# HOWTO -- Documentation

**The documentation is simple and straight forward. Just start with the Introduction and follow the install steps.**

# Introduction

The following installation steps will let you to backup and restore whole systems through your network or removable media (e.g. CDs, DVDs, and USB mass storage devices).

At the end of the installation, you should be able to boot any computer on a CD/DVD or your network and press the F12 key during the boot process to launch the PING software (PXE). Using this tool, you should be able either to backup, either to restore systems and partitions. Think of it as a Ghost-like process.

# Prerequisites

**Using PING on a Directly Attached Hard Drive:**

- **Hard drive space.** To store a full 40 GB hard disk, you may need up to 40 GB.

**Using PING Over a Network:**

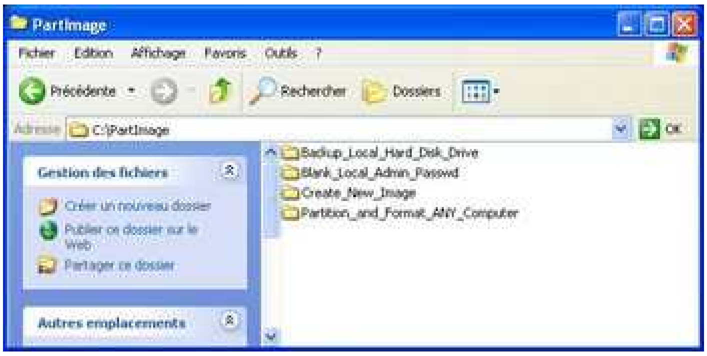
- **A LAN (Local Area Network)**. At least two connected computers. One of these computers is to be copied or restored, and the other is supposed to store the images you'll do with PING.
- **A Share**. The computer used to store the images must be able to share files (any Microsoft Windows computer can). If you are a Linux user, install Samba.
- **Space**. To store a full 40 GB hard disk, you may need up to 40 GB.

# Configuring a PC to serve as a host for your images

*SKIP this step if all you want is a locally-stored backup of your system.*

*Directory names are given here as samples. Adapt the following according to your specific needs and configuration.*

1. Download **PartImage.zip** and save it on C:\
2. Open C:\ and uncompress the archive (**choose: extract here**)
3. You should have a **C:\PartImage** directory, containing several subfolders (eg.: a Create_New_Image subfolder)



C:\PartImage

Backup_Local_Hard_Disk_Drive
Blank_Local_Admin_Passwd
Create_New_Image
Partition_and_Format_ANY_Computer

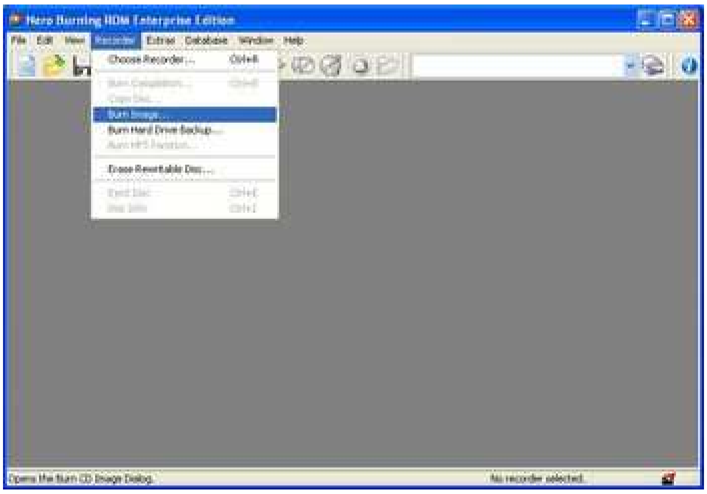
4. Right-click on the **C:\PartImage** folder and **share it**. Set the permissions as suits you. Just keep in mind that you'll have to be able to write on this shared folder from any host on your network. Create a user if necessary.

# Downloading and Burning the PING ISO

*SKIP this step if you only want to setup a PXE server.*

1. **Download PING.2.01.iso**

2. **Burn it** using your favorite burning software
   - **With Nero:**
     1. Open Nero Burning Rom
     2. Select Recorder / Burn Image



     3. **Browse** to PING.2.01.iso

   - **If you have no tool to burn an ISO:**
     1. Go to Alex Feinman's website and download what **IsoRecorder** version suits your configuration. It's freeware. **Install it.**
     2. Then, open the folder containing the **PING.2.01.iso** file, right-click on the ISO, and choose "**Copy Image to CD**"

# Installing your PXE Server to allow network hosts to boot PING (Optional step)

*SKIP this step if you only want a backup of your system stored locally or on removable media.*

*Directory names are given here as samples. Adapt the following according to your specific needs and configuration.*

- **If you've got a working RIS Server (like MS Windows Server 2000 or 2003) :**

1. You probably have a **\\server_name\RemInst\Setup\English\Images\** directory, used to share images for the RIS server.

2. **Create the following folders:**
   - \\server_name\RemInst\Setup\English\Images\**PING**\
   - \\server_name\RemInst\Setup\English\Images\PING\**i386**\
   - \\server_name\RemInst\Setup\English\Images\PING\i386\**templates**\
   - \\server_name\RemInst\Setup\English\Images\PING\i386\templates\**pxelinux.cfg**\ *(yes, this is a folder, not a file)*

3. In the **\\server_name\RemInst\...\templates** directory, download the following files:
   - **kernel**
   - **initrd.gz**
   - **pxelinux.0**

4. In the **\\server_name\RemInst\...\templates** directory, create a text file that you'll name **ping.sif**, containing:

```
[OSChooser]
Description = "PING 2.01 (Partimage Is Not Ghost)"
Help = ""
LaunchFile = "Setup\English\Images\PING\i386\templates\pxelinux.0"
ImageType = Flat
Version="2.01"
```

5. In the **\\server_name\RemInst\...\templates\pxelinux.cfg\** directory, create a text file that you'll name **default** (**not** default.txt), containing:

```
DEFAULT rescue
PROMPT 0
LABEL rescue
KERNEL kernel
APPEND vga=normal devfs=nomount pxe ramdisk_size=33000 load_ramdisk=1
init=/linuxrc prompt_ramdisk=0 initrd=initrd.gz root=/dev/ram0 rw
noapic nolapic lba combined_mode=libata ide0=noprobe nomce pci=nommconf
pci=nomsi irqpoll
```

- **If you have no working RIS Server and want to build your own easily :**

1. Install **TFTPD32** from Ph. Jumin's web site

2. On your future PXE / RIS Server, **create the following directories**:
    - D:\**PXE**\\
    - D:\\PXE\**i386**\\
    - D:\\PXE\\i386\**templates**\\
    - D:\\PXE\\i386\\templates\**pxelinux.cfg**\\

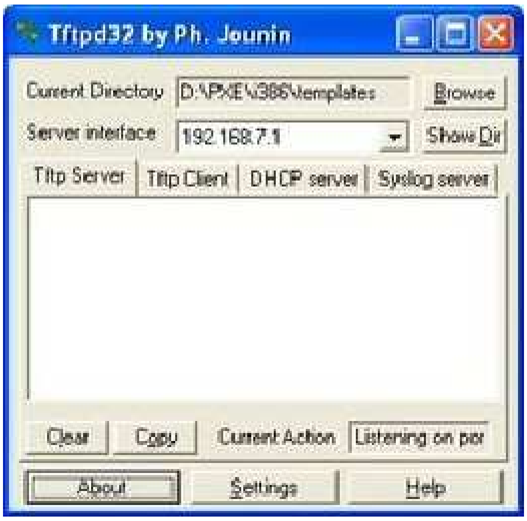
3. In the **D:\PXE\i386\templates\** directory, download the following files:
    - kernel
    - initrd.gz
    - pxelinux.0

4. In the **D:\PXE\i386\templates\pxelinux.cfg\** directory, create a text file that you'll name **default** (not default.txt), containing:

```
DEFAULT rescue
PROMPT 0
LABEL rescue
KERNEL kernel
```

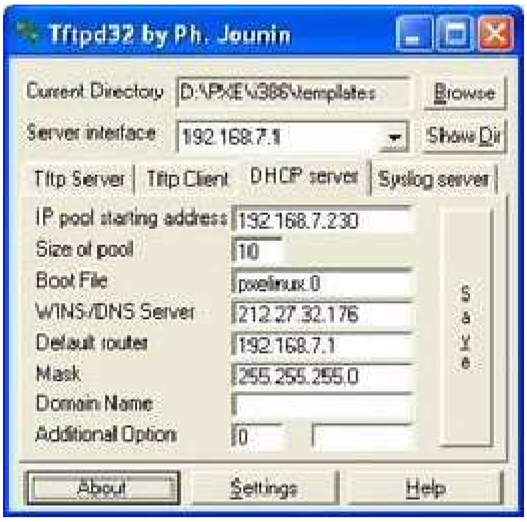
```
APPEND vga=normal devfs=nomount pxe ramdisk_size=33000 load_ramdisk=1
init=/linuxrc prompt_ramdisk=0 initrd=initrd.gz root=/dev/ram0 rw
noapic nolapic lba combined_mode=libata ide0=noprobe nomce pci=nommconf
pci=nomsi irqpoll
```

5. **Launch TFTPD** and configure it the following way:



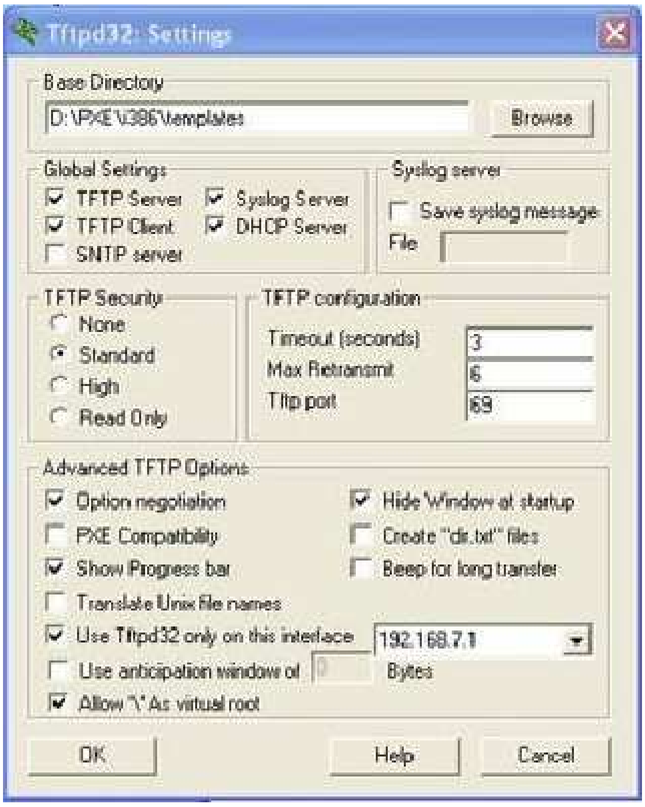
Current Directory: D:\PXE\i386\templates
Server interface: 192.168.7.1
Current Action: Listening on por

If you've got several network cards in your computer, you'll have to choose in the **server interface** area the one that communicates with your other LAN hosts.



Current Directory: D:\PXE\i386\templates
Server interface: 192.168.7.1
IP pool starting address: 192.168.7.230
Size of pool: 10
Boot File: pxelinux.0
WINS/DNS Server: 212.27.32.176
Default router: 192.168.7.1
Mask: 255.255.255.0
Domain Name:
Additional Option: 0

Configure your **DHCP Server** here. Booting with PXE is not possible unless you define it. A common network configuration is to have addresses like 192.168.0.x with 192.168.0.1 as the router (often a workstation connected both to the internet, both to the LAN, and sharing the internet connection on the LAN).



Tftpd32: Settings

Base Directory: D:\PXE\I386\templates [Browse]

Global Settings
☑ TFTP Server ☑ Syslog Server
☑ TFTP Client ☑ DHCP Server
□ SNTP server

Syslog server
□ Save syslog message
File:

TFTP Security
○ None
● Standard
○ High
○ Read Only

TFTP configuration
Timeout (seconds): 3
Max Retransmit: 6
Tftp port: 69

Advanced TFTP Options
☑ Option negotiation ☑ Hide Window at startup
□ PXE Compatibility □ Create "dir.txt" files
☑ Show Progress bar □ Beep for long transfer
□ Translate Unix file names
☑ Use Tftpd32 only on this interface: 192.168.7.1
□ Use anticipation window of 0 Bytes
☑ Allow '\' As virtual root

[OK] [Help] [Cancel]

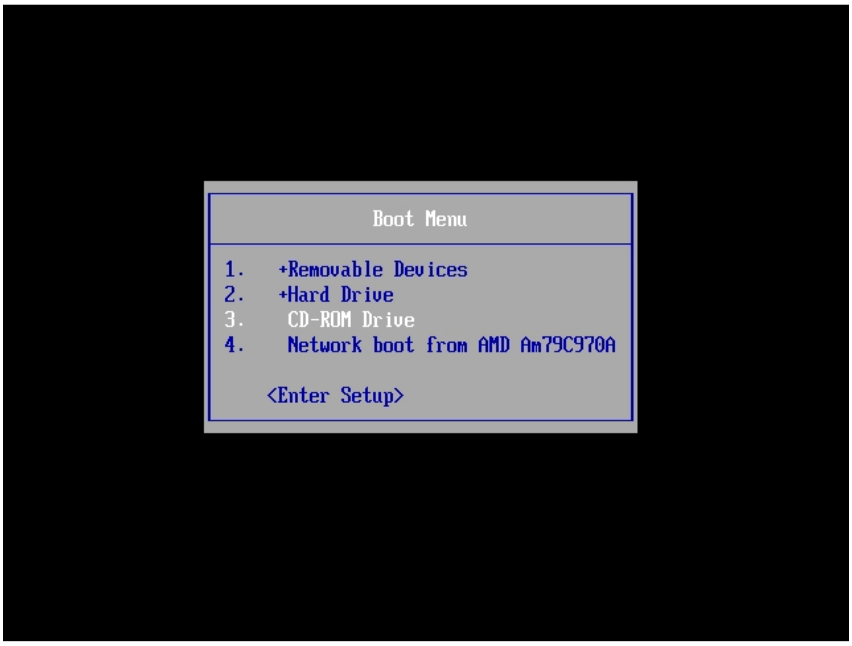
Nothing special here.

6. **That's all !** Test your PXE Server: go to a LAN workstation, reboot it, and press F12 to boot on the network. Note: if you don't see the F12 prompt, you might have to modify the BIOS setup and enable PXE.

   In the following section, we'll be booting from a CD, made by burning the PING ISO. But this should give exactly the same result.

   Please note that some screenshots might not be up to date, for PING is frequently improved.

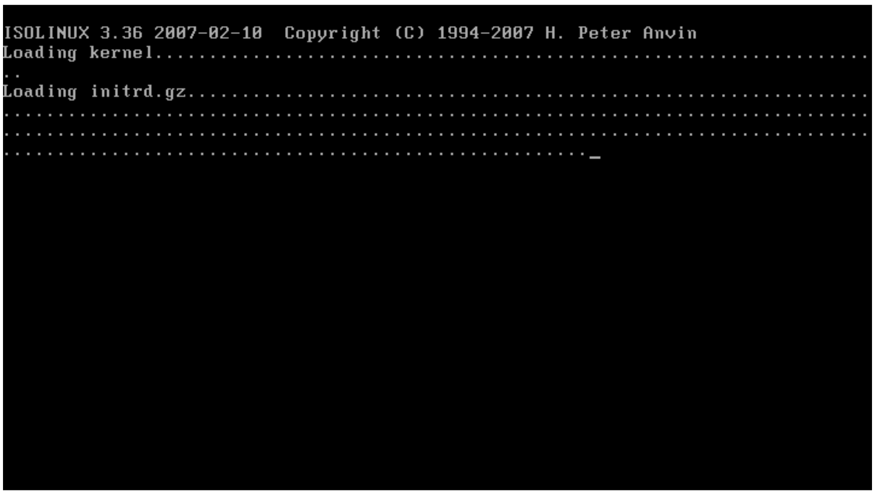
# Using PING



**Choose to boot on the CD you've burnt with the PING ISO.**

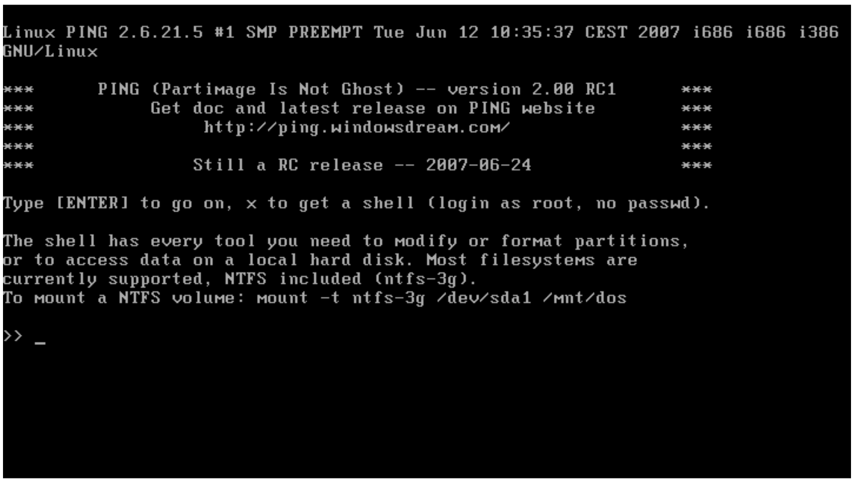
If you don't see the option in your boot sequence, you'll probably have to modify your BIOS settings.

Note that a PXE Server is needed to boot from the network.



```
ISOLINUX 3.36 2007-02-10  Copyright (C) 1994-2007 H. Peter Anvin
Loading kernel...............................................................
..
Loading initrd.gz............................................................
.............................................................................
.............................................................................
......................................................_
```

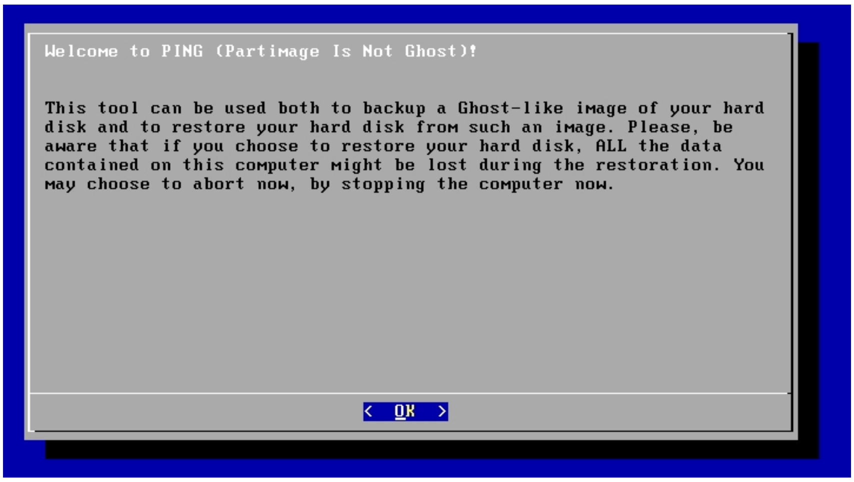
The Linux distribution is starting.



```
Linux PING 2.6.21.5 #1 SMP PREEMPT Tue Jun 12 10:35:37 CEST 2007 i686 i686 i386
GNU/Linux

***      PING (Partimage Is Not Ghost) -- version 2.00 RC1       ***
***          Get doc and latest release on PING website          ***
***               http://ping.windowsdream.com/                  ***
***                                                              ***
***              Still a RC release -- 2007-06-24                ***

Type [ENTER] to go on, x to get a shell (login as root, no passwd).

The shell has every tool you need to modify or format partitions,
or to access data on a local hard disk. Most filesystems are
currently supported, NTFS included (ntfs-3g).
To mount a NTFS volume: mount -t ntfs-3g /dev/sda1 /mnt/dos

>> _
```

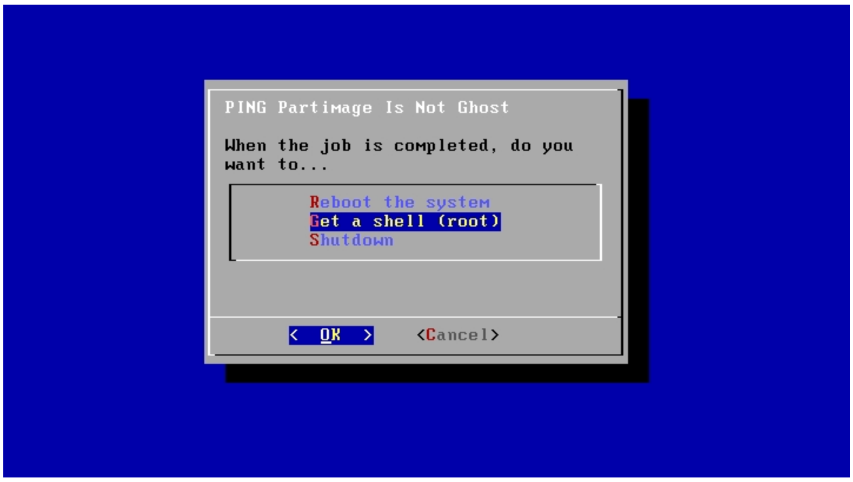
**Press ENTER** to proceed.

If you don't want to use our menu-driven wizard, press **x** and ENTER to get a shell. The root user account has no password.



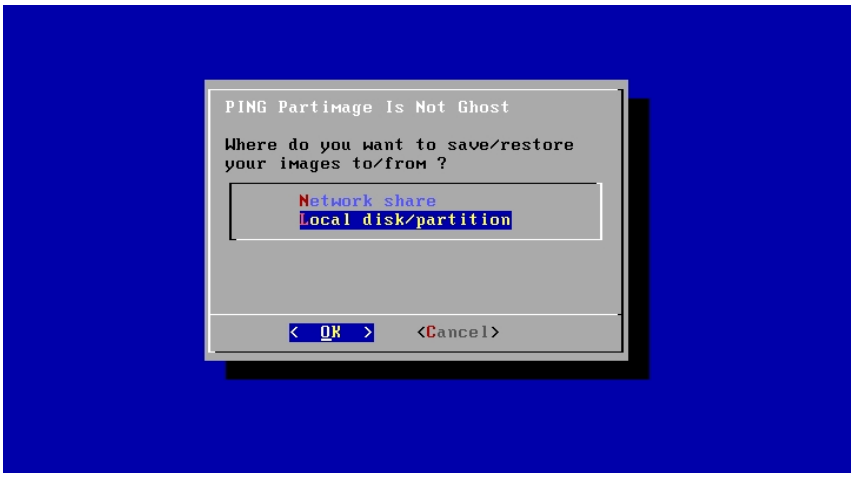
```
Welcome to PING (Partimage Is Not Ghost)!

This tool can be used both to backup a Ghost-like image of your hard
disk and to restore your hard disk from such an image. Please, be
aware that if you choose to restore your hard disk, ALL the data
contained on this computer might be lost during the restoration. You
may choose to abort now, by stopping the computer now.

                              <  OK  >
```

Read the warnings carefully and **press ENTER** to proceed.



```
PING Partimage Is Not Ghost

When the job is completed, do you
want to...

          Reboot the system
          Get a shell (root)
          Shutdown

     <  OK  >       <Cancel>
```

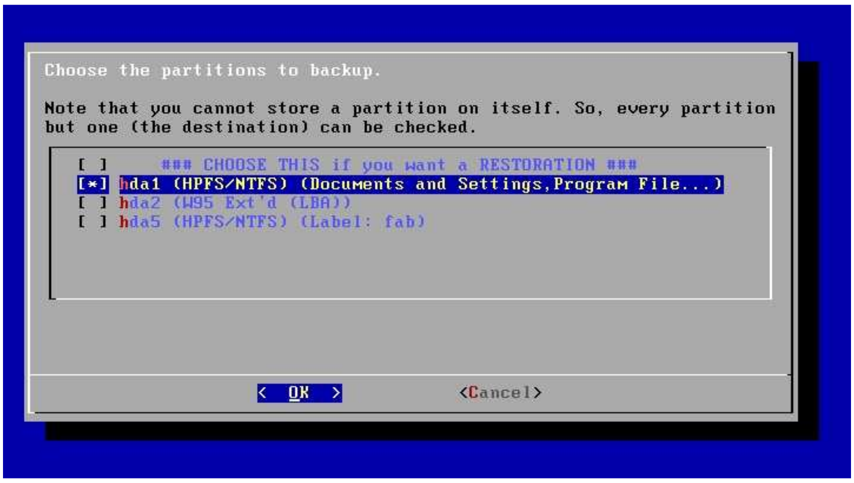
Choose what should happen at the end of the backup or restoration process. You might want to get a shell to access the log. The command line instruction is "vi /tmp/x.log".



```
PING Partimage Is Not Ghost

Where do you want to save/restore
your images to/from ?

        Network share
        Local disk/partition

     <  OK  >     <Cancel>
```

You can backup to and restore from a local partition or a network drive.

**These illustrations will start with a local backup.**

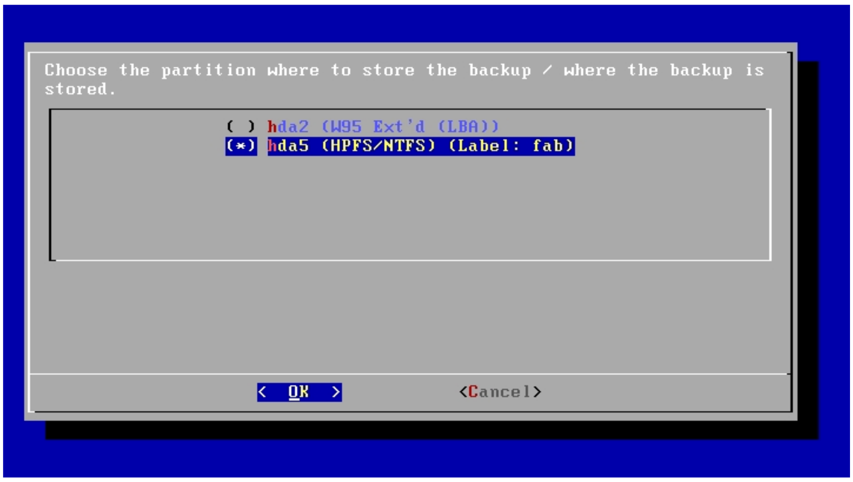


```
Choose the partitions to backup.

Note that you cannot store a partition on itself. So, every partition
but one (the destination) can be checked.

  [ ]     ### CHOOSE THIS if you want a RESTORATION ###
  [*] hda1 (HPFS/NTFS) (Documents and Settings,Program File...)
  [ ] hda2 (W95 Ext'd (LBA))
  [ ] hda5 (HPFS/NTFS) (Label: fab)

              <  OK  >          <Cancel>
```

All available partitions should be enumerated here, with either a NTFS label or the first few directories.

If you want to restore an image to the workstation, select the first option.

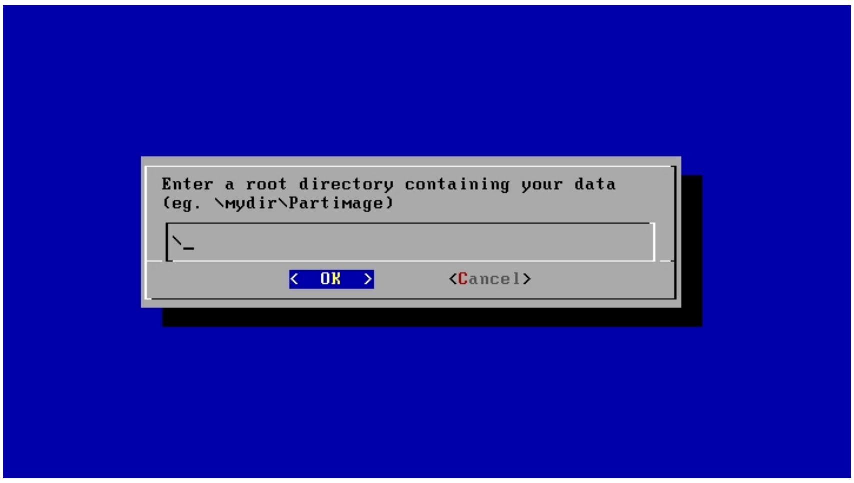
Otherwise, select all partitions you want to backup. Be aware that you should leave at least one partition unselected, as PING won't store a partition onto itself.
An Exception: if you've got only one partition, you'll be able to select it, and the script will suggest splitting the partition into two partitions. Although this works, you may want to repartition the drive before running PING.



```
Choose the partition where to store the backup / where the backup is
stored.

                   ( ) hda2 (W95 Ext'd (LBA))
                   (*) hda5 (HPFS/NTFS) (Label: fab)

              <  OK  >          <Cancel>
```

Now, choose where to store the image.

In our example, we had the C: part on /dev/hda1 and a D: part on /dev/hda5 (which Windows always stores in an extended /dev/hda2 partition). We'll store an image of C: on D:.

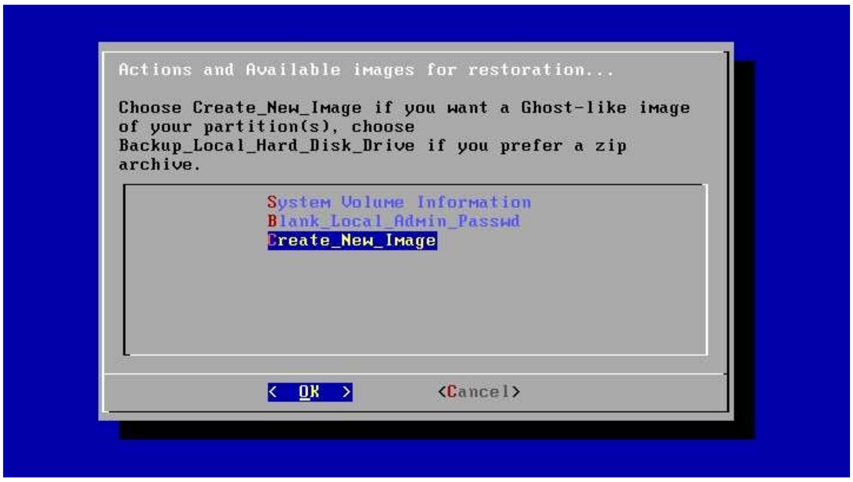


```
Enter a root directory containing your data
(eg. \mydir\Partimage)
\_
```

If you've prepared a \Partimage directory on your destination partition, enter it here.

In our case, we had not, and so we entered the root directory (and our first image will be stored on D:\).

It's a good idea to have everything stored in a dedicated directory.



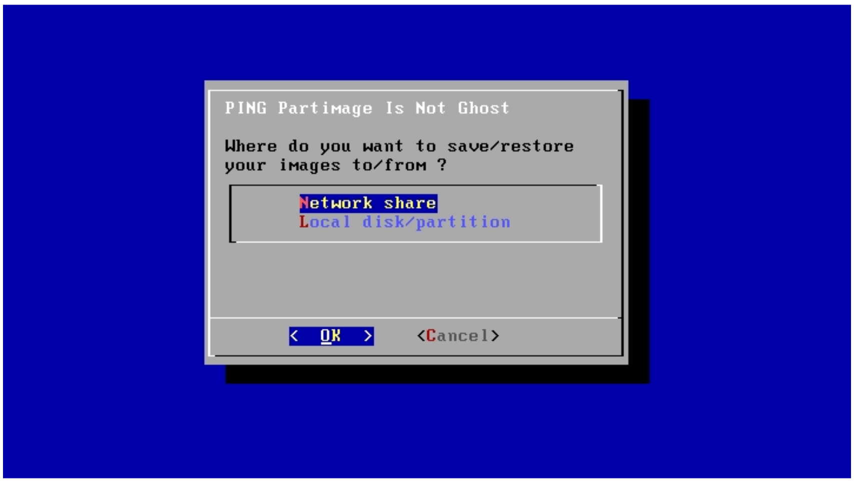
```
Actions and Available images for restoration...

Choose Create_New_Image if you want a Ghost-like image
of your partition(s), choose
Backup_Local_Hard_Disk_Drive if you prefer a zip
archive.

System Volume Information
Blank_Local_Admin_Passwd
Create_New_Image
```

Here, you'll find the **list of available images**. They are stored in the directory we've just chosen. Choosing one image will restore the computer with it, thus erasing all your hard disk.

Apart the images themselves, you'll always find 2 special choices, obviously necessary to let you ask for an image (or blank Windows administrator password, if needed.)

In our example, we had chosen the D:\ directory, and D:\ was empty. That's why you can see a "System Volume Information" directory, present on any NTFS partition. (We should have created a D:\Partimage directory and chosen it for the example to be clearer – sorry.)

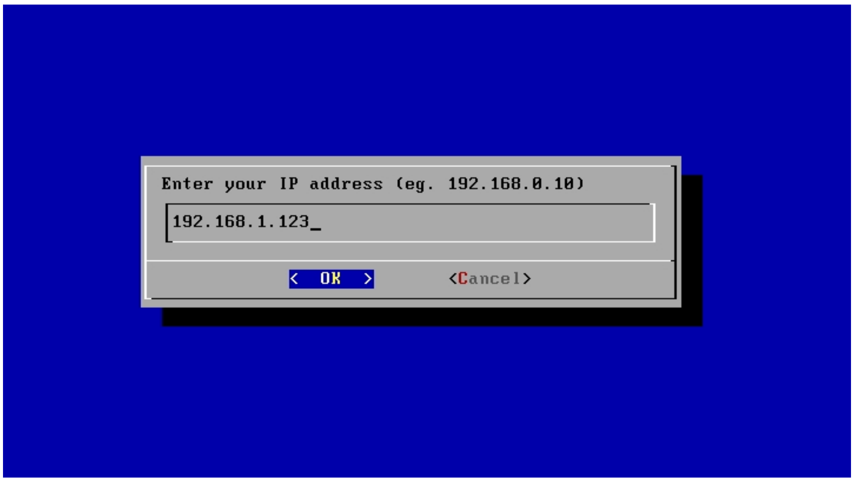
***Now, let's explain what should have happened if we had chosen to store the image onto a Network Share rather than a Local Partition…***



```
PING Partimage Is Not Ghost

Where do you want to save/restore
your images to/from ?

        Network share
        Local disk/partition

        <  OK  >      <Cancel>
```

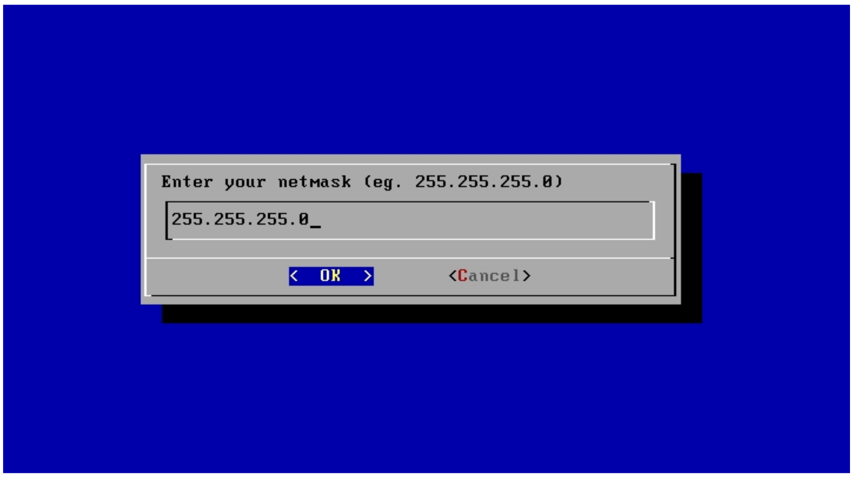
Remember where we were…

When choosing this option, the script will try to get an IP through DHCP. If this fails, you'll have the possibility to give manual parameters, as in next screenshots.



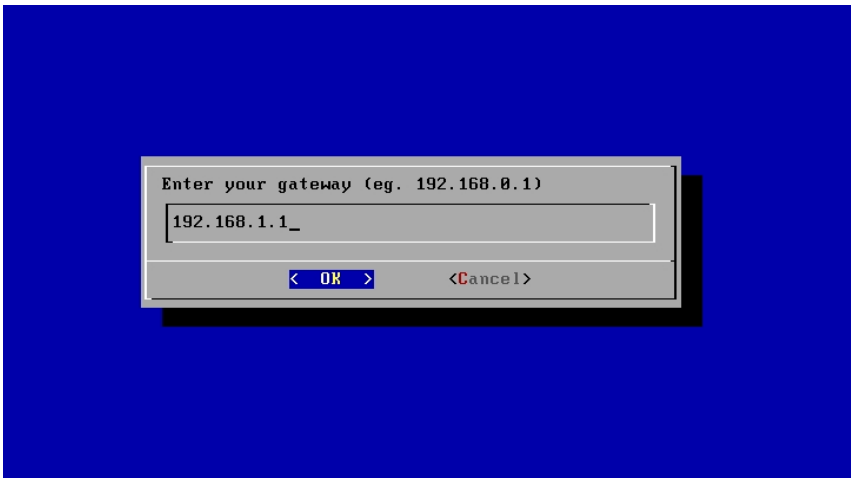
```
Enter your IP address (eg. 192.168.0.10)
192.168.1.123_
        <  OK  >      <Cancel>
```

Give your IP address.



```
Enter your netmask (eg. 255.255.255.0)
255.255.255.0_
        <  OK  >      <Cancel>
```

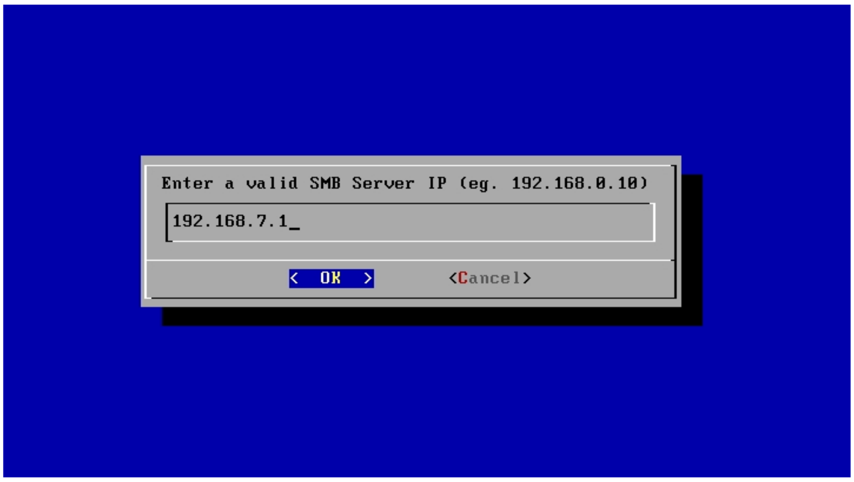
…and your netmask…



```
Enter your gateway (eg. 192.168.0.1)
192.168.1.1_
```

…and finally your gateway.

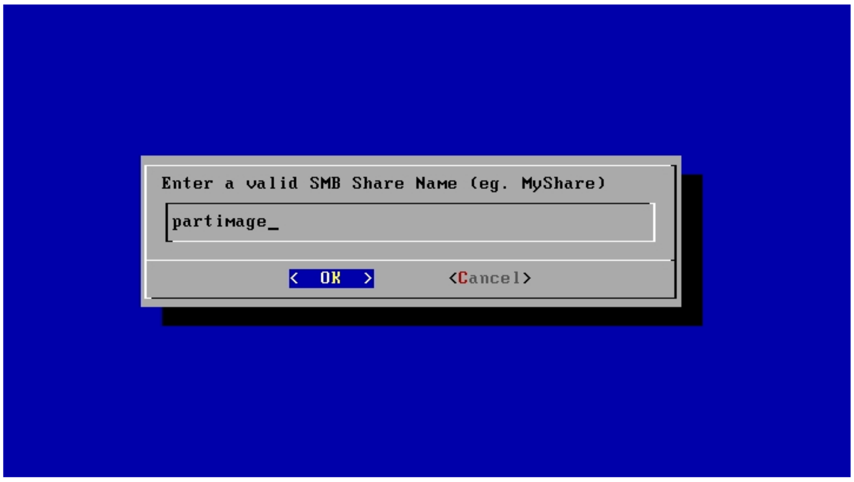
*Note : the following shows accessing a SMB (Microsoft Netbios) network share. But PING has been developed and now supports also NFS, FTP (and, somewhat, SSH). Same idea.*



```
Enter a valid SMB Server IP (eg. 192.168.0.10)
192.168.7.1_
```

**We need to know where to save the image.**

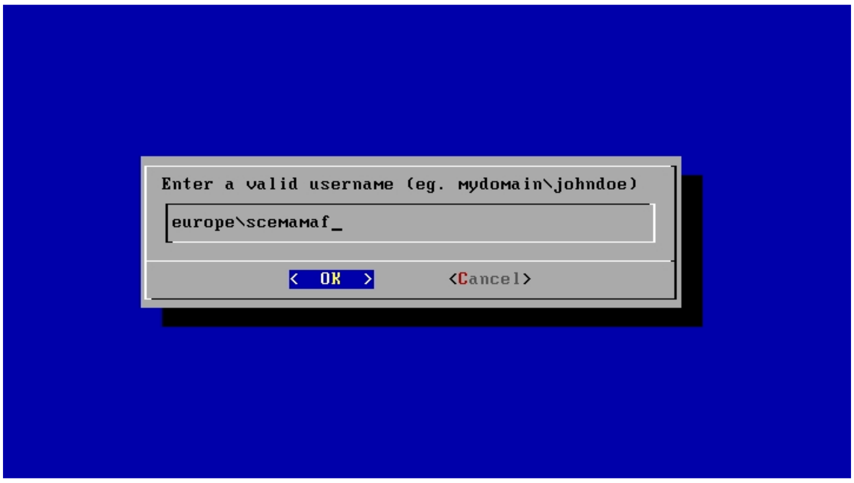
Give the IP of the computer you've shared the **PartImage** folder on (as explained in the prerequisites).



```
Enter a valid SMB Share Name (eg. MyShare)
partimage_
```

Now, the **name of the share.**

Example:
\\mydomain\temp => the share is called temp.

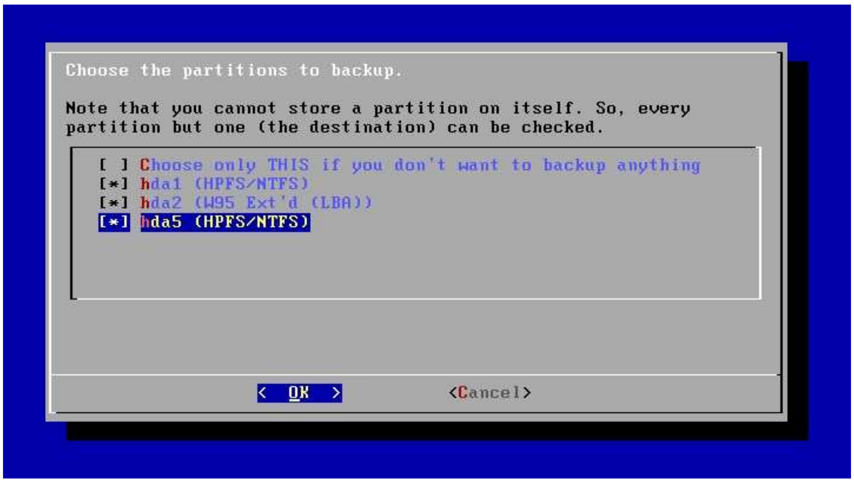


```
Enter a valid username (eg. mydomain\johndoe)
europe\scemamaf_
< OK >        <Cancel>
```

Now, the **user name** on the network.

It might also be a local account (eg.: administrator).

(You'll be asked for the password afterwards, and it won't be displayed when typing it.)

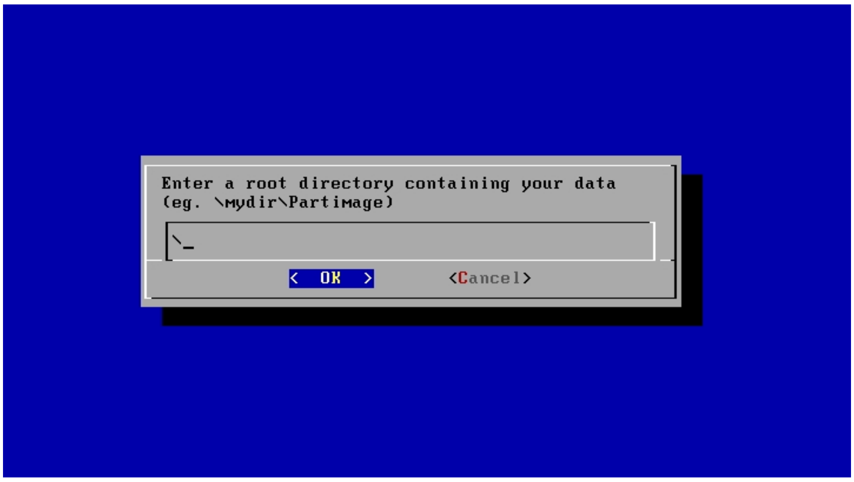


```
Choose the partitions to backup.

Note that you cannot store a partition on itself. So, every
partition but one (the destination) can be checked.

  [ ] Choose only THIS if you don't want to backup anything
  [*] hda1 (HPFS/NTFS)
  [*] hda2 (W95 Ext'd (LBA))
  [*] hda5 (HPFS/NTFS)

< OK >        <Cancel>
```

All connected hard disk drives, USB mass storage devices included, will be scanned and all available partitions should be enumerated here, with either a NTFS label, either first found directories.

If you want to restore an image to the workstation, check the first option.

Elsewhere, choose all the partitions you want to backup. You can choose all of them.

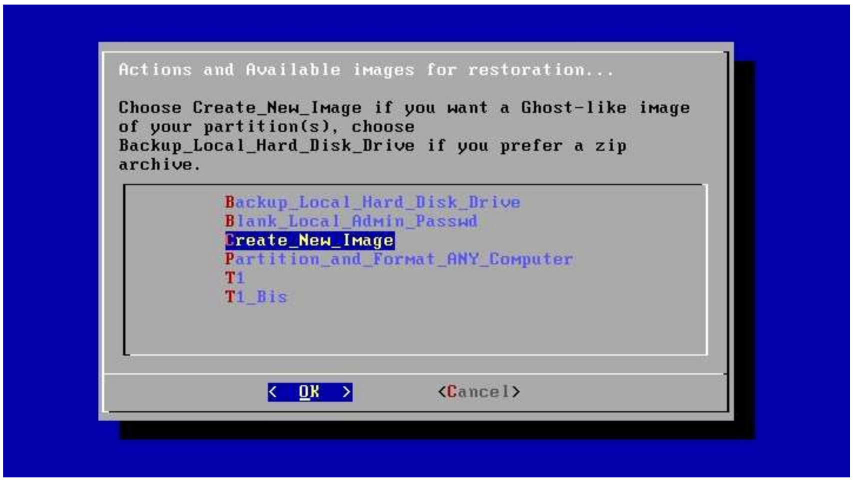


```
Enter a root directory containing your data
(eg. \mydir\Partimage)
\_
< OK >        <Cancel>
```

Now, the **directory path**, from the share to the PartImage directory.

**Example:** if you created C:\PartImage on server Billy, and shared it, then your resource should be **\\billy\partimage**. So, IP is Billy's IP, share is partimage, and directory is nothing.

**Example:** if you created C:\temp\PartImage on server Billy, but shared the temp directory, then your resource should be **\\billy\temp\partimage**. So, IP is Billy's IP, share is temp, and the directory is \partimage.

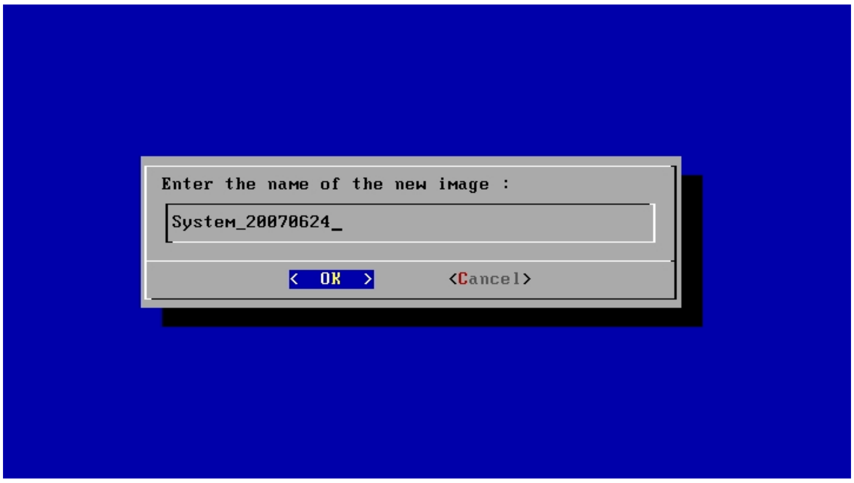


```
Actions and Available images for restoration...

Choose Create_New_Image if you want a Ghost-like image
of your partition(s), choose
Backup_Local_Hard_Disk_Drive if you prefer a zip
archive.

            Backup_Local_Hard_Disk_Drive
            Blank_Local_Admin_Passwd
            Create_New_Image
            Partition_and_Format_ANY_Computer
            T1
            T1_Bis

          <  OK  >          <Cancel>
```

Here, you'll find the **list of available images**. In the example, "T1" and "T1_Bis" are two available images. They are stored in the directory we've just chosen. Choosing one image will restore the computer with it, thus erasing part of or all your hard disk.

- The **Create_New_Image** option is obviously not an image, but a choice permitting to create a new one out of the system you've booted on.
- Do not confuse it with the **Backup_Local_Hard_Disk_Drive** option, which will only make a big zip of your disk (which is rarely useful).
- If you need to blank the password of your local administrator, choose **Blank_Local_Admin_Passwd**.
- Finally, the **Partition_And_Format_ANY_Computer** is used to prepare a disk for the installation of Windows, by making a 20-GB C:\ partition, and a D:\ with the remaining. The C:\ is ntfs-formatted, but not the D:\. This tool is generally useful to admins needing to install Windows XP through an unattended RIS setup. The answers .SIF file has no option to permit a 20-GB C:\ part to be created on the fly…
- Note that **partitionning**, be in Partition_And_Format_ANY_Computer or in the restoration of an image you've created **can be customized through a hda.part / sda.part file**. Go to the Annex for details.

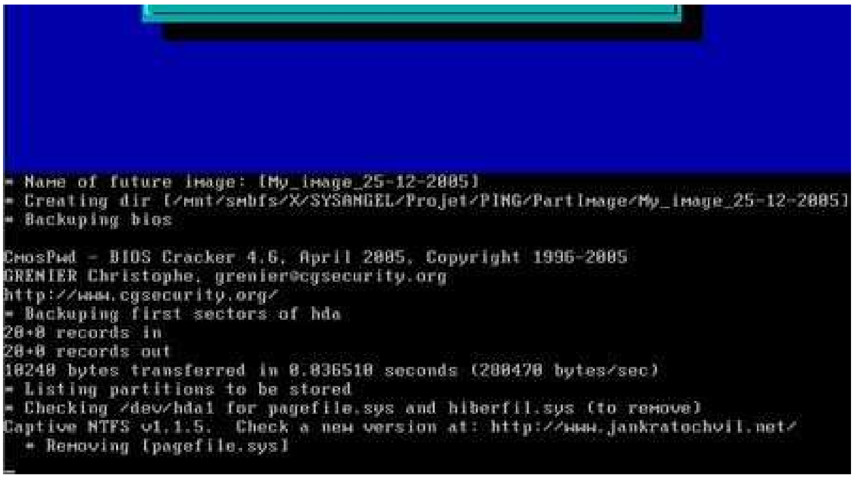
***We're back again to common trunk of the documentation, whether you chose to store the image onto a Network Share or onto a Local Partition.***



```
Enter the name of the new image :
System_20070624_
```

Finally, tell the wizard **what to name your new image.**

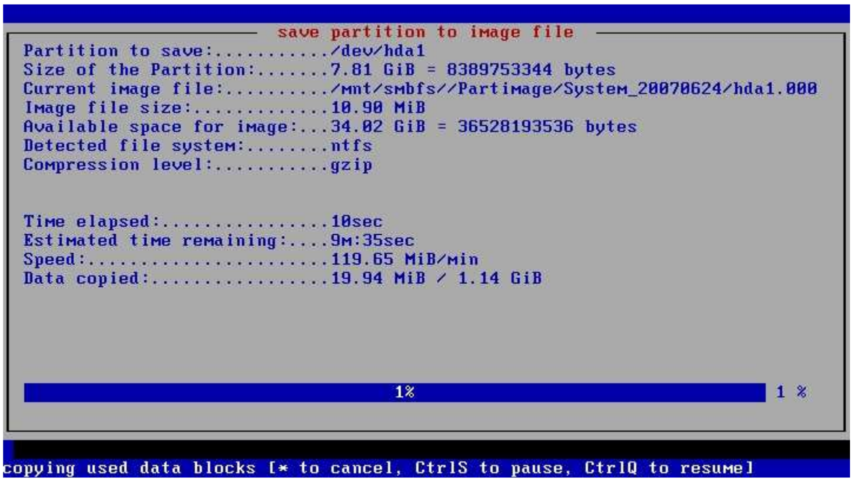
In this example, PING will create a new **\PartImage\System_20070624** directory on the destination drive, and store the backup in it.



```
* Name of future image: [My_image_25-12-2005]
* Creating dir [/mnt/smbfs/X/SYSANGEL/Projet/PING/PartImage/My_image_25-12-2005]
* Backuping bios

CmosPwd - BIOS Cracker 4.6, April 2005, Copyright 1996-2005
GRENIER Christophe, grenier@cgsecurity.org
http://www.cgsecurity.org/
* Backuping first sectors of hda
20+0 records in
20+0 records out
10240 bytes transferred in 0.036510 seconds (280470 bytes/sec)
* Listing partitions to be stored
* Checking /dev/hda1 for pagefile.sys and hiberfil.sys (to remove)
Captive NTFS v1.1.5.  Check a new version at: http://www.jankratochvil.net/
  * Removing [pagefile.sys]
_
```

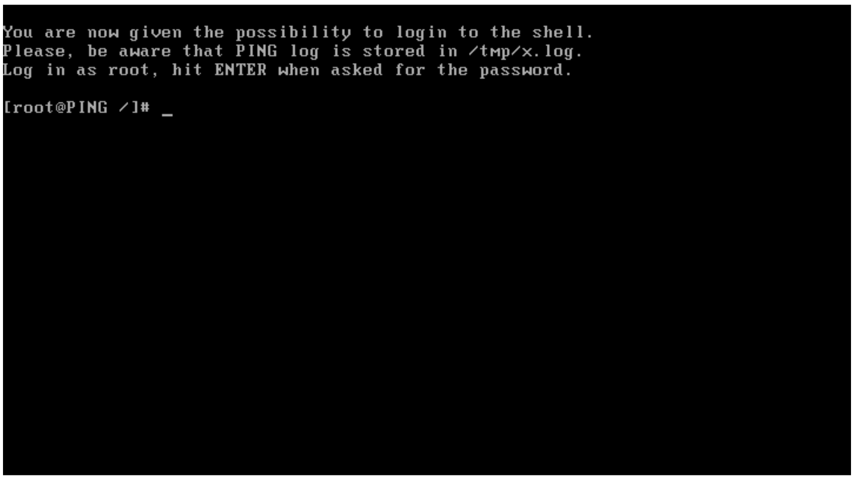
**That's all.**

In this screenshot, BIOS settings are being saved...



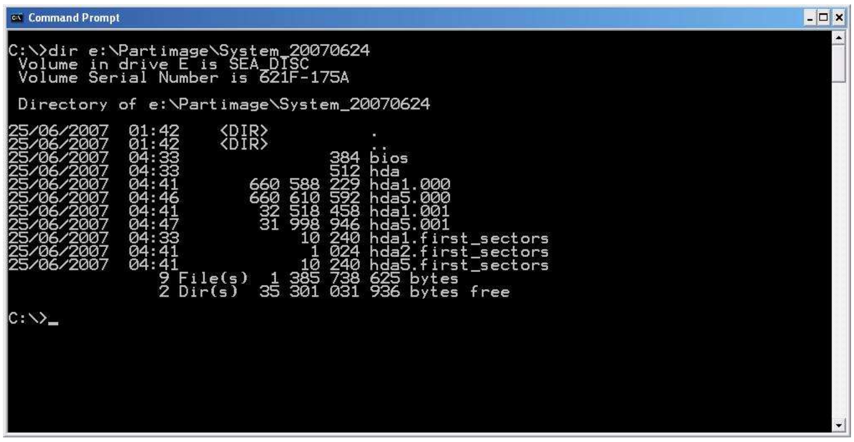
```
                    save partition to image file
Partition to save:............/dev/hda1
Size of the Partition:.......7.81 GiB = 8389753344 bytes
Current image file:........../mnt/smbfs//Partimage/System_20070624/hda1.000
Image file size:.............10.90 MiB
Available space for image:...34.02 GiB = 36528193536 bytes
Detected file system:........ntfs
Compression level:...........gzip


Time elapsed:................10sec
Estimated time remaining:....9m:35sec
Speed:.......................119.65 MiB/min
Data copied:.................19.94 MiB / 1.14 GiB



                           1%                                  1 %

copying used data blocks [* to cancel, CtrlS to pause, CtrlQ to resume]
```

**Then, the partitions...**



```
You are now given the possibility to login to the shell.
Please, be aware that PING log is stored in /tmp/x.log.
Log in as root, hit ENTER when asked for the password.

[root@PING /]# _
```

When the process is over, we get the shell we asked for.



```
C:\>dir e:\Partimage\System_20070624
 Volume in drive E is SEA_DISC
 Volume Serial Number is 621F-175A

 Directory of e:\Partimage\System_20070624

25/06/2007  01:42    <DIR>          .
25/06/2007  01:42    <DIR>          ..
25/06/2007  04:33               384 bios
25/06/2007  04:33               512 hda
25/06/2007  04:41       660 588 229 hda1.000
25/06/2007  04:46       660 610 592 hda5.000
25/06/2007  04:41        32 518 458 hda1.001
25/06/2007  04:47        31 998 946 hda5.001
25/06/2007  04:33            10 240 hda1.first_sectors
25/06/2007  04:41             1 024 hda2.first_sectors
25/06/2007  04:41            10 240 hda5.first_sectors
               9 File(s)  1 385 738 625 bytes
               2 Dir(s)  35 301 031 936 bytes free

C:\>_
```

On the directory we've stored the image into, you should get this kind of directory listing if you use the dir command in the Windows Command Prompt.

**File Descriptions:**

- **bios** stores a copy of your bios settings, and will be restored too. If you don't want to restore or store your BIOS settings, delete this file.
- **hda** stores the first sector of the first hard disk drive. The file might be called sda if SCSI or S-ATA. The Master Boot Record (MBR) should be inside.
- **hda1.000** and **hda1.001** (or sda1.000 if SCSI/S-ATA) are the partimage image files of the first partition of the first hard disk drive. Delete these files if you don't want this partition to be restored.
- **hda1.first_sectors** (or sda1.first_sectors if SCSI/S-ATA) stores the 20 first sectors of the first partition of the first hard disk drive. They will be restored before the partimaged image, to ensure any boot sector stored in a partition rather than in the MBR will be restored. The LVM structure of a partition (Linux-context only) is also stored here.
- **HDD_Look.txt** stores the list of hard disk drives and partitions that have been found on the box you've written an image from. It's not used at this time.

# Storing an image on a Bootable ISO

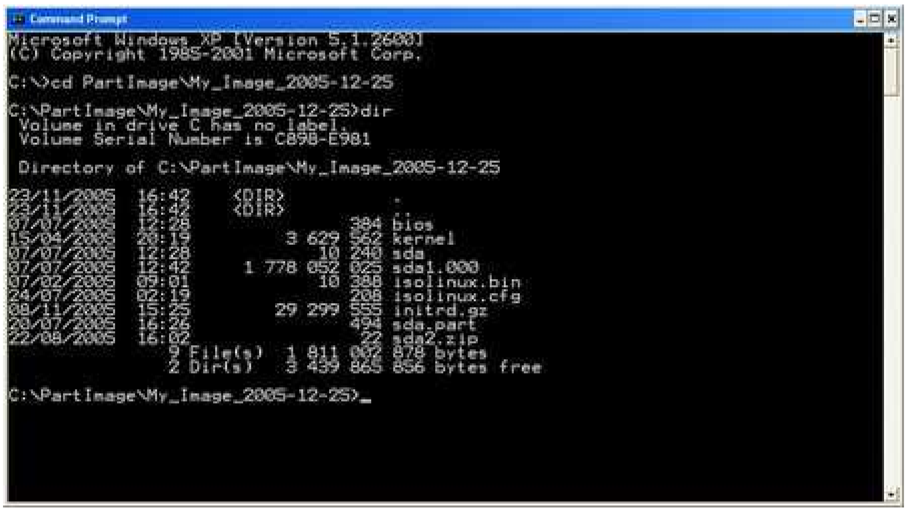
Storing an image on a LAN computer is nice, but sometimes, you may want to burn this image on a bootable DVD. It can be useful to bring such a media to a remote site, and regenerate a computer. It can also be considered as a practical way to backup a whole system.

*Directory names are given here as samples. Adapt the following according to your specific needs and configuration.*

1. We suppose you've recorder an image within the **C:\PartImage\** directory. The image is called **My_Image_2005-12-25**.
2. **Download initrd.gz** and save this file within the **C:\PartImage\My_Image_2005-12-25\** directory.
3. **Download kernel** and save this file within the **C:\PartImage\My_Image_2005-12-25\** directory.
4. **Download isolinux.bin** and save this file within the **C:\PartImage\My_Image_2005-12-25\** directory.
5. **Create a text file, called isolinux.cfg**, within the **C:\PartImage\My_Image_2005-12-25\** directory, containing:

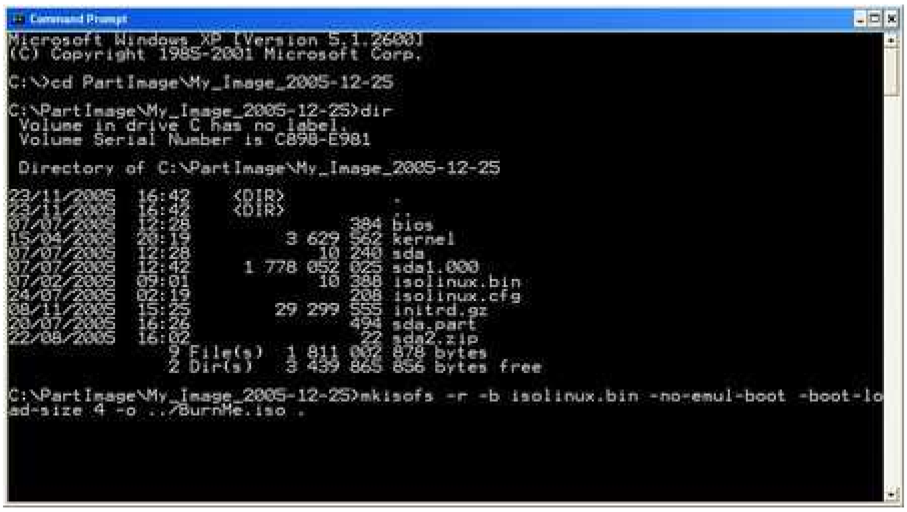
```
DEFAULT rescue
PROMPT 0
LABEL rescue
KERNEL kernel
APPEND vga=normal devfs=nomount pxe ramdisk_size=33000 load_ramdisk=1
init=/linuxrc prompt_ramdisk=0 initrd=initrd.gz root=/dev/ram0 rw noapic
nolapic lba combined_mode=libata ide0=noprobe nomce pci=nommconf pci=nomsi
irqpoll
```

6. **Download CDR Tools 2.0.1**. Install the EXE (there's an embedded MSI) on your computer.
7. Open a **DOS Prompt** and go to your image directory:



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\>cd PartImage\My_Image_2005-12-25

C:\PartImage\My_Image_2005-12-25>dir
 Volume in drive C has no label.
 Volume Serial Number is C898-E981

 Directory of C:\PartImage\My_Image_2005-12-25

23/11/2005  16:42    <DIR>          .
23/11/2005  16:42    <DIR>          ..
07/07/2005  12:28               384 bios
15/04/2005  20:19         3 629 562 kernel
07/07/2005  12:28            10 240 sda
07/07/2005  12:42     1 778 052 025 sda1.000
07/02/2005  09:01            10 388 isolinux.bin
24/07/2005  02:19               208 isolinux.cfg
08/11/2005  15:25        29 299 555 initrd.gz
20/07/2005  16:26               494 sda.part
22/08/2005  16:02                22 sda2.zip
               9 File(s)  1 811 002 878 bytes
               2 Dir(s)   3 439 865 856 bytes free

C:\PartImage\My_Image_2005-12-25>_
```

8. **Use mkisofs** (from CDR Tools) to generate a bootable ISO:

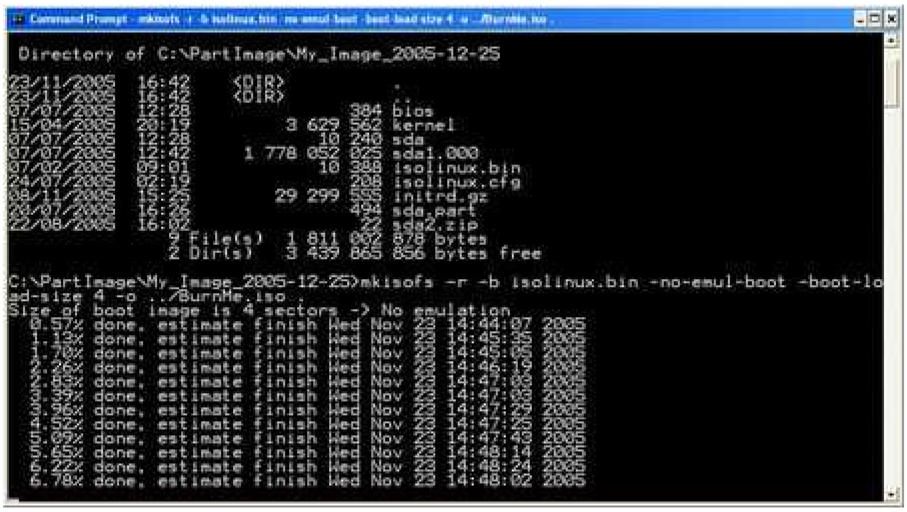


```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\>cd PartImage\My_Image_2005-12-25

C:\PartImage\My_Image_2005-12-25>dir
 Volume in drive C has no label.
 Volume Serial Number is C898-E981

 Directory of C:\PartImage\My_Image_2005-12-25

23/11/2005  16:42    <DIR>          .
23/11/2005  16:42    <DIR>          ..
07/07/2005  12:28               384 bios
15/04/2005  20:19         3 629 562 kernel
07/07/2005  12:28            10 240 sda
07/07/2005  12:42     1 778 052 025 sda1.000
07/02/2005  09:01            10 388 isolinux.bin
24/07/2005  02:19               208 isolinux.cfg
08/11/2005  15:25        29 299 555 initrd.gz
20/07/2005  16:26               494 sda.part
22/08/2005  16:02                22 sda2.zip
               9 File(s)  1 811 002 878 bytes
               2 Dir(s)   3 439 865 856 bytes free

C:\PartImage\My_Image_2005-12-25>mkisofs -r -b isolinux.bin -no-emul-boot -boot-lo
ad-size 4 -o ../BurnMe.iso .
```

mkisofs -r -b isolinux.bin -boot-info-table -no-emul-boot -boot-load-size 4 -o ../BurnMe.iso .

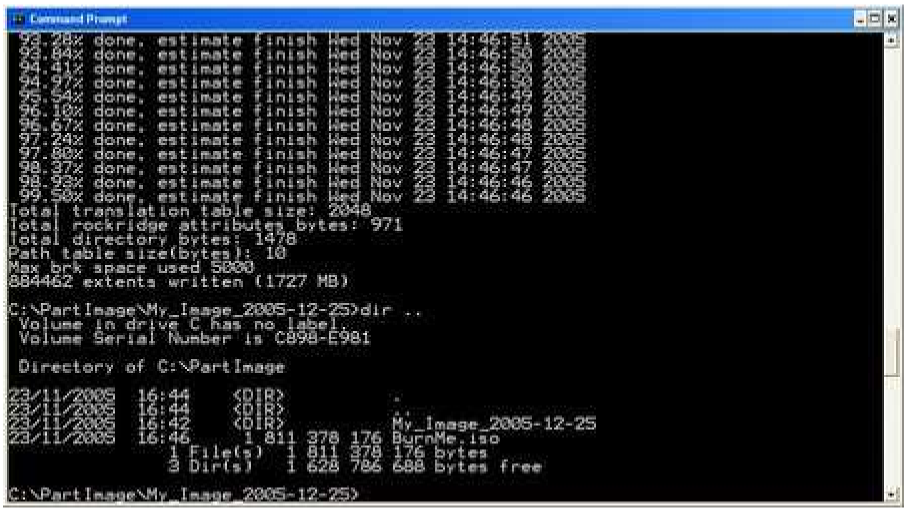
/!\ Don't forget the trailing dot. If you're not familiar with DOS, remember that mkisofs.exe has to be stored somewhere in your PATH for this command to work. Why not store it in your C:\WINDOWS\SYSTEM32 directory ?

9. The **ISO generation** should start...:


Command Prompt - mkisofs -r -b isolinux.bin -no-emul-boot -boot-load-size 4 -o ../BurnMe.iso .

```
 Directory of C:\PartImage\My_Image_2005-12-25

23/11/2005  16:42    <DIR>          .
23/11/2005  16:42    <DIR>          ..
07/07/2005  12:28               384 bios
15/04/2005  20:19         3 629 562 kernel
07/07/2005  12:28            10 240 sda
07/07/2005  12:42     1 778 052 025 sda1.000
07/02/2005  09:01            10 388 isolinux.bin
24/07/2005  02:19               208 isolinux.cfg
08/11/2005  15:25        29 299 555 initrd.gz
20/07/2005  16:26               494 sda.part
22/08/2005  16:02                22 sda2.zip
               9 File(s)  1 811 002 878 bytes
               2 Dir(s)   3 439 865 856 bytes free

C:\PartImage\My_Image_2005-12-25>mkisofs -r -b isolinux.bin -no-emul-boot -boot-lo
ad-size 4 -o ../BurnMe.iso .
Size of boot image is 4 sectors -> No emulation
  0.57% done, estimate finish Wed Nov 23 14:44:07 2005
  1.13% done, estimate finish Wed Nov 23 14:45:35 2005
  1.70% done, estimate finish Wed Nov 23 14:45:05 2005
  2.26% done, estimate finish Wed Nov 23 14:46:19 2005
  2.83% done, estimate finish Wed Nov 23 14:47:03 2005
  3.39% done, estimate finish Wed Nov 23 14:47:03 2005
  3.96% done, estimate finish Wed Nov 23 14:47:29 2005
  4.52% done, estimate finish Wed Nov 23 14:47:25 2005
  5.09% done, estimate finish Wed Nov 23 14:47:43 2005
  5.65% done, estimate finish Wed Nov 23 14:48:14 2005
  6.22% done, estimate finish Wed Nov 23 14:48:24 2005
  6.78% done, estimate finish Wed Nov 23 14:48:02 2005
```

10. And you should get a full working **C:\PartImage\BurnMe.iso** image.



```
 93.28% done, estimate finish Wed Nov 23 14:46:51 2005
 93.84% done, estimate finish Wed Nov 23 14:46:50 2005
 94.41% done, estimate finish Wed Nov 23 14:46:50 2005
 94.97% done, estimate finish Wed Nov 23 14:46:50 2005
 95.54% done, estimate finish Wed Nov 23 14:46:49 2005
 96.10% done, estimate finish Wed Nov 23 14:46:49 2005
 96.67% done, estimate finish Wed Nov 23 14:46:48 2005
 97.24% done, estimate finish Wed Nov 23 14:46:48 2005
 97.80% done, estimate finish Wed Nov 23 14:46:47 2005
 98.37% done, estimate finish Wed Nov 23 14:46:47 2005
 98.93% done, estimate finish Wed Nov 23 14:46:46 2005
 99.50% done, estimate finish Wed Nov 23 14:46:46 2005
Total translation table size: 2048
Total rockridge attributes bytes: 971
Total directory bytes: 1478
Path table size(bytes): 10
Max brk space used 5000
884462 extents written (1727 MB)

C:\PartImage\My_Image_2005-12-25>dir ..
 Volume in drive C has no label.
 Volume Serial Number is C898-E981

 Directory of C:\PartImage

23/11/2005  16:44    <DIR>          .
23/11/2005  16:44    <DIR>          ..
23/11/2005  16:42    <DIR>          My_Image_2005-12-25
23/11/2005  16:46     1 811 378 176 BurnMe.iso
               1 File(s)  1 811 378 176 bytes
               3 Dir(s)   1 628 786 688 bytes free

C:\PartImage\My_Image_2005-12-25>
```

11. **Burn the ISO.** That's all.

**Note:** You might use Ahead Nero or any third-party software to generate a bootable ISO. Your mileage might vary.

# Annex

## Notes :

- To **suggest a new feature**, please visit the **Forum** (PING section).
- **TIP:** if you've got to **duplicate an image over several computers** within a **LAN**, you'll have to **regenerate Windows' SID**. To do so, use **newsid.exe** from **SysInternals**.

## Special features / FAQ :

- **The PING.CONF file (read the whole paragraph before doing…)**

  If you're bored with always having to re-type network and share settings…

  - You can add an answers file to the Linux image. It's path is /etc/ping.conf and it already contains all available keys, in commented lines.
  - To modify it to suit your needs :

```
1. Boot PING on a machine that has at least 256 MB of RAM
2. Get a shell when proposed to, by typing 'x' and Enter
3. Login as root, no password
4. Connect to your network:
   - If there's a DHCP server, type: dhcpcd eth0
   - Else, type (adjust to your config):
     - ifconfig eth0 192.168.0.26 netmask 255.255.255.0 up
     - route add default gw 192.168.0.1
5. Mount a remote share (where your initrd.gz file can be found):
   - mount.smbfs //192.168.0.1/d\$ /mnt/smbfs -o
     username=domain\\username
     (or just username=username if this is a local user)
6. Assuming both your initrd.gz and ping.conf files are located
   in the root of the share, type:
   - gzip -d /mnt/smbfs/initrd.gz
   - mount -o loop /mnt/smbfs/initrd /mnt/loop
   - mkdir /tmp/rootfs
   - cd /tmp/rootfs
   - tar xvfj /mnt/loop/rootfs.tar.bz2
   - cp /mnt/smbfs/ping.conf /tmp/rootfs/etc/ping.conf
   - tar cvf - * |bzip2 -9 - >/mnt/loop/rootfs.tar.bz2
   - umount /mnt/loop
   - gzip -9 /mnt/smbfs/initrd

Remember that Linux's shell is case-sensitive.
```

From PING 2.00 RC8, it's possible to avoid having to modify the image to get the same result. Every `key=value` setting contained in the ping.conf file can be passed directly to the kernel at boot time. To do this, simply edit your isolinux.cfg file (if CD/DVD, pxelinux.cfg\DEFAULT if PXE), and modify the APPEND line accordingly. Be careful that DOS linefeeds will cause problems – so, either edit the sample file available from the download section of the PING web site, either use a unix-friendly editor like Ultra-Edit (tks Fubar Bundy for the tip).

For example, to have PING leave you to a shell after job completion, you'd add the string `After_Completion=Shell` to the `APPEND` line. Note that the syntax `key="value"` is accepted, allowing space chars to be passed too.

If the ping.conf file has been customized, you still can pass keys/values to the kernel at boot time. They'll be given priority over ping.conf's data.

**List of currently supported settings :**

| | Key name | Possible Values / Examples | Description |
|---|---|---|---|
| Client network configuration | IP | Ex.: 192.168.0.1 | The IP address you want the client to be assigned. Pass no IP if you've got a DHCP and want to use it. |
| | Netmask | Ex.: 255.255.255.0 | The netmask corresponding to the IP. |
| | Gateway | Ex.: 192.168.0.1 | The gateway corresponding to the IP. |
| | DHCP_Timeout | Ex.: 15 | If you need to set a non-default timeout for DHCP, set it here in seconds. |
| Images location section | Server | Ex.: 192.168.0.1<br>Ex.: /dev/hda2,/dev/sda2 | The IP of the host where images made with PING are located.<br>The images can also be stored on a local partition; in such a case, mention it. If the images can be stored on different partitions, use commas to give several possible choices. |
| | Share | Ex.: MyShare | Name of the MS Share. |
| | User | Ex.: | User to use (can be domain\user). |

| | | | |
|---|---|---|---|
| | | administrator | |
| | Passwd | Ex.: secret | Password to send. |
| | Directory | Ex.: / | If you've made a directory in your share, give it here (ex.: \\192.168.0.1\MyShare\Partimage) |
| | | | |
| Misc | Repart | Y or N | If Repart is set to Y, and if you've got only one partition on your local hard disk drive and want to store an image of it locally, then this unique partition will be splitted for you. If set to N, PING will simply exit. And if commented, you'll be asked for what to do. May we suggest you'd better partition yourself for this kind of situation. |
| | After_Completion | Shell or Reboot or Shutdown | What will PING do after the job is finished ? if Shell, you'll be dropped to a Linux shell. Useful if you want to check the /tmp/x.log log. |
| | Replace_BIOS | Y or N | Will we replace the BIOS if a "bios" file is found ? Note that you can simply delete this "bios" file before making your restorations to avoid having your bios params modified, but in some cases, you might want to have the bios settings recorded, and to restore them on a selection of target stations. |
| | CIFS_Preferred | Y or N | To map a MS share, SAMBA can use either SMBFS either CIFS. For some reasons, it has been constated that CIFS can be much faster than SMBFS on some types of networks. And the contrary is true, too! |
| | NFS_Preferred | Y or N | Will use Sun's NFS protocol instead of Microsoft's SMB to mount the network share. Note that in this case, parameters Username and Passwd are useless. |
| | FTP_Preferred | Y or N | Will use the FTP protocol to map a distant share. Parameters Username, Passwd, Server and Directory are used, but not the Share parameter. Note that you can only restore an image by FTP, but not create one (yet). |
| | Zsplit_Preferre | Y or N | Set this to Y if you want zsplit, which is a dd-like utility, to be used instead of partimage. |

| | | | |
|---|---|---|---|
| | d | | Most users should use partimage. |
| | No_Shell | Y or N | Set this option to Y if you don't want PING to propose the user to get a shell.<br>Note that if you have also set the After_Completion key to Shell, the user will finally be given a shell. If the After_Completion key is not set at all, then the user won't be proposed to get a shell at the end of PING's job. |
| | Minimize_Before_Storing | Y or N | It can be interesting to reduce the size of the partition before storing it. Thus, it's possible to restore it later into a smaller partition. Note that even if we decide later to restore into a equally-big or bigger partition, PING will always try to maximize the size of the filesystem at the end.<br>*You are strongly advised to have a full working backup of your system before choosing this option.* |
| | AUTO | Y or N | If you want NO QUESTION to be raised, no popup at all, set this value to Y. It's equivalent to creating an empty AUTO file on the root of a PING CD/DVD (which is now deprecated, yet still supported for backwards compatibility). |
| | Cmd_1 | | Cmd_1 is a shell command for PING to execute before the first blue welcome screen. |
| | Cmd_2 | | Cmd_2 is a shell command for PING to execute after the mounting of SMB shares. |

| | | | |
|---|---|---|---|
| | Cmd_3 | | Cmd_3 is a shell command for PING to execute at the end (before giving a shell or reboot or shutdown). |
| | Force_Dirty_NTFS_Mount | Y or N | If set to Y, we'll mount R/W NTFS filesystems even if they have the dirty flag set. It's not recommended, of course, but it might help sometimes. If not set, the question will be raised. |
| | Its_HDA_Stupid | Y or N | When trying to restore an image made on a PATA drive (called for example by linux hda) to a SATA drive (called sda), first versions of PING would complain and leave. Now, if the expected target device cannot be found, PING will propose to choose among detected HDDs. If there's only one possible choice and you don't want any possible interaction to be given to the user, set this parameter to Y. |
| | Compression_Type | Ex.: gzip | Set the compressor to be used. Else, will be asked. Can be set to 'gzip', 'bzip2' or 'no compression'. |

- **Can I restore an image to a bigger hard disk drive ?**
    - Yes. Moreover, if the file system is NTFS, PING will always try to maximize its size, so you won't end up with unallocated disk space.

- **Can I restore an image to a smaller hard disk drive ?**
    - Yes, roughly, starting with version 2.01.10 of PING.
    - How does it work ? at the recording of the image, you should choose to have the filesystem minimized as much as possible before being written. Then, it

should be easy to restore it anywhere, since you can restore an image to a bigger partition.

- **I've made a bootable DVD with my image and want no user interactivity at all.**
    - Before creating your ISO with mkisofs, add an empty file to the directory containing your image, and call it "AUTO" (no extention). When seeing this file, PING will raise no question… unless the answer is really needed. Combine the AUTO file with a customized /etc/ping.conf.
    - You can also set the AUTO parameter, either in the /etc/ping.conf file, either by passing it to the kernel. See the params section for more details.
    - Of course, use at your own risk, as such a DVD will boot and scratch everything ☺

- **What happens if I use a filesystem that is unknown to partimage ?**
    - Then PING will record the partition with zsplit. It's just like dd, yet faster. It will store your partition as bytes and not as files. Therefore, it will be much longer than what partimage would have done, and use far more space.
    - If you've got to industrialize the deployment of such partitions, consider blanking unused space (dd if=/dev/zero of=foobar ; rm foobar), and defragmenting. Then you should lose no space.

- **Can I boot on a USB flash drive ?**
    - Yes. Thanks to Blas and Fubar Bundy for the research & testing.
    - Mount your USB drive on Windows (say it's F: ), preferably FAT32-formated.
    - Copy the following PING files to it => kernel, initrd.gz, isolinux.cfg .
    - On F:, rename isolinux.cfg to syslinux.cfg .
    - Download SYSLINUX from http://syslinux.zytor.com/download.php
    - Extract the file syslinux.exe from the file you've downloaded to your F: drive (it should be found here in the archive: x:\syslinux-3.51\win32\syslinux.exe)
    - Open a DOS prompt, and type: f:\syslinux.exe –ma f:

- **My image is bigger than the capacity of a DVD. Can I store it on several DVDs ?**
    - Yes. This is a pretty new feature.
    - Say you've got a directory structure like this :

```
Directory of E:\Partimage\T1_Bis
```

```
24/06/2007  06:38               384 bios
24/06/2007  06:38               512 hda
24/06/2007  06:55       660 588 216 hda1.000
24/06/2007  06:56        32 518 458 hda1.001
24/06/2007  06:38            10 240 hda1.first_sectors
24/06/2007  06:56             1 024 hda2.first_sectors
24/06/2007  06:56            10 240 hda5.first_sectors
24/06/2007  10:43                38 hda2_0.spl.zp
24/06/2007  09:30       660 616 876 hda5_0.spl.zp
24/06/2007  09:38       660 615 020 hda5_1.spl.zp
24/06/2007  09:45       660 613 560 hda5_2.spl.zp
24/06/2007  09:53       660 618 499 hda5_3.spl.zp
24/06/2007  10:05       310 638 295 hda5_4.spl.zp
              13 File(s)  3 646 231 362 bytes
```

- And say you'd like to store it in 6 CD-Roms.
- Create an empty file in that directory, and call it "MULTI" (no extention).
- Add PING files to make the future ISO bootable (initrd.gz, kernel, isolinux.cfg, and isolinux.bin).
- You should have now :

```
Directory of E:\Partimage\T1_Bis

24/06/2007  06:38               384 bios
24/06/2007  06:38               512 hda
12/06/2007  10:37         4 164 232 kernel
25/06/2007  19:46                 0 MULTI
24/06/2007  06:55       660 588 216 hda1.000
24/06/2007  06:56           438 213 hda5.000
24/06/2007  06:56        32 518 458 hda1.001
15/02/2007  11:44            10 932 isolinux.bin
22/06/2007  20:33               199 isolinux.cfg
24/06/2007  06:38            10 240 hda1.first_sectors
24/06/2007  06:56             1 024 hda2.first_sectors
24/06/2007  06:56            10 240 hda5.first_sectors
25/06/2007  14:10        22 766 432 initrd.gz
24/06/2007  10:43                38 hda2_0.spl.zp
24/06/2007  09:30       660 616 876 hda5_0.spl.zp
24/06/2007  09:38       660 615 020 hda5_1.spl.zp
24/06/2007  09:45       660 613 560 hda5_2.spl.zp
24/06/2007  09:53       660 618 499 hda5_3.spl.zp
24/06/2007  10:05       310 638 295 hda5_4.spl.zp
              19 File(s)  3 673 611 370 bytes
```

- **First CD** should contain the following files :

```
Directory of E:\Partimage\T1_Bis

24/06/2007  06:38               384 bios
24/06/2007  06:38               512 hda
```

```
12/06/2007  10:37         4 164 232 kernel
25/06/2007  19:46                 0 MULTI
24/06/2007  06:55       660 588 216 hda1.000
24/06/2007  06:56        32 518 458 hda1.001
15/02/2007  11:44            10 932 isolinux.bin
22/06/2007  20:33               199 isolinux.cfg
24/06/2007  06:38            10 240 hda1.first_sectors
24/06/2007  06:56             1 024 hda2.first_sectors
25/06/2007  14:10        22 766 432 initrd.gz
24/06/2007  10:43                38 hda2_0.spl.zp
              12 File(s)    720 060 667 bytes
```

- **Second CD** should contain the following files :

```
Directory of E:\Partimage\T1_Bis

25/06/2007  19:46                 0 MULTI
24/06/2007  06:56            10 240 hda5.first_sectors
24/06/2007  09:30       660 616 876 hda5_0.spl.zp
               3 File(s)    660 627 116 bytes
```

(Note: you'll call mkisofs this way (no boot): mkisofs -r -o ../Image_2.iso .)

- **Third CD** should contain the following files :

```
Directory of E:\Partimage\T1_Bis

25/06/2007  19:46                 0 MULTI
24/06/2007  09:38       660 615 020 hda5_1.spl.zp
               2 File(s)    660 615 020 bytes
```

- **Forth CD** should contain the following files :

```
Directory of E:\Partimage\T1_Bis

25/06/2007  19:46                 0 MULTI
24/06/2007  09:45       660 613 560 hda5_2.spl.zp
               2 File(s)    660 613 560 bytes
```

- **Fifth CD** should contain the following files :

```
Directory of E:\Partimage\T1_Bis

25/06/2007  19:46                 0 MULTI
24/06/2007  09:53       660 618 499 hda5_3.spl.zp
```

```
               2 File(s)    660 618 499 bytes
```

- Finally, **sixth CD** should contain the following files : (no more MULTI).

```
Directory of E:\Partimage\T1_Bis

24/06/2007  10:05       310 638 295 hda5_4.spl.zp
               1 File(s)    310 638 295 bytes
```

- **I've got some problems with the US keyboard…**
    - To gain space, PING has been compiled with the US keyboard only. It's qwerty, and we don't plan to add other flavors.
    - It might help you to know that you can use ASCII codes. For instance, you can type a backslash (\) by holding the ALT key down and typying 92 at the same time. The ASCII code for the pipe (|) is 124.

- **How to create a custom boot image on a floppy drive for PING ?** If you happen to get an error like **"PXE-E32: TFTP open timeout"**, this may mean you need to create a custom boot disk for your NIC. Here are the steps... *(Thanks Curtis & Bobmetelsky for the tip)*
    - Identify the NIC type and model. Windows users, right click on my computer, manage, device manager, network adapters.
    - Go to http://rom-O-Matic.Net/5.4.3/ and download the image yournic###.zdsk that matches your NIC.
    - Download the Windows GUI of rawwrite http://www.chrysocome.net/rawwrite to write the image file into a floppy. (UNIX users will just cat the zdsk file to /dev/fd0).
    - Create the boot floppy. Rawwite wants the name of the yournic###.zdsk and an A:\. From there, rawwrite will do all the work of creatiing the boot floppy. You do NOT need to create a system disk or anyting like that.
    - Boot the PC from the A:\ and and answer (N)etwork when prompted. The PC will boot and then look for your PXE server. From there, follow the standard PING documentation.

- **The hard disk drive of my DELL series 2800 and 2850, which uses the Perc 4e/Di Controller, cannot be seen.**
    - The Linux kernel uses MegaRAID drivers, and they are correct.
    - Yet, you should enter your MegaRAID BIOS Set Up Utility.
    - Enter the Adapter settings menu.
    - Under Other Adapter Options, select Emulation and set it to Mass Storage.

- If the adapter is incorrectly set to "I2O" emulation, the system will attempt to load the i2o driver. This will fail, and prevent the proper driver from being loaded.

- **How can I put the PING ISO into my harddisk and boot from it ?**
  - PING can be used so to set up some kind of utility partition.
  - There are several ways to do it, and here's one of them. I've chosen to use **lilo** as a boot manager (either written in the MBR, either written into a file and called from the C:\boot.ini file), and to keep unchanged the files shipped with PING releases (that is, "kernel" and "initrd.gz"). That's the reason why there's still a ramdisk in the method described below, and even the keyword "pxe" in an append line. Really, other ways are possible.
  - Also, don't forget that you can ship your computer with an image stored on a second partition, and a bootable CDRom configured to automatically restore that image.

  - **First method – using lilo within the MBR :**
    - Boot a PING CDRom.
    - Get a shell when proposed to (type 'x').
    - In my system, there's only one IDE disk, which name is /dev/had (try sda for SATA/SCSI). So, here's the output of the "fdisk -l /dev/hda" command :

```
Disk /dev/hda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/hda1   *           1        1044     8385898+   7  HPFS/NTFS
```

- We're going to store the restoration image and PING files in a second partition. We'll use EXT3 as the filesystem, to make the partition unreadable from Windows (well, difficult to read). Note that we could store the sole kernel file in a small EXT3 partition, and the cloned images on any filesystem. Let's keep it simple – one EXT3 part.
- In our example, a 2-GB-big partition is enough. First, let's resize our unique hda1 partition with the "ntfsresize -f -s 6000M /dev/hda1" command. Output :

```
ntfsresize v2.0.0 (libntfs 10:0:0)
```

```
Device name        : /dev/hda1
NTFS volume version: 3.1
Cluster size       : 4096 bytes
Current volume size: 8587153920 bytes (8588 MB)
Current device size: 8587160064 bytes (8588 MB)
New volume size    : 5999993344 bytes (6000 MB)
Checking filesystem consistency ...
100.00 percent completed
Accounting clusters ...
Space in use       : 1577 MB (18.4%)
Collecting resizing constraints ...
Needed relocations : 0 (0 MB)
WARNING: Every sanity check passed and only the dangerous
operations left.
Make sure that important data has been backed up! Power outage or
computer
crash may result major data loss!
Are you sure you want to proceed (y/[n])?
```

- Now, using fdisk, we must resize hda1 to 6000M, and create a new partition. Command: “fdisk /dev/hda”.
- In our case, the fdisk commands are “p d n p 1 1 +6000M t 7 a 1 n p 2 enter enter w”, resulting in this “fdisk –l /dev/hda” output :

```

Disk /dev/hda: 8589 MB, 8589934592 bytes
255 heads, 63 sectors/track, 1044 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

   Device Boot      Start         End      Blocks   Id  System
/dev/hda1   *           1         730     5863693+   7  HPFS/NTFS
/dev/hda2             731        1044     2522205   83  Linux
```

- Finally, let's format hda2: “mkfs.ext3 /dev/hda2”.
- Now, let's mount the two parts :
  - mount -t ntfs-3g /dev/hda1 /mnt/dos
  - mount /dev/hda2 /mnt/linux
- df should output something like :

```

Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/hdc                 19582     19582         0 100% /mnt/cdrom
/dev/hda1              5859368   1539268   4320100  27% /mnt/dos
/dev/hda2              2482556     69304   2287144   3% /mnt/linux
```

- We've got to store PING's files in our new linux partition :
  - cp /mnt/cdrom/initrd.gz /mnt/linux
  - cp /mnt/cdrom/kernel /mnt/linux

- Let's now write a lilo.conf file for /sbin/lilo to rewrite our MBR. It must be able to boot Windows on our hda1 partition, and Linux on the second one, with a (customizable but not customized here) text menu. Command: "vi /etc/lilo.conf" (i to insert text, :wq to save and exit). My lilo.conf file :

```
boot = /dev/hda
map = /mnt/dos/map
prompt
timeout = 150
vga = normal
lba32
delay = 30
read-only
large-memory
root = current

other = /dev/hda1
label = windows

image = /mnt/linux/kernel append=" vga=normal devfs=nomount pxe
ramdisk_size=33000 load_ramdisk=1 init=/linuxrc prompt_ramdisk=0
initrd=initrd.gz root=/dev/ram0 rw noapic nolapic lba
combined_mode=libata ide0=noprobe nomce pci=nommconf pci=nomsi
irqpoll"
label = restoration
initrd = /mnt/linux/initrd.gz
```

- Rewrite the MBR. Command: "lilo". It should output :

```
Added windows *
Added restoration
```

- Optionally, you can create directories on your linux partition, to replicate the contents of the Partimage.zip file. But PING will always propose you to create an image, and you can store it on the root of the partition.

- You should now be able to boot your system and access PING if needed. You'll do so at least once, so to write on the linux partition the image of your windows system. Afterwards, if you don't want any more users to be able to boot this linux partition and do whatever they want with it, you still can customize the append="" section of your /etc/lilo.conf file (and rewrite the mbr). The optional parameters are described in the howto's annex.

- **Second method – writing lilo to the linux partition, and having Windows's boot sector boot optionally the linux restoration partition through the edition of the C:\boot.ini file :**
  - The procedure is almost similar to the first method.
  - The /etc/lilo.conf file will be slightly different :
    - First line will be: boot = /dev/hda2
    - You'll have to add a ignore-table entry.
    - We'll remove the Windows entry, and set the delay to 0 second.
    - So, your /etc/lilo.conf file should look like this :

```
boot = /dev/hda2
map = /mnt/dos/map
prompt
timeout = 150
vga = normal
lba32
delay = 0
read-only
large-memory
root = current
ignore-table

image = /mnt/linux/kernel append=" vga=normal devfs=nomount pxe
ramdisk_size=33000 load_ramdisk=1 init=/linuxrc prompt_ramdisk=0
initrd=initrd.gz root=/dev/ram0 rw noapic nolapic lba
combined_mode=libata ide0=noprobe nomce pci=nommconf pci=nomsi
irqpoll"
label = restoration
initrd = /mnt/linux/initrd.gz
```

  - When running the "lilo" command, you might get warnings, that can be safely ignored.
  - After running "lilo", we'll have to extract the boot sector that has been written into the /dev/hda2 partition, and store it as a file in our Windows partition. Command: "dd if=/dev/hda2 of=/mnt/dos/bootsect.lnx bs=512 count=1".
  - Finally, on our C:\boot.ini, we'll append a line to the [operating systems] paragraph so to call our new C:\bootsect.lnx sector : "echo "C:\\BOOTSECT.LNX="Restoration" >> /mnt/dos/boot.ini".

- **How can I use PING with VMWare's SCSI controllers ?**

  - You'll have to check the LSI box (*tks Tahussle*).

- **How can I add components to the PING image without modifying the initrd.gz file ?**
  - From version 2.01.11, you can add or modify files after the PING OS has booted. To do so, simply add to the root of your CD/DVD files called addon-xxxx.tar.gz. All addon-*.tar.gz found at this place will be untarred to the root of the OS.
  - For example, we have used this feature to provide the users with a PING ISO which contains the CLAM anti-virus. The initrd.gz file contained in the PING-2.01-WithAddons.iso image is exactly same as the one contained in the PING-2.01.iso image, but the file addon-clamav.tar.gz has been added to the WithAddons ISO.
  - If you plan to use this feature in a PXE environment, then the addon-*.tar.gz files should be stored in the directory that contains your image directories. Obviously, they will only be untarred after the network shares are mounted.
  - TIP: you might want to combine this addon feature with parameters Cmd_1, Cmd_2 and Cmd_3.

- **I wanted to save an image to my Maxtor USB-drive, but ping can only see the ordinary IDE and the SATA disks. I can see the USB drive in the bios.**
  - If your BIOS settings has an option for plug and play (PNP) operating system, set it to disable or off (*tks Ken*).

- **I'm working on creating a SYSPREPed image with a recovery partition on the hard drive that can be used in the event of a catastrophic failure of windows ( Which is about once every 5 - 6 months Very Happy .... or less ).**
  **However the people that will eventually be using my imaged machines are all braindead and at some point one of them will want to choose the restore option which as we all know will cause me major grief.**
  **I'm looking to have a prompt pop up for a password here. Does anyone know if this can be done..**
  - In the /etc/lilo.conf you need to add password=your_chosen_password into the entry for which ever image you want to password *(tks laoistom aka Thomas).*

- **Can I modify the contents of partimaged files (sda1.000 etc.) ?**

- These binary files cannot be mounted.
- However, you can add a file called sda1.zip to your image directory. After restoring the partimaged files, PING will mount the restored partition and unzip this archive to the root of its filesystem. You can also add several archives, by calling them sda1.000.zip, sda1.001.zip, and so on.
- Think of it as an easy way to update quickly an image without having to restore it, modify it, and re-build it.

- **Can I use sshfs to securely mount a remote share ?**
  - You won't see the sshfs option among the proposed protocols, because ssh supports several complex authentication ways.
  - Nevertheless, the possibility is given to you to do it yourself. To do so, add a addon-xxx.tar.gz file to your image directory, so to have your ssh keys added to the live filesystem, and use the Cmd_1 parameter to pass a shell command that will mount the share. The mount point must be "/mnt/smbfs".
  - Note: all files called addon-xxx.tar.gz and found on the root of a CD/DVD, or in the directory of an image, are untarred to the root of the live PING OS as soon as possible, which means: immediately in the case of a CD/DVD, and just after the share is mounted and the image chosen in the other case. So, obviously, you won't be able to use sshfs unless you deploy with a CD/DVD…