

---

# Table of Contents

Introduction	1.1
101.1. Determine and configure hardware settings	1.2
101.2. Boot the system	1.3
101.3. Change runlevels and shutdown or reboot system	1.4
102.1. Design hard disk layout	1.5
102.2. Install a boot manager	1.6
102.3. Manage shared libraries	1.7
102.4. Use Debian package management	1.8
102.5. Use RPM and YUM package management	1.9
103.1. Work on the command line	1.10
103.2. Process text streams using filters	1.11
103.3. Perform basic file management	1.12
103.4. Use streams, pipes and redirects	1.13
103.5. Create, monitor and kill processes	1.14
103.6. Modify process execution priorities	1.15
103.7. Search text files using regular expressions	1.16
103.8. Perform basic file editing operations using vi	1.17
104.1. Create partitions and filesystems	1.18
104.2. Maintain the integrity of filesystems	1.19
104.3. Control mounting and unmounting of filesystems	1.20
104.4. Manage disk quotas	1.21
104.5. Manage file permissions and ownership	1.22
104.6. Create and change hard & symbolic links	1.23
104.7. Find system files and place files in the correct location	1.24
105.1. Customize and use the shell environment	1.25
105.2. Customize or write simple scripts	1.26
105.3. SQL data management	1.27
106.1. Install and configure X11	1.28
106.2. Setup a display manager	1.29
106.3. Accessibility	1.30

---

107.1. Manage user and group accounts and related system files	1.31
107.2. Automate system administration tasks by scheduling jobs	1.32
107.3. Localisation and internationalisation	1.33
108.1. Maintain system time	1.34
108.2. System logging	1.35
108.3. Mail Transfer Agent (MTA) basics	1.36
108.4. Manage printers and printing	1.37
109.1. Fundamentals of internet protocols	1.38
109.2. Basic network configuration	1.39
109.3. Basic network troubleshooting	1.40
109.4. Configure client side DNS	1.41
110.1. Perform security administration tasks	1.42
110.2. Setup host security	1.43
110.3. Securing data with encryption	1.44

# LPIC 1 study guide in plain English

This book is originally released as a part of a series of free video tutorials for LPIC1 created by Jadi (in Persian language). But it is also possible to use it as a reference for self-study to prepare for LPIC1-101 & LPIC-102 exam.

If you want to download Farsi LPIC 101 & LPIC 102 exam videos [use this torrent magnet link](#)

Latest versions of this book is always available at <http://j.mp/jadilpic1>

I've tried to write a short, easy to understand but still comprehensive exam guide. I'm not a native English speaker be ready for some possible language flaws.

Enjoy!

- Personal site: [Jadi.ir](http://Jadi.ir)
- Persian Weblog: [jadi.net](http://jadi.net)
- twitter: [@jadi](https://twitter.com/jadi)

Donations? <http://jadi.ir/support-donate/> :D

# 101.1 Determine and configure hardware settings

*Weight: 2*

Candidates should be able to determine and configure fundamental system hardware.

## Objectives

- Enable and disable integrated peripherals.
  - Configure systems with or without external peripherals such as keyboards.
  - Differentiate between the various types of mass storage devices.
  - Set the correct hardware ID for different devices, especially the boot device.
  - Know the differences between coldplug and hotplug devices.
  - Determine hardware resources for devices.
  - Tools and utilities to list various hardware information (e.g. `lsusb`, `lspci`, etc.)
  - Tools and utilities to manipulate USB devices
  - Conceptual understanding of `sysfs`, `udev`, `hald`, `dbus`
- 
- `/sys`
  - `/proc`
  - `/dev`
  - `modprobe`
  - `lsmod`
  - `lspci`
  - `lsusb`

## Find out about the hardware

### HAL

**HAL** is Hardware Abstraction Layer. It abstracts your hardware details from you, say any first network card will be `eth0`. This way Linux will see any hardware as an *standard* hardware and you will be able to replace the hardware easily.

### dbus

A line like a bus that connects all parts of the OS to each other. dbus lets different parts of the system to communicate with each other. For example, when you install a USB into your computer, dbus lets GNOME know about it. Using dbus, hardware & software can talk with each other.

### udev

Supplies the software with the events and access info of devices and can handle rules.

There are a lot of devices in `/dev/` and if you plugin any device, it will have a file in `/dev` (say `/dev/sdb2`). **udev** lets you control what will be what in `/dev`. For example, you can use a rule to force your 8GB flash drive with one specific vendor to be `/dev/mybackup` all the time or you can tell it to copy all photos to your home directory as soon as your camera is connected.

### sysfs

The `/sys` directory is where **HAL** keeps its database of everything connected to the system.

```
jadi@funlife:~$ ls /sys
block bus class dev  devices firmware fs  hypervisor kernel module power
```

All block devices are at the `block` and `bus` directory has all the connected PCI, USB, serial, .. devices. Note that here in `sys` we have the devices based on their technology but `/dev/` is abstracted.

### proc directory

This is where kernel keeps its settings and properties. This directory is created on ram and files might have write accessible.

[Click here to download full PDF material](#)