.

### Step 6. Additional settings for the DHCP Server

Select **Configure Options...** in the **Action** menu and check the **066 Boot Server Host Name** parameter. Below indicate IP address of your TFTP Server, in our case this IP address coincides with the IP of the current PC. After that check the **067 Bootfile Name** parameter and indicate **pxelinux.0** in the **String Value** box.



Make sure that this scope is activated.



### Step 7. Boot disk creating.

There are several ways to create a boot diskette:

**A.** This way is the most convenient. We created a special universal DOS boot image that works with any brand of NIC that includes a PXE-compliant boot ROM - you no longer need to create multiple image files to accommodate NIC specific packet drivers. Using this image file you can choose what folders in the LAN you want to share, what programs you want to start, for example Hard Disk Manager (just configure autoexec.bat inside this image), so you don't need to use **NET USE** command every time you start PXE client.

**B.** To create a Boot Recovery HDM or DB diskette without network support you should:

- Install **HDM** or **DB**;
- Open its program group in the **Start** menu;
- Select **Recovery Media Builder**
- Insert a diskette to the floppy drive
- Press **Next**
- Select **Floppy Disk** and press **Next**;
- Select **Typical** and press **Next**;
- Select the floppy drive and press **Next** twice to start.

**C.** To create a DOS boot disk with network support and then load HDM DOS through the LAN you may use, for example, the [Bart's Network Boot Disk](#) site. Go to this site, download and unzip the contents of [BFD full package v1.0.7](#) in the same directory as BCD (**D:\bcd**). You can easily add other

drivers (101 available) from the [list of network adapter plug-ins](#), just download the **.cab** files you need and copy them (**do not unpack**) into the **D:\bcd\cabs\drivers\ndis** directory. Open a command prompt (press **Start**, select **Run**, type **cmd** and press **Enter**) and browse to the BCD folder (**cd D:\bcd**). Run the command **bfd msnet** in order to create the bootable floppy.

Share a folder in Windows with HDM or DB DOS to start one of them and share a folder with all of the files you need for restoring your PC. Type **NET USE X: \\pxeservername\sharename** at the command prompt on your PXE client to get access to the shared folders, switch to the X: (**cd X:**) to use it.

### **Step 8. Syslinux files preparing.**

Open **Syslinux-3.07.zip** or higher with your favourite zip utility. We will need two files from the archive. Extract **pxelinux.0** from the root directory of the zip file into your PXE folder (**C:\tftpboot**), browse to the zip file's subfolder memdisk and extract the file called **memdisk** into your PXE folder.

### **Step 9. pxelinux.cfg creating.**

Make a subdirectory in your PXE folder called **pxelinux.cfg** (despite the **.cfg** this is a folder, not a file). In the **pxelinux.cfg** folder create a new text document and open it using Notepad. Insert the following lines:

```
default boot  
prompt 0  
say booting...  
label boot  
kernel memdisk  
append initrd=image.IMZ keeppxe
```

Save the text document and rename it as **default**. Do NOT use an extension like **.txt**. Windows hides the txt extension by default so if you may need to start **Windows Explorer**, click on **Tools --> Folder options**, select the **View** tab, and uncheck **Hide extensions for known file types**. Make sure the file is named **default** and not **default.txt**.

### Step 10. Boot Image creating.

If you used the **B** or **C** variant of the [step 7](#) to create a boot disk then install WinImage, and start the program. Click the **Disk** menu and make sure that the drive containing your newly created Boot diskette is selected. Next, from the disk menu select **Read Disk**. This will take a minute or two. If you need additional files on the Boot Disk you can add them using this program now. If you are happy with the default Boot Disk, select **File --> Save**. Set the file type to **Compressed Image file (\*.imz)**, name the image as **image** and save the image to your PXE folder.

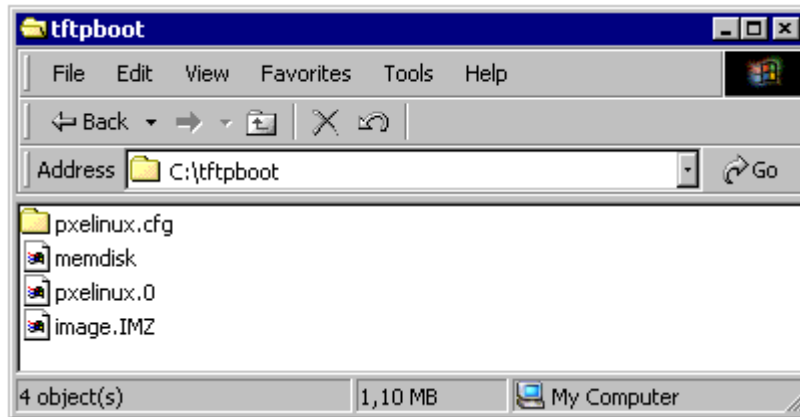
If you preferred the A variant then copy the **image.IMZ** file from PXE Network Deployment Package to the PXE folder.

PXE Network Deployment Package contains also two files, namely, **bzImage** and **initrd.gz** that you can use as the boot image to boot up your PXE client. Using these files you will be able to configure your network environment using Paragon NetConfig utility, load HDM for Linux and use many other things such as Boot Corrector, Simple Restore Wizard and Recording files to CD/DVDs utility. This variant requires no less than 256 MB RAM on your PXE client.

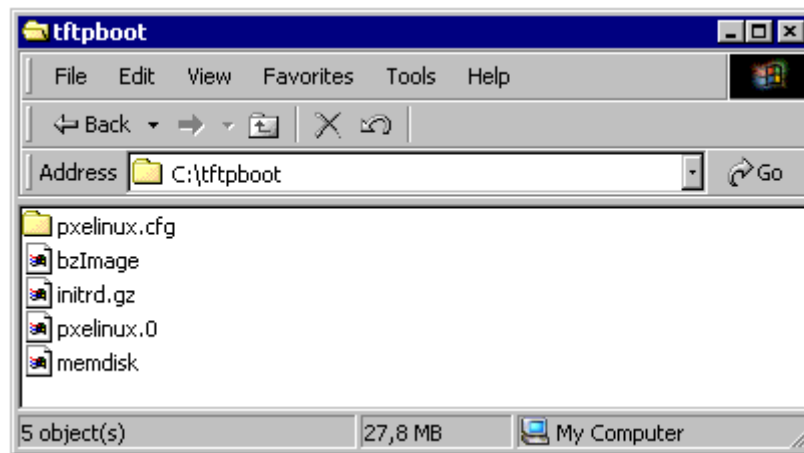
**Note:** If you are going to use **bzImage** and **initrd.gz** files to boot up your PXE client you should edit the **pxelinux.cfg/default** file in the following way:

```
label linux
kernel bzImage
append ip=auto
append initrd=initrd.gz ramdisk_size=256000 splash=silent quiet eng_ver vga=788 root=/dev/ram0
ipappend 1
```

**Note:** You can also find the **bzImage** and **initrd.gz** files in any Paragon Recovery CD but they will not contain the Paragon NetConfig utility and as a consequence you will not be able to configure your network environment. All other utilities and HDM for Linux will be available.

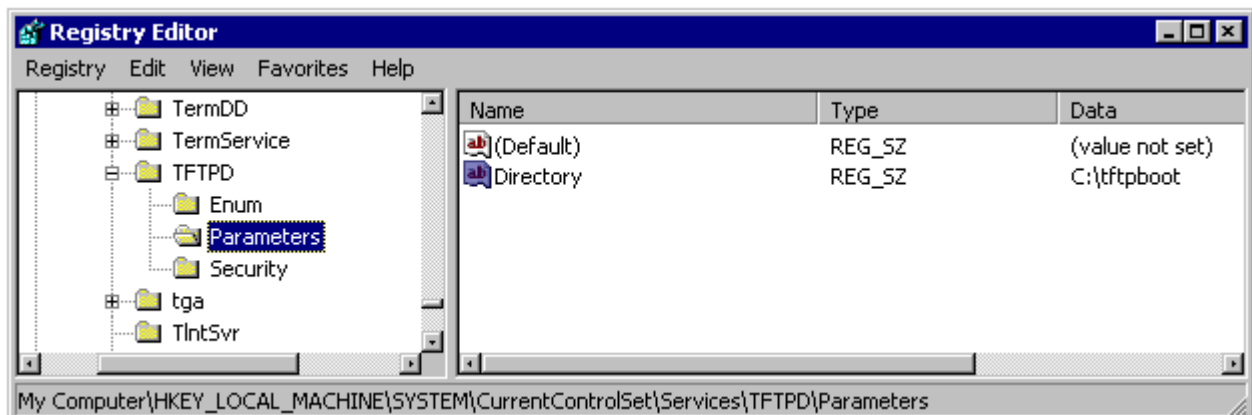


or (it depends on the boot image you are going to use)



### Step 11. Registry editing.

Go to **Start -> Run** and type **regedit**. Add the folder **\HKEY\_LOCAL\_MACHINE\System\CurrentControlSet\Services\TFTPDP\Parameters**. In the Parameters folder, add a String Value called **Directory**, with a value of the TFTP root directory path (**C:\tftboot**).



### **Step 12. Finishing.**

The server setup is DONE! Make sure the DHCP Server and TFTP Daemon are running on your PXE server and your PXE client is connected to it on a wired network. Restart your PXE client and press F12 right as it comes on. Select the second icon from the right for network booting. You can also go to the BIOS and set LAN as **First Boot Device** (if you don't want to press F12.). After restarting the PXE client you will be able to restore this PC using HDM or DB.

## **3.2. Workshop №2 – using third-party software under Windows OS.**

This workshop provides step-by-step instructions on setting up the PXE environment using Windows operating system as the PXE server (DHCP and TFTP servers) and DOS or Linux as clients with and without network support.

### **3.2.1 What we need to carry out this workshop:**

6. PC with a PXE-enabled NIC.
7. Second computer running Windows to act as the PXE server.
8. Wired network connecting the PC to the PXE server.
9. DHCP and TFTP Server software for Windows (TFTPD32 utility by Philippe Jounin works well for this). It can be downloaded from its [home page](#).

Tftpd32 includes DHCP, TFTP, SNTP and Syslog servers as well as a TFTP client. The TFTP client and server are fully compatible with TFTP option support (tsize, blocksize and timeout), which allow the maximum performance when transferring the data. Some extended features such as directory facility, security tuning, interface filtering; progress bars and early acknowledgments enhance usefulness and throughput of the TFTP protocol for both client and server. The included DHCP server provides unlimited automatic or static IP address assignment.

10. DOS boot disk with or without network support. You can also download Paragon PXE Network Deployment Package where you can find universal DOS boot image file with network support for all PXE-enabled NIC. The Paragon PXE Network Deployment Package contains also Paragon Recovery PXE files to boot Paragon Recovery CD with network support (Linux part only).

11. Syslinux-3.07.zip or higher from [kernel.org](http://kernel.org).

12. WinImage software for creating the boot disk image. Download winimaXX.exe from [WinImage](#)

### **3.2.2 Step by step instruction:**

#### **Step 1. Downloading.**

Begin by downloading the necessary software and files into a temporary folder on the computer that will act as the PXE Server.

#### **Step 2. PXE folder creating.**

Create a folder on the server to use for hosting a boot image file, for example **C:\tftpboot**. We call this folder as PXE folder.

**Step 3. Boot disk creating.** Carry out this step as the [step 7](#) (Boot disk creating) in the [Workshop №1](#).

**Step 4. Syslinux files preparing.** Carry out this step as the [step 8](#) (Syslinux files preparing) in the [Workshop №1](#).

**Step 5. pxelinux.cfg creating.** Carry out this step as the [step 9](#) (pxelinux.cfg creating) in the [Workshop №1](#).

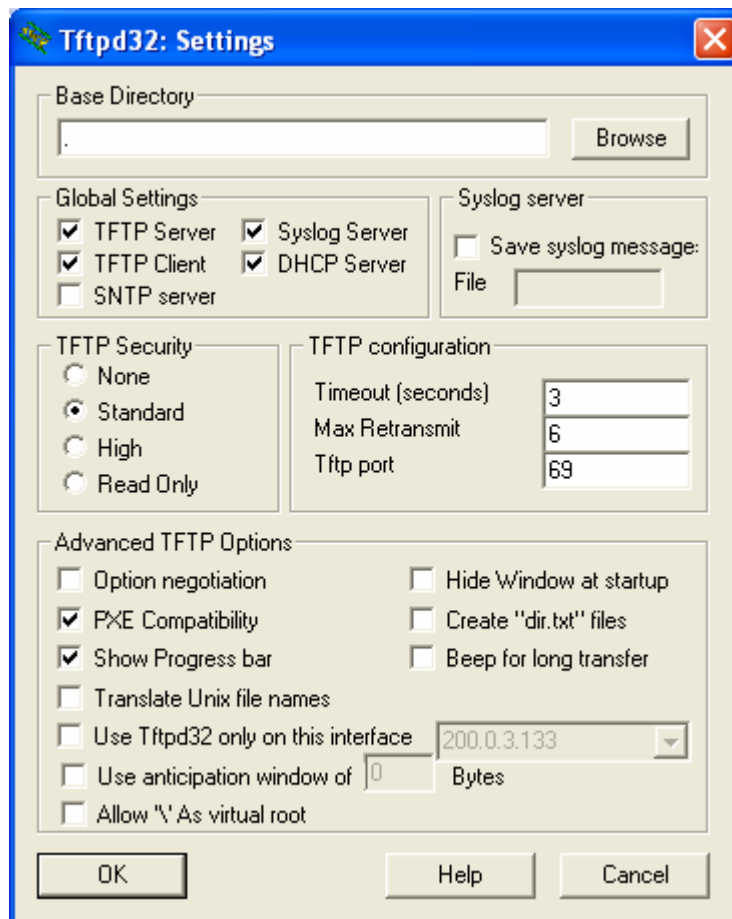
**Step 6. Boot Image creating.** Carry out this step as the [step 10](#) (Boot Image creating) in the [Workshop №1](#).

#### **Step 7. Installing Tftpd32.**

Go to the temporary folder, open **tftpd32.xxx.zip** and extract **tftpd32.exe** into the **C:\tftpboot** folder.

### Step 8. Starting up Tftpd32.

Browse to the **C:\tftpbboot** folder and start **tftpd32.exe**. The current directory should come up as your PXE server folder, but if it does not, change it to that folder. Click on the **Settings** button. The base directory should be set to **'.'** Make sure that all of the boxes in the **Global Settings** are checked except for **SNTP server**. Check **PXE Compatibility**. Leave all of the other settings at their defaults and click **OK**.

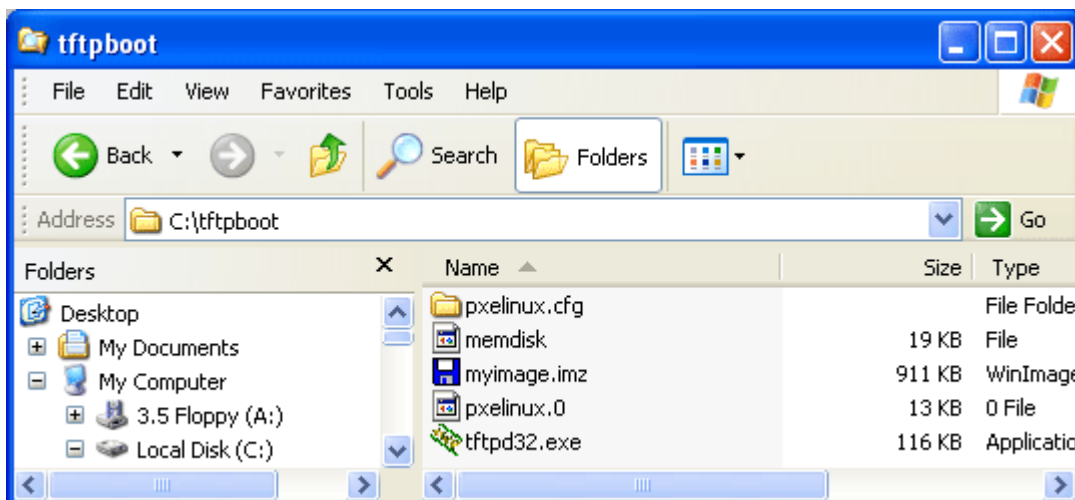
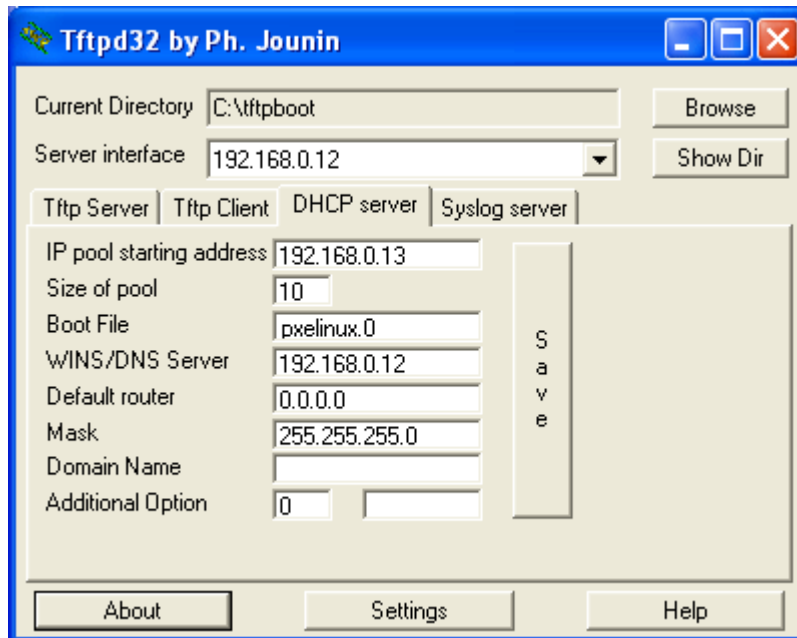


### Step 9. Setting up DHCP.

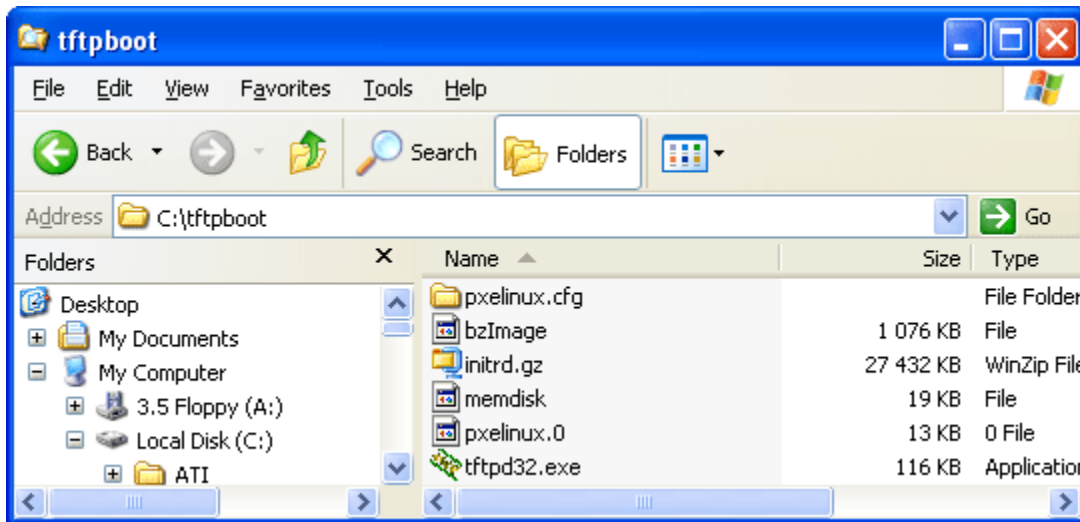
Click on the **DHCP** tab. PXE booting requires a special DHCP server that supplies the booting computer with the information it needs to get started and find the necessary files (on your PXE server). If you have a router or some other device on your network that is already acting as a DHCP server,

you might consider disabling DHCP on that device or you should set special options for this DHCP server yourself so that it may indicate TFTP Server and bootstrap program to the PXE client.

Set the **IP Pool Starting address** to, for example, 192.168.0.13. The **Size of pool** option specifies how many IP addresses the DHCP server has available to assign. Set, for example, to 10. Set the **Boot file** to **pxelinux.0**. The next four fields in this tab depend on your network. If you know the **WINS/DNS Server** IP address and the router address, you can fill them in. Otherwise leave them blank or you can also use the current IP address of the PXE Server as WINS/DNS one. Set the **Mask** to 255.255.255.0 unless you know it to be otherwise. Click the **Save** button along the side of the tab. This program keeps its DHCP and TFTP servers running as long as it is open.



or (it depends on the boot image you are going to use)



### Step 10. Finishing.

The server setup is DONE! Make sure the Tftpd32 is running on your PXE server and your PXE client is connected to it on a wired network. Restart your PXE client and press F12 right as it comes on. Select the second icon from the right for network booting. You can also go to the BIOS and set **LAN as First Boot Device** (if you don't want to press F12.). After restarting the PXE client you will be able to restore this PC using HDM or DB.

**Note:** If you didn't fill the **WINS/DNS Server** and **Default router** fields then computers that will assign a new IP address from this DHCP Server will probably not be able to access the internet. For this reason, you should only keep the Tftpd32 program running long enough to get your PXE client booted. This program isn't necessary for accessing shared folders on your PXE server after you have booted your PXE client.

### 3.3. Workshop №3 – similar to the workshop №2 but using alternative third-party software.

This workshop is the same as the [Workshop №2](#) but will describe another possible way for creating a **Boot Image File**.

#### 3.3.1 What we need to carry out this workshop:

1. PC with a PXE-enabled NIC.
2. Second computer running Windows to act as the PXE server.

3. Wired network connecting the PC to the PXE server.
4. DHCP and TFTP Server software for Windows (TFTPD32 utility by Philippe Jounin works well for this). It can be downloaded from its [home page](#).
5. DOS boot disk with or without network support. You can also download Paragon PXE Network Deployment Package where you can find universal DOS boot image file with network support for all PXE-enabled NIC. The Paragon PXE Network Deployment Package contains also Paragon Recovery PXE files to boot Paragon Recovery CD with network support (Linux part only).
6. [3Com Managed Boot Agent \(MBA\)](#) package from 3Com. This package will help to create a DOS boot disk. It can be downloaded [here](#).

### **3.3.2 Step by step instruction:**

#### **Step 1. Downloading.**

Begin by downloading the necessary software into a temporary folder on the computer that will act as the PXE Server.

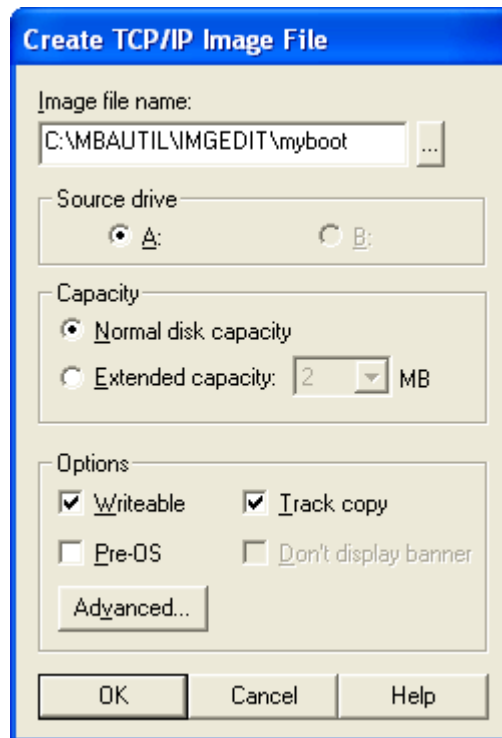
#### **Step 2. PXE folder creating.**

Create a folder on the server to use for hosting the necessary files, for example **C:\tftpboot**.

**Step 3. Boot disk creating.** Carry out this step as the [step 7](#) (Boot disk creating) in the [Workshop №1](#). Unfortunately this workshop cannot use the A variant for creating the boot image.

#### **Step 4. Boot Image creating.**

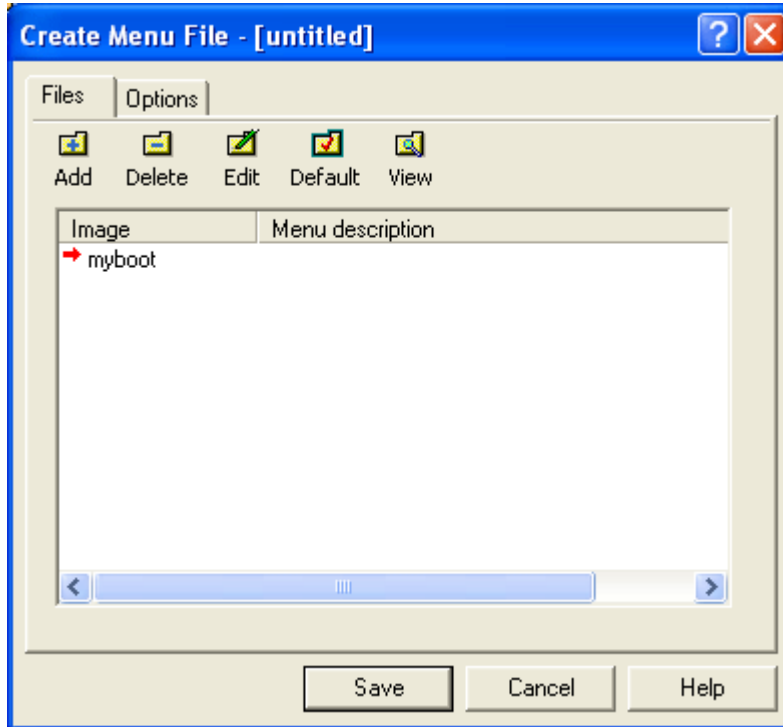
Go to the temporary folder. Double click on **3Com Managed Boot Agent (MBA)**, it usually has the **util430.exe** name. Extract the files to any folder on the PXE server, for example **C:\MBAUTIL**. Insert the boot disk to the floppy drive. Go to the **C:\MBAUTIL\IMGEDIT\** folder and double click on the **IMGEDIT.EXE** file. Press the **Next** button twice, uncheck **Enable file association** and press **Finish**. In the appeared dialog press the **Create a TCP/IP or PXE image...** You will see the following dialog:



Name the file as you wish (for example, **myboot**) and use the same setting as on the figure above, press the **Advanced...** button and select **Keep UNDI in memory**. Press **OK**. This will take a minute or two.

#### **Step 5. Boot menu creating.**

Once the image disk ready press **Create a PXE menu boot file...** In the appeared dialog press **Add**, select the file that you created on the previous step (**myboot**). Press **OK**.



Press **Save**. Name the file as **mba.pxe** (as by default) and press **Save** once again. Press **Exit** to close **3Com Managed Boot Agent (MBA)**.

#### **Step 6. Boot menu and Boot Image copying.**

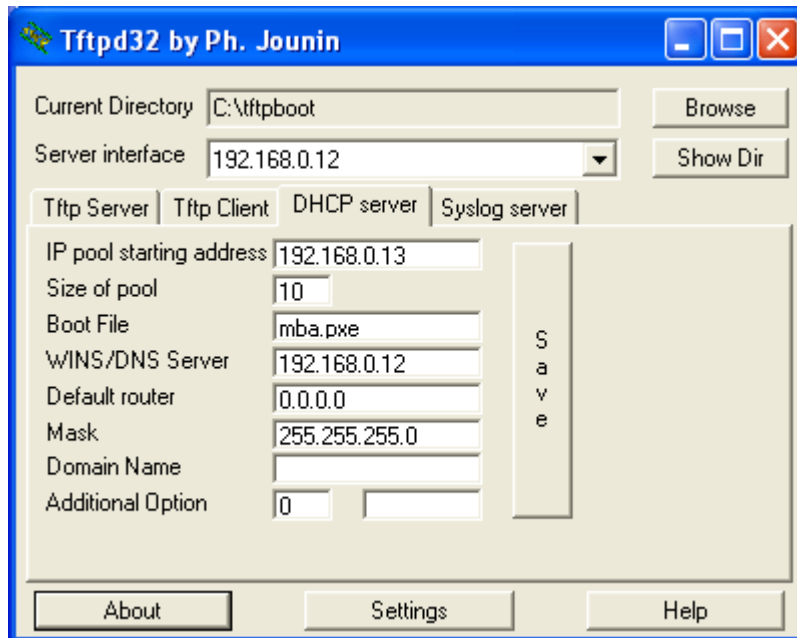
Copy the **mba.pxe** and **myboot** files to the **C:\tftboot** folder.

#### **Step 7. Installing Tftpd32.**

Go to the temporary folder, open **tftpd32.xxx.zip** and extract **tftpd32.exe** into the **C:\tftboot** folder.

**Step 8. Starting up Tftpd32.** Carry out this step as the [step 8](#) (Starting up Tftpd32) in the [Workshop №2](#).

**Step 9. Setting up DHCP.** Carry out this step as the [step 9](#) (Setting up DHCP) in the [Workshop №2](#) but use **mba.pxe** as **Boot File** rather than **pxelinux.0**.



### Step 10. Finishing.

The server setup is DONE! Make sure the Tftpd32 is running on your PXE server and your PXE client is connected to it on a wired network. Restart your PXE client and press F12 right as it comes on. Select the second icon from the right for network booting. You can also go to the BIOS and set **LAN as First Boot Device** (if you don't want to press F12.). After restarting the PXE client you will be able to restore this PC using HDM or DB.

**Note:** If you didn't fill the **WINS/DNS Server** and **Default router** fields then computers that will assign a new IP address from this DHCP Server will probably not be able to access the internet. For this reason, you should only keep the Tftpd32 program running long enough to get your PXE client booted. This program isn't necessary for accessing shared folders on your PXE server after you have booted your PXE client.

### 3.4. Workshop №4 – using Linux software only.

This workshop provides step-by-step instructions on setting up the PXE environment using Linux operating system as the PXE server (DHCP and TFTP servers) and Linux or DOS as clients with and without network support. It is assumed that the reader has a fundamental grasp of networking concepts. DHCP, Linux Kernel and TFTP will be covered here, as well, and some knowledge of those

topics is required. Below are the steps to perform PXE configuration using Mandrake 10.2 as the server OS and Paragon Recovery CD, namely its Linux part as the client OS.

### 3.4.1 What we need to carry out this workshop:

1. PC with a PXE-enabled NIC.
2. Second computer running Linux to act as the PXE server.
3. Wired network connecting the PC to the PXE server.
4. DHCP Server software for Linux. You can download it [here](#).
5. TFTP Server software for Linux. You can download it [here](#).
6. Syslinux package. You can download it [here](#).
7. PXE Network Deployment Package, namely, **bzImage** and **initrd.gz** files or any Paragon Recovery CD **iso** image to extract these files.

### 3.4.2 Step by step instruction:

#### Step 1. Downloading.

Begin by downloading the necessary files into a temporary folder on the computer that will act as the PXE Server (**mkdir /store**).

#### Step 2. PXE folder creating.

Create a folder on the server to use for hosting the necessary files, for example “/tftpboot”.

A note about files in /tftpboot: They should all have permissions that allow global reading of the files.

#### Step 3. Installing and setting up DHCP Server.

1. Use the ISC DHCP server (v. 2.0 or 3.0), as it is highly recommended, is easy to configure. You can download it, for example, [here](#).
2. Install the package for DHCP Server:  
**rpm -i /store/dhcp-3.02rc3-2.i386.rpm**
3. Create configure file **/etc/dhcpd.conf**.  
**/etc/dhcp.conf** is the main file on my machine for configuring the DHCP server. A very generic **dhcp.conf** would look something like this:

```
allow booting;
allow bootp;
```

```
# Standard configuration directives...
```

```
option domain-name [ domain_name ];
option subnet-mask [ subnet_mask ];
option broadcast-address [ broadcast_address ];
option domain-name-servers [ dns_servers ];
option routers [ default_router ];
```

```
# Group the PXE bootable hosts together
```

```
group {
    # PXE-specific configuration directives...
    next-server [ TFTP_server_address ];
    filename "/tftpboot/pxelinux.0";

    # You need an entry like this for every host
    # unless you're using dynamic addresses
    host hostname {
        hardware ethernet [MAC address, in format nn:nn:nn:nn:nn:nn];
        fixed-address [ip address that will be assigned to client];
    }
}
```

All the values in square brackets should be filled in with the correct values for your setup. The man page for **dhcp.conf** is very good. If you get stuck on this step, try consulting it. The really important parts of the example are those in the **group** block. These are the parameters specific to TFTP booting clients. Mainly, the configuration specifies what tftp server has boot images (next-server), what file they should try to get from the next server (filename) and then there is a section defining every host on the network that should be TFTP booted. That section tells the DHCP server to send the host with the specified hardware address the fixed-address and all other fields in the group section.

Bellow is **dhcp.conf** that we used for this workshop:

```
allow booting;
allow bootp;
```

```
option subnet-mask 255.255.255.0;  
  
option broadcast-address 192.168.0.255;
```

```
default-lease-time-1;  
use-host-decl-names on;  
ddns-update-style ad-hoc;  
filename "pxelinux.0";
```

```
subnet 192.168.0.0 netmask 255.255.255.0 {  
range 192.168.0.13 192.168.0.20;  
}
```

**Note:** We didn't create the group section because we will boot one PC and use one boot program only, TFTP Server will be installed together with DHCP Server on the same PXE Server.

#### Step 4. Installing TFTP Server.

1. You can download TFTP Server, for example, [here](#).
2. Install the package for TFTP Server:

```
rpm -i /store/tftp-server-0.40-1.i386.rpm
```

**Note:** TFTP Server must support the "tsize" TFTP option (RFC 1784/RFC 2349). To check that your TFTP Server supports the "tsize" TFTP option type **man in.tftpd**. The appeared "man" page must contain any information about the "tsize" TFTP option.

#### Step 5. Syslinux files preparing.

1. You can download Syslinux package, for example, [here](#).
2. Install the package:

```
rpm -i /store/syslinux-3.09-1.i386.rpm
```

3. After installing the package, copy the **pxelinux.0** file into your tftpboot area (**/tftpboot**). As in the following example:

```
cp /usr/lib/syslinux/pxelinux.0 /tftpboot
```

**Note:** It is not necessary installing the syslinux package, you can just copy a compressed tar archive (for example, **syslinux-3.09.tar.bz2**) to your PC, extract it and copy **pxelinux.0** to your tftpboot area (**/tftpboot**).

```
bzip2 -d syslinux-3.09.tar.bz2
```

```
tar -xvf syslinux-3.09.tar
```

You can download a compressed tar archive of syslinux [here](#).

4. Go to the **/tftpboot** directory. Make a directory called **pxelinux.cfg**.

```
$ cd /tftpboot
```

```
$ ls
```

```
pxelinux.0
```

```
$ mkdir pxelinux.cfg
```

Using the editor of your choice, create a file in the **/tftpboot/pxelinux.cfg** directory called **default**. It should look like this:

```
label linux
```

```
kernel bzImage
```

```
append ip=auto
```

```
append initrd=initrd.gz ramdisk_size=256000 splash=silent quite eng_ver vga=788 root=/dev/ram0
```

```
ipappend 1
```

LABEL identifies the name of the boot image. KERNEL indicates the first file to get, which in most cases is the Linux kernel. APPEND appends kernel parameters. IPAPPEND, if greater than zero, tells PXELINUX to generate and append the IP information to the boot request. In almost every case, you want this.

If you use **bzImage** and **initrd.gz** files from PXE Network Deployment Package then you will be able to configure your network environment. If you use the same files from Paragon Recovery CD then you will not be able to configure the network.

This is about the most generic, unspecific way of setting up the PXELINUX bootloader. There are a number of other options, including setting up specific config files for specific clients, including boot messages, and loading a default image (without showing the "boot:" menu). The pxelinux.doc and syslinux.doc documents explain these other options.

**Note:** If you want to load DOS on the PXE client rather than Linux, carry out steps [7](#) (Boot Disk creating), [8](#) (Syslinux files preparing), [9](#) (pxelinux.cfg creating) and [10](#) (Boot Image creating) of the [Workshop №1](#). Don't forget to copy the **memdisk** and boot disk image file to the PXE folder (/tftpboot).

### Step 6. Starting up all services.

Once the configuration of DHCPD and TFTP is done, start the dhcpd and tftpd servers:

```
# as root ...
```

```
$ in.tftpd -v -l -s /tftpboot
```

The '-v' option will increase verbosity. It's useful for testing. If things go wrong, you can use '-vv' to increase debugging output. Messages are logged to syslog. '-l' sets the server to run in standalone. If you are running the server in inetd, don't use the '-l' option. '-s /tftpboot' makes the tftpd server use /tftpboot as it's root directory.

The main services that must be started and accessible for the PXE booting process are:

- dhcpd
- tftpd
- xinetd (for tftpd)

The way to check is to use **chkconfig**.

```
$ /sbin/chkconfig -- list tftp
```

```
tftp          off
```

```
$ /sbin/chkconfig tftp on
```

```
$ /sbin/chkconfig -- list xinetd
```

```
xinetd      0:off 1:off 2:off 3:on 4:on 5:on 6:off
```

```
$ /sbin/chkconfig -- list dhcpd
```

```
dhcpd       0:off 1:off 2:off 3:off 4:off 5:off 6:off
```

```
$ /sbin/chkconfig dhcpd on
```

```
dhcpd       0:off 1:off 2:on 3:on 4:on 5:on 6:off
```

In the above example, tftp and dhcpd was not configured to start by default. However, xinetd was. To configure tftp and dhcps to start automatically, **chkconfig <service> on** was run. Since tftp is a part of xinetd, as long as xinetd is running, nothing past the **chkconfig tftp on** needs to be done to use the tftp service.

The below example illustrates confirming that xinetd is running:

```
# /sbin/service xinetd status  
xinetd (pid 5157) is running ...
```

However, if you type:

```
# /sbin/service dhcpd status
```

you will see:

```
dhcpd is stopped.
```

To start dhcpd service type:

```
# /sbin/service dhcpd start
```

if your network settings are correct (**ifconfig**) you should see:

```
Starting dhcpd: [ OK ]
```

### Step 7. Finishing.

The server setup is DONE! Make sure the DHCP Server and TFTP Daemon are running on your PXE server and your PXE client is connected to it on a wired network. Restart your PXE client and press F12 right as it comes on. Select the second icon from the right for network booting. You can also go to the BIOS and set LAN as **First Boot Device** (if you don't want to press F12.). After restarting the PXE client you will be able to restore this PC using Paragon Recovery CD, HDM or DB.