How to use PreUpgrade

(Redirected from PreUpgrade)

In other languages: English (en) Русский (ru) (https://fedoraproject.org/wiki/How_to_use_PreUpgrade/ru) [edit (https://fedoraproject.org/w/index.php?title=Template:Lang/How_to_use_PreUpgrade&action=edit)]

PreUpgrade is an application users run on a existing release, that resolves and downloads packages required to upgrade to a newer release of Fedora. While PreUpgrade downloads the necessary packages, users are free to continue using their systems. This gives an experience similar to a live upgrade. For additional information, please refer to the PreUpgrade feature page.

Upgrade to current release directly
Preupgrade provides an upgrade directly to the latest version of Fedora. It is not necessary to upgrade to intermediate versions. For example, it is possible to go from Fedora 12 to Fedora 14 directly.

Prepare the system

While preupgrade provides a generally hassle free upgrade experience. The following steps are recommended before proceeding.

1. *Backup* - Before performing any system maintenance, it is recommended that you back up all important data before proceeding.
2. *Update* - Apply available updates Fedora updates before proceeding. As the root user, issue the following command.
```
yum update
```
3. *Install* - As of Fedora 10, the preupgrade utility is included in a default Fedora install. However, the package can be installed manually using the `yum` command.
```
yum install preupgrade
```

Perform the upgrade

Typically, you will be prompted by PackageKit (https://admin.fedoraproject.org/pkgdb/acls/name/PackageKit) when upgrades are available for your system. However, if you choose to manually upgrade using `preupgrade`, the steps are listed below.

1. As the root user, run the command `preupgrade` to start the Preupgrade application. If you prefer a command line application, the command `preupgrade-cli` is available.
2. On the *Choose desired release* screen, select the Fedora release you want to upgrade to , and click the *Apply* button.
3. When all of the packages have downloaded, reboot your system to start the Fedora installer and upgrade to the next release.

Remote upgrade

`preupgrade` has a switch that allows a remote upgrade via VNC. If you're using preupgrade for a remote upgrade, you'll more than likely be dealing with a machine that has a static IP. This is handled via the `preupgrade` command:

```
preupgrade-cli --vnc[=password] --ip=[IPADDR] --netmask=[NETMASK] --gateway=[IPADDR] --dns=[DNSSERVER] "Fedora 13 (Goddard)"
```

Common post-upgrade tasks

After the upgrade is complete, additional steps are recommended to complete the process.

Removing unsupported packages

Some packages may no longer be supported by the new release. You may wish to remove these packages because they will no longer get security updates, and they may cause later conflicts with supported packages. These can be identified with the following command:

```
package-cleanup --orphans
```

Examine `.rpmsave` and `.rpmnew` files

After completing the upgrade process, you may notice file names ending with `.rpmsave` and `.rpmnew`. Don't be alarmed. The upgrade process will always preserve any locally modified configuration files. The file names ending with `.rpmsave` contain your local configuration changes. While the file names ending with `.rpmnew` represent the configuration file originally packages with the software.

You should examine all `.rpmsave` and `.rpmnew` files created by the upgrade. Depending on the differences, you may need to manually merge configuration files. You can locate all matching files using the `find` command.

```
find / -print | egrep "rpm(new|save)$"
```

You may wish to speed up repeat searches as you edit, by running the `updatedb` command first, and then using `locate` for subsequent searches.

```
updatedb
locate --regex "rpm(new|save)$"
```

Verify the upgrade

Run

```
 yum repolist
```

to confirm repository settings are correct. Then run

```
 yum distro-sync
```

to sync the packages with the versions in the repository.

Troubleshooting

Not enough space in /boot

Fedora 13 and above has a 500 MB default boot partition. The default `/boot` filesystem size of 200MB for previous releases can be a problem for users upgrading from that release. In many cases, the diskspace that is likely free is just enough for preupgrade to find enough space to download the installer to but not quite enough for it to run the installer and install the new kernel after reboot. This section outlines several tips that have been known to work. As always when performing administrative tasks, be sure to back-up any data before proceeding.

There are two basic methods to make preupgrade work in this situation. In the first method, you need to free up sufficient space for the installer to install the new kernel packages. In the second method, you need to temporarily fill up enough of the space on `/boot` to force preupgrade to download the installer after rebooting.

Method 1: Free up space

First, try to remove any kernel packages not currently in use on your system. The kernel-prune.py (http://skvidal.fedorapeople.org/misc/kernel-prune.py) script can be used to identify kernels that may be safely removed. If you choose to remove additional kernels, be prepared with installation media (http://fedoraproject.org/en/get-fedora) should you be unable to return to your previously installed system.

The installer will need approximately 26M of free space in /boot. Use the following command to determine the amount of free space in the /boot partition:

```
df -h /boot
```

To identify kernels that may be safely removed, run the following from a command line:

```
curl -O 'http://skvidal.fedorapeople.org/misc/kernel-prune.py'
chmod a+x kernel-prune.py
./kernel-prune.py
```

Now, to actually remove the kernel versions listed by the above command, run the following as root:

```
# PKGS=`./kernel-prune.py`
# echo $PKGS
# yum remove $PKGS
```

Next, adjust the number of reserved filesystem blocks using the command `tune2fs`. You'll first need to identify the block device for your `/boot` file system. In the example below, `/dev/sda1` is the block device for the `/boot` filesystem.

```
# mount | grep "/boot"
/dev/sda1 on /boot type ext4 (rw)
```

Now, adjust the number of reserved blocks for the `/boot` filesystem using the command `tune2fs`. Normally, a small amount of space on ext filesystem formatted partitions is 'reserved' and can only be used by the system administrator; this is to prevent an entirely full partition from rendering a system unbootable, and allow the administrator some space in which to work in order to clean up 'full' partitions. However, neither of this cases really applies to the `/boot` filesystem, so removing this reserved space is safe.

```
# tune2fs -r 0 /dev/sda1
```

Last, try removing unnecessary files from the `/boot` filesystem. This will largely depend on how your system is set up. Removing the incorrect files may result in a unbootable system. Some candidates for removal include `/boot/efi` and `/boot/grub/splash.xpm.gz`.

Method 2: Trick preupgrade into downloading the installer

This method requires you to have a wired connection to the internet during the install. If you are on wireless and cannot connect via an ethernet cable you will have to use Method 1 instead.

First, find out how much space is available on the `/boot` filesystem. `df` is the command you want for this:

```
$ df /boot
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/sda1               198337     30543    157554  17% /boot
```

Second, create a file that takes up enough space that preupgrade decides it cannot install stage2 now. Preupgrade needs approximately 120MB for the installer image so we'll make sure we have a bit less than 100MB. For the example filesystem, that means we need to fill up 60MB. Here's how to do that as root:

```
# dd if=/dev/zero of=/boot/preupgrade_filler bs=1024 count=61440
# df /boot
Filesystem           1K-blocks      Used Available Use% Mounted on
/dev/sda1               198337     92224     95873  50% /boot
```

Third, run preupgrade as normal. In the early stages, before downloading packages, preupgrade should tell you that there wasn't enough room to download part of the installer but it can download it after reboot if you have a wired connection to the network. You can click continue for that. When preupgrade is done don't reboot immediately. Instead, remove the `/boot/preupgrade_filler` file and make sure your computer is connected to the network via an ethernet cable. Then you can reboot.

```
# rm /boot/preupgrade_filler
```

Fourth, the computer should boot into the installer, connect to the internet via the ethernet cable and start downloading the stage2 installer image. Then it should continue upgrading as normal.

Upgrade does not install upon reboot

Explanation

If you have a multiboot configuration, the `/boot` menu that grub uses might be different from the `/boot` menu that preupgrade modifies. In this case, you will need to point grub to the appropriate file to complete upgrade upon reboot. If this is not done, once preupgrade has completed downloading and installing files, upon reboot no change will be visible. The system will simply reboot as to the older version.

Preupgrade boots into an upgrade kernel as an intermediate step. Once the system has been upgraded, preupgrade replaces the temporary upgrade kernel option with an option for the upgraded kernel. In other words, there are two modifications made to the bootloader: a temporary upgrade option, followed by an option that is permanent until the next upgrade.

The grub bootloader can be used to boot from its command line, or `/boot/grub/menu.lst` can be modified to create a boot menu option. (Example screenshot of grub boot menu. (http://en.wikipedia.org/wiki/File:GRUB_screenshot.png)) (For further detail on grub, refer to the grub manual. (http://www.gnu.org/software/grub/manual/grub.html))

Either grub option can be used for either preupgrade step. To be comprehensive, the following describes both the command line option and doing so by editing the `menu.lst` file.

However, since the upgrade should only be run once, and the upgraded system will probably require repeated booting, the most expedient approach is probably to manually boot the upgrade via the grub command line, then once the upgrade has completed, to add an option to `menu.lst` add the upgraded Fedora installation to the grub boot menu. This would correspond to using Method 1, Steps 1 - 3, followed by Method 2, Step 4.

Method 1: Boot Manually from Grub Command Line

STEP 1: Identify Partition Location

Identify the drive and partition of your Fedora `/boot` folder. (See Grub Naming Convention (http://www.gnu.org/software/grub/manual/html_node/Naming-convention.html) for details.) For example, if you installed Fedora entirely on the fourteenth partition of a second hard drive, `/boot` would be located on `root (hd1,13)`

STEP 2: Boot from Partition Location

Upon reboot, enter 'c' to enter the grub command line. Using the appropriate drive and partition numbers, enter the following commands:

```
root (hd1,13)
kernel /boot/upgrade/vmlinuz
initrd /boot/upgrade/initrd.img
boot
```

This will boot the upgrade installation.

STEP 3: Select Installation Image

The upgrade installation will then run an ncurses dialog. After selecting language and keyboard types, select `hard drive` for `installation method`. The next dialog box will require partition and directory information of the installation image. Select the partition from the drop-down menu. (Note that the numbering will be one off from the grub partition. In other words `root (hd1,13)` will appear as `/dev/sdf14`.) Finally, enter the location of the install image file: `/boot/upgrade/install.img`.

The installation will run normally at this point. After completing the upgrade, you will need to either boot the upgraded system manually by entering the newly installed `kernel` and `initrd.img` files at the grub command line, or add an entry to the `menu.lst` file. This step is detailed in the next section.

Method 2: Edit the Grub `menu.lst` File

As an alternative to entering the commands at the grub command prompt upon reboot, you can also edit grub's `menu.lst` file to add an option that will allow you to select booting to the upgrade process from grub's boot menu. Since the upgrade should only be run once, after you've upgraded you will then need to re-edit `menu.lst`, remove the upgrade boot menu option, and add a boot entry for the new kernel.

STEP 1: Identify Partition Location

Identify the drive and partition of your Fedora `/boot` folder. Grub Naming Convention (http://www.gnu.org/software/grub/manual/html_node/Naming-convention.html) . For example, if you installed Fedora entirely on the fourteenth partition of a second hard drive, `/boot` would be located on `root (hd1,13)`

STEP 2: Edit `menu.lst`

Locate and open the `/boot/grub/menu.lst` file that grub actually uses. If this file is on another partition, check the `/media` files.

Using the appropriate drive and partition information in the `root` line format, add the following entry to the `menu.lst` file:

```
title Fedora Upgrade
root (hd*,*)
```

```
kernel /boot/upgrade/vmlinuz
initrd /boot/upgrade/initrd.img
savedefault
boot
```

Save the file, and reboot. Select the `Fedora Upgrade` from the grub boot menu.

STEP 3: Select Installation Image

The upgrade installation will then run an ncurses dialog. After selecting language and keyboard types, select `hard drive` for `installation method`. The next dialog box will require partition and directory information of the installation image. Select the partition from the drop-down menu. (Note that the numbering will be one off from the grub partition. In other words `root (hd1,13)` will appear as `/dev/sdf14`.)

Finally, enter the location of the install image file: `/boot/upgrade/install.img`. The installation will run normally at this point.

STEP 4: Cleanup `menu.lst`

After completing the upgrade, you will need to either boot the upgraded system manually by entering the newly installed `kernel` and `initrd.img` files at the grub command line, or by adding an entry to the `menu.lst` file.

The following is an example of a grub entry for a Fedora Core 10 installation, located on the fourteenth partition of the second hard drive.

```
title Fedora Core 10 (on /dev/sdb14)
root (hd1,13)
kernel /boot/vmlinuz-2.6.27.5-117.fc10.x86_64 ro quiet splash
initrd /boot/initrd-2.6.27.5-117.fc10.x86_64
savedefault
boot
```

Locate the updated `kernel` and `initrd` files located in the `/boot` folder of the Fedora partition, and create an entry with modified `title, kernel,` and `initrd` lines that will match those of the upgrade.

Finally, remove the upgrade boot entry from `menu.lst`.

Retrieved from "https://fedoraproject.org/wiki/How_to_use_PreUpgrade"
Categories: Documentation | How to | PreUpgrade