

Chapter 1

Using Kickstart

1.1 Understanding Kickstart Usage

The purpose of using kickstart is to automatically provide the settings we'd normally provide manually to the OS during installation. Now, to use kickstart, we need to provide the kickstart file (which contains the settings) at a place available to the installer.

Let us consider a minimal installation using *boot.iso*. Also, let us assume the kickstart file is named *myks.cfg*. Either USB keys (or any other storage media) or a server could host the file and provide it to the installer. It doesn't matter whether the installation is from a local media (e.g., usb key/ DVD) or a repository server. Once the installation starts, there is nothing more to do for the SysAdmin to do but to wait for it to finish, and the kickstart installation needs no manual intervention whatsoever!

1.2 Creating a Kickstart file

If the goal is to install just one server, it is easier to just install it manually, but for a number of servers (with the same configuration), using kickstart is much better. The kickstart file is passed to the installer.

The home directory or user *root* contains two kickstart files:

```
1 # ls -l /root
2 total 8
3 -rw-----, 1 root root 2183 Nov 25 09:09 anaconda-ks.cfg
4 -rw-r--r--, 1 root root 2214 Nov 25 09:47 initial-setup-ks.cfg
```

To use kickstart with only minor variations of the given configuration in either of those files, we could make a copy of the file and edit the attributes till they meet our requirements. However, a custom installation by creating a new kickstart file is also possible. For this, we need the *system-config-kickstart* utility (isn't installed by default).

The *system-config-kickstart* command launches the GUI tool that creates the required kickstart files. This GUI has the ability to ask for all the options asked during boot. However, some essential options like the option to create LVM is missing from the kickstart configuration. When done, the options can be saved to the file system.

1.2.1 Installation Scripts

The real power of kickstart comes from the fact that it is capable of running pre-installation and post-installation scripts.

1.3 Using the Kickstart file for Automatic installations

For network booting using a kickstart file, first we copy the kickstart file to a FTP server.

```
1 # scp -P 22 automate-install-ks.cfg somu@infraServer.somuVMnet.com:~
2 # ssh -p 22 root@infraServer.somuVMnet.com
3 # cp automate-install-ks.cfg /var/ftp/pub
4 # cd /var/ftp/pub
5 # chmod 644 automate-install-ks.cfg # Others need read access to be able to use the
   ↪ kickstart file.
6 # systemctl status vsftpd # Checking if the FTP server is functioning correctly!
```

Now on the machine where the OS is being installed, when the installer has fully loaded and provides the option to either directly install the OS, or test installation media and then install the OS, we need to press tab to set the installation options.

There, at the end, we need to type provide the location of the kickstart file on the network:

```
1 ks=ftp://infraServer.somuVMnet.com/ftp/pub/automated-install-ks.cfg
```

Then depending on which options were not given in the kickstart file, the installer may ask for some options, but if all the necessary details are given, then no manual intervention will be required at all!

In cases of data-centers and places where a large number of servers need to be installed, the use of a DVD is impractical and an installation server should be set up.

1.4 Using Kickstart files in fully automated data-centers

There are two servers here - one on the left on which the OS will be installed, and the other on the right, the installation server, which contains the boot image and installer. The server on which the OS will be installed performs a PXE boot (read as *pixie* boot), which is essentially booting from an installation image on the network through the NIC, while a server (installation server) is waiting for it.

On the installation server there is a **DHCP** (Dynamic Host Configuration Protocol) server and a **TFTP** (Trivial File Transfer Protocol). The DHCP server hands out IP addresses and the TFTP server makes the boot image available for download and installation. This boot image is received on the server on which the OS will be installed, and the installer is loaded. On a normal network installation this is where the network's role would end.

