

— LPI Certification —  
— General Linux 2 —  
(Study Notes)<sup>1 2</sup>

geoffrey hector robertson  
geoffrey@zip.com.au

September 22, 2002

<sup>1</sup>Copyright ©2002 Geoffrey Robertson. Permission is granted to make and distribute verbatim copies or modified versions of this document provided that this copyright notice and this permission notice are preserved on all copies under the terms of the GNU General Public License as published by the Free Software Foundation—either version 2 of the License or (at your option) any later version.

<sup>2</sup>RCS Id = Id: lpic.general-linux-1.notes.tex,v 1.2 2002/02/28 23:06:04 geoffrey Exp geoffrey



# Contents

|  |           |
|--|-----------|
| <b>Topic 105: Kernel</b>   | <b>13</b> |
| <b>Objective 105.1: Manage/Query kernel and kernel modules at runtime</b>                  | <b>13</b> |
| 1.1 Overview . . . . .   | 13        |
| 1.1.1 Weight: [] . . . . .   | 13        |
| 1.1.2 Statement of Objective: . . . . .  | 13        |
| 1.1.3 Key files, terms, and utilities include: . . . . .                                   | 13        |
| 1.1.4 Resources of Interest: . . . . .   | 13        |
| 1.2 Notes . . . . .  | 14        |
| 1.3 Lab . . . . .  | 14        |
| 1.4 Questions . . . . .  | 14        |
| <b>Objective 105.2: Reconfigure, build, and install a custom kernel and kernel modules</b> | <b>15</b> |
| 2.1 Overview . . . . .   | 15        |
| 2.1.1 Weight: [] . . . . .   | 15        |
| 2.1.2 Statement of Objective: . . . . .  | 15        |
| 2.1.3 Key files, terms, and utilities: . . . . .   | 15        |
| 2.1.4 Resources of Interest: . . . . .   | 15        |
| 2.2 Notes . . . . .  | 16        |
| 2.3 Lab . . . . .  | 16        |
| 2.3.1 Build a Debian Kernel . . . . .  | 16        |
| 2.4 Questions . . . . .  | 17        |
| <b>Topic 106: Boot, Initialization, Shutdown and Runlevels</b>                             | <b>21</b> |
| <b>Objective 106.1: Boot the system</b>  | <b>21</b> |
| 1.1 Overview . . . . .   | 21        |
| 1.1.1 Weight: [] . . . . .   | 21        |
| 1.1.2 Statement of Objective: . . . . .  | 21        |
| 1.1.3 Key files, terms, and utilities include: . . . . .                                   | 21        |
| 1.1.4 Resources of Interest: . . . . .   | 21        |
| 1.2 Notes . . . . .  | 22        |
| 1.3 Lab . . . . .  | 22        |
| 1.4 Questions . . . . .  | 22        |

|   |           |
|---|-----------|
| <b>Objective 106.2: Change runlevels and shutdown or reboot system</b>  | <b>23</b> |
| 2.1 Overview  | 23        |
| 2.1.1 Weight: []  | 23        |
| 2.1.2 Statement of Objective:   | 23        |
| 2.1.3 Key files, terms, and utilities include:                          | 23        |
| 2.1.4 Resources of Interest:  | 23        |
| 2.2 Notes   | 24        |
| 2.3 Lab   | 24        |
| 2.4 Questions   | 24        |
| <br>  |           |
| <b>Topic 107: Printing</b>  | <b>27</b> |
| <b>Objective 107.1: Manage printers and print queues</b>                | <b>27</b> |
| 1.1 Overview  | 27        |
| 1.1.1 Weight: []  | 27        |
| 1.1.2 Statement of Objective:   | 27        |
| 1.1.3 Key files, terms, and utilities include:                          | 27        |
| 1.1.4 Resources of Interest:  | 27        |
| 1.2 Notes   | 28        |
| 1.3 Lab   | 28        |
| 1.4 Questions   | 28        |
| <br>  |           |
| <b>Objective 107.3: Print files</b>                                     | <b>29</b> |
| 3.1 Overview  | 29        |
| 3.1.1 Weight: []  | 29        |
| 3.1.2 Statement of Objective:   | 29        |
| 3.1.3 Key files, terms, and utilities include:                          | 29        |
| 3.1.4 Resources of Interest:  | 29        |
| 3.2 Notes   | 30        |
| 3.3 Lab   | 30        |
| 3.4 Questions   | 30        |
| <br>  |           |
| <b>Objective 107.4: Install and configure local and remote printers</b> | <b>31</b> |
| 4.1 Overview  | 31        |
| 4.1.1 Weight: []  | 31        |
| 4.1.2 Statement of Objective:   | 31        |
| 4.1.3 Key files, terms, and utilities include:                          | 31        |
| 4.1.4 Resources of Interest:  | 31        |
| 4.2 Notes   | 32        |
| 4.3 Lab   | 32        |
| 4.4 Questions   | 32        |
| <br>  |           |
| <b>Topic 108: Documentation</b>   | <b>35</b> |
| <b>Objective 108.1: Use and manage local system documentation</b>       | <b>35</b> |
| 1.1 Overview  | 35        |
| 1.1.1 Weight: []  | 35        |
| 1.1.2 Statement of Objective:   | 35        |

|  |  |           |
|--|--|-----------|
| 1.1.3  | Key files, terms, and utilities include: . . . . .   | 35        |
| 1.1.4  | Resources of Interest: . . . . .                     | 35        |
| 1.2  | Notes . . . . .                                      | 36        |
| 1.3  | Lab . . . . .  | 36        |
| 1.4  | Questions . . . . .                                  | 36        |
| <b>Objective 108.2: Find Linux documentation on the Internet</b> |  | <b>37</b> |
| 2.1  | Overview . . . . .                                   | 37        |
| 2.1.1  | Weight: [] . . . . .                                 | 37        |
| 2.1.2  | Statement of Objective: . . . . .                    | 37        |
| 2.1.3  | Key files, terms, and utilities include: . . . . .   | 37        |
| 2.1.4  | Resources of Interest: . . . . .                     | 37        |
| 2.2  | Notes . . . . .                                      | 38        |
| 2.3  | Lab . . . . .  | 38        |
| 2.4  | Questions . . . . .                                  | 38        |
| <b>Objective 108.5: Notify Users on System-Related Issues</b>    |  | <b>39</b> |
| 5.1  | Overview . . . . .                                   | 39        |
| 5.1.1  | Weight: [] . . . . .                                 | 39        |
| 5.1.2  | Statement of Objective: . . . . .                    | 39        |
| 5.1.3  | Key files, terms, and utilities include: . . . . .   | 39        |
| 5.1.4  | Resources of Interest: . . . . .                     | 39        |
| 5.2  | Notes on /etc/issue and /etc/motd . . . . .          | 40        |
| 5.2.1  | Customize the Local Login screen with /etc/issue . . | 40        |
| 5.2.2  | man issue . . . . .                                  | 40        |
| 5.2.3  | man getty—embedded in the /etc/issue . . . . .       | 40        |
| 5.2.4  | Telnet uses /etc/issue.net . . . . .                 | 41        |
| 5.2.5  | Message of the Day—motd . . . . .                    | 41        |
| 5.2.6  | Manpage for motd . . . . .                           | 41        |
| 5.3  | Lab on /etc/issue and /etc/motd . . . . .            | 41        |
| 5.4  | Questions . . . . .                                  | 42        |
| <b>Topic 109: Shells, Scripting, Programming, Compiling</b>      |  | <b>45</b> |
| <b>Objective 109.1: Customize and use the shell environment</b>  |  | <b>45</b> |
| 1.1  | Overview . . . . .                                   | 45        |
| 1.1.1  | Weight: [] . . . . .                                 | 45        |
| 1.1.2  | Statement of Objective: . . . . .                    | 45        |
| 1.1.3  | Key files, terms, and utilities include: . . . . .   | 45        |
| 1.1.4  | Resources of Interest: . . . . .                     | 45        |
| 1.2  | Notes . . . . .                                      | 46        |
| 1.2.1  | Bash Configuration Files . . . . .                   | 46        |
| 1.2.2  | Bash Aliases . . . . .                               | 46        |
| 1.2.3  | Bash Functions . . . . .                             | 46        |
| 1.2.4  | Function Example . . . . .                           | 46        |
| 1.2.5  | Valid Function Definitions . . . . .                 | 47        |
| 1.2.6  | Invalid Function Definitions . . . . .               | 47        |
| 1.2.7  | Invalid Function Definitions . . . . .               | 47        |
| 1.2.8  | Example from Jeffrey Dean’s Nutshell Book . . . . .  | 47        |

|  |  |           |
|--|--|-----------|
| 1.3  | Lab                                      | 47        |
| 1.3.1  | Exercise                                 | 47        |
| 1.4  | Questions                                | 47        |
| <b>Objective 109.2: Customize or write simple scripts</b>  |  | <b>49</b> |
| 2.1  | Overview                                 | 49        |
| 2.1.1  | Weight: []                               | 49        |
| 2.1.2  | Statement of Objective:                  | 49        |
| 2.1.3  | Key files, terms, and utilities:         | 49        |
| 2.1.4  | Resources of Interest:                   | 49        |
| 2.2  | Notes                                    | 51        |
| 2.3  | Lab                                      | 51        |
| 2.4  | Questions                                | 51        |
| <b>Topic 111: Administrative Tasks</b>   |  | <b>55</b> |
| <b>Objective 111.1: Manage users and group accounts and related system files</b>                     |  | <b>55</b> |
| 1.1  | Overview                                 | 55        |
| 1.1.1  | Weight: []                               | 55        |
| 1.1.2  | Statement of Objective:                  | 55        |
| 1.1.3  | Key files, terms, and utilities include: | 55        |
| 1.1.4  | Resources of Interest:                   | 55        |
| 1.2  | Notes                                    | 56        |
| 1.3  | Lab                                      | 56        |
| 1.4  | Questions                                | 56        |
| <b>Objective 111.2: Tune the user environment and system environment variables</b>                   |  | <b>57</b> |
| 2.1  | Overview                                 | 57        |
| 2.1.1  | Weight: []                               | 57        |
| 2.1.2  | Statement of Objective:                  | 57        |
| 2.1.3  | Key files, terms, and utilities include: | 57        |
| 2.1.4  | Resources of Interest:                   | 57        |
| 2.2  | Notes                                    | 58        |
| 2.3  | Lab                                      | 58        |
| 2.4  | Questions                                | 58        |
| <b>Objective 111.3: Configure and use system log files to meet administrative and security needs</b> |  | <b>59</b> |
| 3.1  | Overview                                 | 59        |
| 3.1.1  | Weight: []                               | 59        |
| 3.1.2  | Statement of Objective:                  | 59        |
| 3.1.3  | Key files, terms, and utilities include: | 59        |
| 3.1.4  | Resources of Interest:                   | 59        |
| 3.2  | Notes                                    | 60        |
| 3.3  | Lab                                      | 60        |
| 3.4  | Questions                                | 60        |

|  |           |
|--|-----------|
| <b>Objective 111.4: Automate system administration tasks by scheduling jobs to run in the future</b> | <b>61</b> |
| 4.1 Overview . . . . .   | 61        |
| 4.1.1 Weight: [] . . . . .   | 61        |
| 4.1.2 Statement of Objective: . . . . .  | 61        |
| 4.1.3 Key files, terms, and utilities include: . . . . .   | 61        |
| 4.1.4 Resources of Interest: . . . . .   | 61        |
| 4.2 Notes . . . . .  | 62        |
| 4.3 Lab . . . . .  | 62        |
| 4.4 Questions . . . . .  | 62        |
| <b>Objective 111.5: Maintain an effective data backup strategy</b>                                   | <b>63</b> |
| 5.1 Overview . . . . .   | 63        |
| 5.1.1 Weight: [] . . . . .   | 63        |
| 5.1.2 Statement of Objective: . . . . .  | 63        |
| 5.1.3 Key files, terms, and utilities include: . . . . .   | 63        |
| 5.1.4 Resources of Interest: . . . . .   | 63        |
| 5.2 Notes . . . . .  | 64        |
| 5.2.1 Backup Overview . . . . .  | 64        |
| 5.2.2 Backup & Restore methods . . . . .   | 65        |
| 5.2.3 Software . . . . .   | 67        |
| 5.2.4 Rotation & off-site strategies . . . . .   | 67        |
| 5.3 Lab . . . . .  | 68        |
| 5.4 Questions . . . . .  | 68        |
| <b>Objective 111.6: Maintain system time</b>   | <b>69</b> |
| 6.1 Overview . . . . .   | 69        |
| 6.1.1 Weight: [] . . . . .   | 69        |
| 6.1.2 Statement of Objective: . . . . .  | 69        |
| 6.1.3 Key files, terms, and utilities include: . . . . .   | 69        |
| 6.1.4 Resources of Interest: . . . . .   | 69        |
| 6.2 Notes . . . . .  | 70        |
| 6.2.1 Display or Set System Date & Time: <code>date</code> . . . . .                                 | 70        |
| 6.2.2 The Hardware Clock: <code>hwclock</code> . . . . .   | 71        |
| 6.2.3 NTP - Network Time Protocol . . . . .  | 71        |
| 6.3 Lab . . . . .  | 74        |
| 6.4 Questions . . . . .  | 74        |
| <b>Topic 112: Networking Fundamentals</b>  | <b>77</b> |
| <b>Objective 112.1: Fundamentals of TCP/IP</b>   | <b>77</b> |
| 1.1 Overview . . . . .   | 77        |
| 1.1.1 Weight: [] . . . . .   | 77        |
| 1.1.2 Statement of Objective: . . . . .  | 77        |
| 1.1.3 Key files, terms, and utilities include: . . . . .   | 77        |
| 1.1.4 Resources of Interest: . . . . .   | 77        |
| 1.2 Notes . . . . .  | 78        |
| 1.3 Lab . . . . .  | 78        |
| 1.4 Questions . . . . .  | 78        |

|  |           |
|--|-----------|
| <b>Objective 112.3: TCP/IP configuration and troubleshooting</b>                 | <b>79</b> |
| 3.1 Overview   | 79        |
| 3.1.1 Weight: []   | 79        |
| 3.1.2 Statement of Objective:  | 79        |
| 3.1.3 Key files, terms, and utilities include:                                   | 79        |
| 3.1.4 Resources of Interest:   | 79        |
| 3.2 Notes  | 81        |
| 3.3 Lab  | 81        |
| 3.4 Questions  | 81        |
| <b>Objective 112.4: Configure Linux as a PPP client</b>                          | <b>83</b> |
| 4.1 Overview   | 83        |
| 4.1.1 Weight: []   | 83        |
| 4.1.2 Statement of Objective:  | 83        |
| 4.1.3 Key files, terms, and utilities include:                                   | 83        |
| 4.1.4 Resources of Interest:   | 83        |
| 4.2 Notes  | 84        |
| 4.3 Lab  | 84        |
| 4.4 Questions  | 84        |
| <b>Topic 113: Networking Services</b>  | <b>87</b> |
| <b>Objective 113.1: Configure and manage inetd, xinetd, and related services</b> | <b>87</b> |
| 1.1 Overview   | 87        |
| 1.1.1 Weight: []   | 87        |
| 1.1.2 Statement of Objective:  | 87        |
| 1.1.3 Key files, terms, and utilities include:                                   | 87        |
| 1.1.4 Resources of Interest:   | 87        |
| 1.2 Notes  | 88        |
| 1.3 Lab  | 88        |
| 1.4 Questions  | 88        |
| <b>Objective 113.2: Operate and perform basic configuration of sendmail</b>      | <b>89</b> |
| 2.1 Overview   | 89        |
| 2.1.1 Weight: []   | 89        |
| 2.1.2 Statement of Objective:  | 89        |
| 2.1.3 Key files, terms, and utilities include:                                   | 89        |
| 2.1.4 Resources of Interest:   | 89        |
| 2.2 Notes  | 90        |
| 2.3 Lab  | 90        |
| 2.4 Questions  | 90        |
| <b>Objective 113.3: Operate and perform basic configuration of Apache</b>        | <b>91</b> |
| 3.1 Overview   | 91        |
| 3.1.1 Weight: []   | 91        |
| 3.1.2 Statement of Objective:  | 91        |
| 3.1.3 Key files, terms, and utilities include:                                   | 91        |
| 3.1.4 Resources of Interest:   | 91        |



|   |  |           |
|---|--|-----------|
| 3.2   | Notes . . . . .  | 92        |
| 3.3   | Lab . . . . .  | 92        |
| 3.4   | Questions . . . . .  | 92        |
| <b>Objective 113.4: Properly manage the NFS, smb, and nmb daemons</b> |  | <b>93</b> |
| 4.1   | Overview . . . . .   | 93        |
| 4.1.1   | Weight: [] . . . . .   | 93        |
| 4.1.2   | Statement of Objective: . . . . .  | 93        |
| 4.1.3   | Key files, terms, and utilities include: . . . . .                               | 93        |
| 4.1.4   | Resources of Interest: . . . . .   | 93        |
| 4.2   | Notes . . . . .  | 94        |
| 4.3   | Lab . . . . .  | 94        |
| 4.4   | Questions . . . . .  | 94        |
| <b>Objective 113.5: Setup and Configure Basic DNS Services</b>        |  | <b>95</b> |
| 5.1   | Overview . . . . .   | 95        |
| 5.1.1   | Weight: [] . . . . .   | 95        |
| 5.1.2   | Statement of Objective: . . . . .  | 95        |
| 5.1.3   | Key files, terms, and utilities include: . . . . .                               | 95        |
| 5.1.4   | Resources of Interest: . . . . .   | 95        |
| 5.2   | Notes . . . . .  | 96        |
| 5.2.1   | Setup and Configure basic DNS services . . . . .                                 | 96        |
| 5.2.2   | Setup and Configure basic DNS services . . . . .                                 | 96        |
| 5.2.3   | DNS - DOMAIN NAME SERVICE . . . . .  | 96        |
| 5.2.4   | DNS - DOMAIN NAME SERVICE . . . . .  | 96        |
| 5.2.5   | RESOLVING A NAME . . . . .   | 97        |
| 5.2.6   | The <code>nsswitch.conf</code> file . . . . .                                    | 97        |
| 5.2.7   | The <code>nsswitch.conf</code> file . . . . .                                    | 97        |
| 5.2.8   | An example <code>nsswitch</code> file: . . . . .                                 | 98        |
| 5.2.9   | The <code>resolv.conf</code> file . . . . .                                      | 98        |
| 5.2.10  | BIND - Berkley Internet Name Domain . . . . .                                    | 98        |
| 5.2.11  | BIND Configuration . . . . .   | 99        |
| 5.2.12  | An Example Config file: . . . . .  | 99        |
| 5.2.13  | Zone files: . . . . .  | 99        |
| 5.2.14  | Zone Records: . . . . .  | 100       |
| 5.2.15  | Example Forward file <code>/var/named/aes.zone</code> . . . . .                  | 100       |
| 5.2.16  | Example reverse file <code>/var/named/1.27.10.in-addr.arpa.zone</code> . . . . . | 101       |
| 5.2.17  | Configuring a Caching only Nameserver . . . . .                                  | 101       |
| 5.2.18  | Testing DNS . . . . .  | 101       |
| 5.2.19  | <code>nslookup</code> . . . . .  | 102       |
| 5.2.20  | <code>dig</code> . . . . .   | 102       |
| 5.2.21  | <code>host</code> . . . . .  | 102       |
| 5.2.22  | Exercise: . . . . .  | 103       |
| 5.3   | Lab . . . . .  | 103       |
| 5.4   | Questions . . . . .  | 103       |

|   |            |
|---|------------|
| <b>Objective 113.7: Set up secure shell (OpenSSH)</b>         | <b>105</b> |
| 7.1 Overview  | 105        |
| 7.1.1 Weight: []  | 105        |
| 7.1.2 Statement of Objective:                                 | 105        |
| 7.1.3 Key files, terms, and utilities include:                | 105        |
| 7.1.4 Resources of Interest:                                  | 105        |
| 7.2 Notes   | 106        |
| 7.3 Lab   | 106        |
| 7.4 Questions   | 106        |
| <br>  |            |
| <b>Topic 114: Security</b>                                    | <b>109</b> |
| <b>Objective 114.1: Perform security administration tasks</b> | <b>109</b> |
| 1.1 Overview  | 109        |
| 1.1.1 Weight: []  | 109        |
| 1.1.2 Statement of Objective:                                 | 109        |
| 1.1.3 Key files, terms, and utilities include:                | 109        |
| 1.1.4 Resources of Interest:                                  | 109        |
| 1.2 Notes   | 110        |
| 1.3 Lab   | 110        |
| 1.4 Questions   | 110        |
| <br>  |            |
| <b>Objective 114.2: Setup host security</b>                   | <b>111</b> |
| 2.1 Overview  | 111        |
| 2.1.1 Weight: []  | 111        |
| 2.1.2 Statement of Objective:                                 | 111        |
| 2.1.3 Key files, terms, and utilities include:                | 111        |
| 2.1.4 Resources of Interest:                                  | 111        |
| 2.2 Notes   | 112        |
| 2.3 Lab   | 112        |
| 2.4 Questions   | 112        |
| <br>  |            |
| <b>Objective 114.3: Setup user level security</b>             | <b>113</b> |
| 3.1 Overview  | 113        |
| 3.1.1 Weight: []  | 113        |
| 3.1.2 Statement of Objective:                                 | 113        |
| 3.1.3 Key files, terms, and utilities include:                | 113        |
| 3.1.4 Resources of Interest:                                  | 113        |
| 3.2 Notes   | 114        |
| 3.3 Lab   | 114        |
| 3.4 Questions   | 114        |
| <br>  |            |
| <b>List of topics</b>   | <b>115</b> |
| .1 Topics moved to General Linux 1                            | 116        |

# **Topic 105**

## **Kernel**



# Objective 105.1

## Manage/Query kernel and kernel modules at runtime

### 1.1 Overview

#### 1.1.1 Weight: []

#### 1.1.2 Statement of Objective:

Candidates should be able to manage and/or query a kernel and kernel loadable modules. This objective includes using command-line utilities to get information about the currently running kernel and kernel modules. It also includes manually loading and unloading modules as appropriate. It also includes being able to determine when modules can be unloaded and what parameters a module accepts. Candidates should be able to configure the system to load modules by names other than their file name.

#### 1.1.3 Key files, terms, and utilities include:

```
/lib/modules/kernel-version/modules.dep  
/etc/modules.conf & /etc/conf.modules  
depmod  
insmod  
lsmod  
rmmod  
modinfo  
modprobe  
uname
```

#### 1.1.4 Resources of Interest:

TBA

**1.2 Notes**

**1.3 Lab**

**1.4 Questions**

# Objective 105.2

## Reconfigure, build, and install a custom kernel and kernel modules

### 2.1 Overview

#### 2.1.1 Weight: []

#### 2.1.2 Statement of Objective:

Candidates should be able to customize, build, and install a kernel and kernel loadable modules from source This objective includes customizing the current kernel configuration, building a new kernel, and building kernel modules as appropriate. It also includes installing the new kernel as well as any modules, and ensuring that the boot manager can locate the new kernel and associated files (generally located under /boot, see objective 1.102.2 for more details about boot manager configuration).

#### 2.1.3 Key files, terms, and utilities:

```
/usr/src/linux/*
/usr/src/linux/.config
/lib/modules/kernel-version/*
/boot/*
make
make targets: config, menuconfig, xconfig, oldconfig, modules, install, modules_install, depmod
```

#### 2.1.4 Resources of Interest:

TBA

## 2.2 Notes

## 2.3 Lab

### 2.3.1 Build a Debian Kernel

#### Catch a fresh kernel

- Kernel sources are large so if you already have a recent kernel it is best to obtain patches to bring it up to date.
- Fresh kernels are to be had from

`http://www.kernel.org`

be sure to use a mirror near you.

- Place your kernel in a suitable place:
  - Under `/usr/src/` Note that this directory has group owner of `src`, so make yourself a member of the `src` group:

```
# usermod -G src fred ↵
```

```
- /tmp
```

```
- /home/fred
```

#### apt-get the Necessary Tools

```
# apt-get install debhelper modutils libncurses5-dev build-essential fakeroot ↵
```

#### Edit `/etc/pkg-source.conf`

```
# vi /etc/kernel-pkg.conf ↵
```

#### Unarchive the Source

```
$ cd /usr/src/ ↵
```

```
$ tar zxvf linux-2.5.34.tar.gz ↵
```

```
$ cd linux-2.5.34 ↵
```

#### Use Your Existing `.config` as a Starting Place

```
$ cp /boot/config-2.4.18-386 .config ↵ #YMMV
```

#### Configure Your Kernel

```
$ make oldconfig ↵
```

```
$ make xconfig ↵ #or menuconfig or config
```



## 2.4 Questions



## **Topic 106**

# **Boot, Initialization, Shutdown and Runlevels**



# Objective 106.1

## Boot the system

### 1.1 Overview

#### 1.1.1 Weight: []

#### 1.1.2 Statement of Objective:

Candidates should be able to guide the system through the booting process. This includes giving commands to the boot loader and giving options to the kernel at boot time, and checking the events in the log files.

#### 1.1.3 Key files, terms, and utilities include:

```
dmesg /var/log/messages  
/etc/conf.modules or /etc/modules.conf  
LILO GRUB
```

#### 1.1.4 Resources of Interest:

TBA

**1.2 Notes**

**1.3 Lab**

**1.4 Questions**

# Objective 106.2

## Change runlevels and shutdown or reboot system

### 2.1 Overview

#### 2.1.1 Weight: []

#### 2.1.2 Statement of Objective:

Candidates should be able to manage the runlevel of the system. This objective includes changing to single user mode, shutdown or rebooting the system. Candidates should be able to alert users before switching runlevel, and properly terminate processes. This objective also includes setting the default runlevel.

#### 2.1.3 Key files, terms, and utilities include:

```
shutdown  
init  
/etc/inittab
```

#### 2.1.4 Resources of Interest:

TBA

**2.2 Notes**

**2.3 Lab**

**2.4 Questions**



# **Topic 107**

## **Printing**



# Objective 107.1

## Manage printers and print queues

### 1.1 Overview

#### 1.1.1 Weight: []

#### 1.1.2 Statement of Objective:

The candidate should be able to manage print queues and user print jobs. This objective includes monitoring print server and user print queues and troubleshooting general printing problems

#### 1.1.3 Key files, terms, and utilities include:

lpc  
lpq  
lprm  
lpr  
/etc/printcap

#### 1.1.4 Resources of Interest:

TBA

**1.2 Notes**

**1.3 Lab**

**1.4 Questions**

# Objective 107.3

## Print files

### 3.1 Overview

#### 3.1.1 Weight: []

#### 3.1.2 Statement of Objective:

Candidates should be able to manage print queues and manipulate print jobs. This objective includes adding and removing jobs from configured printer queues and converting text files to postscript for printing.

#### 3.1.3 Key files, terms, and utilities include:

lpr  
lpq  
mpage

#### 3.1.4 Resources of Interest:

TBA

**3.2 Notes****3.3 Lab****3.4 Questions**

# Objective 107.4

## Install and configure local and remote printers

### 4.1 Overview

#### 4.1.1 Weight: []

#### 4.1.2 Statement of Objective:

Candidate should be able to install a printer daemon, install and configure a print filter (e.g.: `apsfilter`, `magicfilter`). This objective includes making local and remote printers accessible for a Linux system, including postscript, non-postscript, and Samba printers.

#### 4.1.3 Key files, terms, and utilities include:

```
lpd
/etc/printcap
/etc/apsfilter/*
/var/lib/apsfilter/*/
/etc/magicfilter/*/
/var/spool/lpd/*/
```

#### 4.1.4 Resources of Interest:

TBA

**4.2 Notes****4.3 Lab****4.4 Questions**



# **Topic 108**

## **Documentation**



# **Objective 108.1**

## **Use and manage local system documentation**

### **1.1 Overview**

#### **1.1.1 Weight: []**

#### **1.1.2 Statement of Objective:**

Candidates should be able to use and administer the man facility and the material in `/usr/share/doc/`. This objective includes finding relevant man pages, searching man page sections, finding commands and man pages related to them, and configuring access to man sources and the man system. It also includes using system documentation stored in `/usr/share/doc/` and determining what documentation to keep in `/usr/share/doc/`.

#### **1.1.3 Key files, terms, and utilities include:**

#### **1.1.4 Resources of Interest:**

TBA

**1.2 Notes**

**1.3 Lab**

**1.4 Questions**

# **Objective 108.2**

## **Find Linux documentation on the Internet**

### **2.1 Overview**

#### **2.1.1 Weight: []**

#### **2.1.2 Statement of Objective:**

Candidates should be able to find and use Linux documentation. This objective includes using Linux documentation at sources such as the Linux Documentation Project (LDP), vendor and third-party websites, newsgroups, newsgroup archives, and mailing lists.

#### **2.1.3 Key files, terms, and utilities include:**

#### **2.1.4 Resources of Interest:**

TBA

**2.2 Notes****2.3 Lab****2.4 Questions**

# **Objective 108.5**

## **Notify Users on System-Related Issues**

### **5.1 Overview**

#### **5.1.1 Weight: []**

#### **5.1.2 Statement of Objective:**

Candidates should be able to notify the users about current issues related to the system. This objective includes automating the communication process, e.g. through logon messages.

#### **5.1.3 Key files, terms, and utilities include:**

```
/etc/issue  
/etc/issue.net  
/etc/motd
```

#### **5.1.4 Resources of Interest:**

TBA

## 5.2 Notes on `/etc/issue` and `/etc/motd`

### 5.2.1 Customize the Local Login screen with `/etc/issue`

- The login screen for RH73 looks like this:

```
Red Hat Linux release 7.3 (Valhalla)
Kernel 2.4.18-3 on an i686
```

```
login:
```

- The `getty` process spawned by `init` presents the contents of the file `/etc/issue` and provides a `login:` prompt for the user.
- The file `/etc/issue`

```
$ cat /etc/issue ←
Red Hat Linux release 7.3 (Valhalla)
Kernel \r on an \m
```

### 5.2.2 `man issue`

```
ISSUE(5)                Linux Programmer's Manual                ISSUE(5)

NAME
    issue - pre-login message and identification file

DESCRIPTION
    The file /etc/issue is a text file which contains a message or
    system identification to be printed before the login prompt. It
    may contain various @char and \char sequences, if supported by
    getty(1).

FILES
    /etc/issue

SEE ALSO
    getty(1), motd(5)

Linux
1993-07-24                ISSUE(5)
```

### 5.2.3 `man getty`—embedded in the `/etc/issue`

```
\d    insert current day (localtime),
\l    insert line on which mingetty is running,
\m    inserts machine architecture (uname -m),
\n    inserts machine's network node hostname (uname -n),
\o    inserts domain name,
\r    inserts operating system release (uname -r),
\t    insert current time (localtime),
\s    inserts operating system name,
\u    resp. \U the current number of users which are currently
      logged in. \U inserts "n users", where as \u only inserts "n".
\v    inserts operating system version (uname -v).
```



### 5.2.4 Telnet uses /etc/issue.net

The following sequences are supported by telnetd:

```
%l      - show the current tty
%h, %n  - show the system node name (FQDN)
%D, %o  - show the name of the NIS domain
%d, %t  - show the current time and date
%s      - show the name of the operating system
%m      - show the machine (hardware) type
%r      - show the operating system release
%v      - show the operating system version
%%      - display a single '%' character
```

### 5.2.5 Message of the Day—motd

#### Login Sequence

- Contents of /etc/issue is displayed.
- `getty` displays the login prompt.
- /bin/login handles the login process.
  - User is authorized.
  - Contents of /etc/motd is displayed.
  - The login shell is executed.

### 5.2.6 Manpage for motd

```
MOTD (5)                Linux Programmer's Manual                MOTD (5)

NAME
    motd - message of the day

DESCRIPTION
    The contents of /etc/motd are displayed by login(1) after a successful login but just before it executes the login shell.

    The abbreviation "motd" stands for "message of the day", and this file has been traditionally used for exactly that (it requires much less disk space than mail to all users).

FILES
    /etc/motd

SEE ALSO
    login(1), issue(5)

Linux                1992-12-29                MOTD (5)
```

### 5.3 Lab on `/etc/issue` and `/etc/motd`

1. Before altering any of the system files back them up:

```
# cp issue issue.orig ↵  
# cp issue.net issue.net.orig ↵  
# cp motd motd.orig ↵
```

2. Edit `/etc/issue` and place a suitable message there. Try out some of the embedded codes from the man page such as `\t`. Login from another virtual terminal and check your results.
3. Edit `/etc/issue.net` and place a suitable message there. Try out some of the embedded codes from the man page such as `%t`. Telnet to you host from another system and check your results.
4. Edit `/etc/motd` and place a suitable message there. Login from another virtual terminal and check your results.
5. Replace the system files with the original versions:

```
# cp issue.orig issue ↵  
# cp issue.net.orig issue.net ↵  
# cp motd.orig motd ↵
```

6. Login and check that the systems are “as installed”.

### 5.4 Questions

## **Topic 109**

# **Shells, Scripting, Programming, Compiling**



# Objective 109.1

## Customize and use the shell environment

### 1.1 Overview

#### 1.1.1 Weight: []

#### 1.1.2 Statement of Objective:

Candidate should be able to customize shell environments to meet users' needs. This objective includes setting environment variables (e.g. PATH) at login or when spawning a new shell. It also includes writing bash functions for frequently used sequences of commands.

#### 1.1.3 Key files, terms, and utilities include:

```
~/ .bash_profile
~/ .bash_login
~/ .profile
~/ .bashrc
~/ .bash_logout
~/ .inputrc
function (Bash built-in command)
export
env
set (Bash built-in command)
unset (Bash built-in command)
```

#### 1.1.4 Resources of Interest:

TBA

## 1.2 Notes

### 1.2.1 Bash Configuration Files

- When a user logs in to a `bash` shell the following configuration files are usually executed:

`/etc/profile` System wide profile, common to all users and shells

`~/.bash_profile` Executed after `/etc/profile` at login

`~/.bashrc` Executed after  `~/.bash_profile` at login

- Note  `~/.bashrc` is executed when any new `bash` shell is spawned

### 1.2.2 Bash Aliases

- 

### 1.2.3 Bash Functions

- Functions work similarly to aliases but allow more complex constructions.
- They have the following syntax:

```
$ [ function ] NAME () { COMMAND_LIST; } ←
```

- Where

`function` Optional tag

`NAME ()` The name of the function

`COMMAND_LIST` The body of the function

- Functions may be stored in  `~/.bashrc`

### 1.2.4 Function Example

- This simple function prints the current working directory and the list of files in it:

```
$ function look () { pwd; ls; } ←
```

- This function would be used like this:

```
$ look ←
/home/geoffrey/lpic/general-linux-2/notes
CVS      _whizzy_gl2.notes.fmt
_whizzy_gl2.notes.pag
```

### 1.2.5 Valid Function Definitions

- `$ function look() { pwd; ls; }`
- `$ function look { pwd; ls; }`
- `$ look() { pwd; ls; }`
- `$ look()  
> {  
> pwd;  
> ls;  
> }`

### 1.2.6 Invalid Function Definitions

- `$ function look() pwd; ls;`
- `$ look() { pwd; ls }`
- `$ function look() {pwd; ls;}`

### 1.2.7 Invalid Function Definitions

- `$ function look() pwd; ls;`
- `$ look() { pwd; ls }`
- `$ function look() {pwd; ls;}`

### 1.2.8 Example from Jeffrey Dean's Nutshell Book

- `$ laps () { ↵  
> ls -l $1  
> ps aux | grep '/usr/bin/basename $1'  
> }`
- Use the `laps()` function:

```
$ laps /usr/sbin/sshd ↵
-rwxr-xr-x  1 root    root      276200 Jun 29 01:28 /usr/sbin/sshd
root      255  0.0  0.3  2792 1216 ?        S    Aug31   0:00 /usr/sbin/sshd
geoffrey 1187  0.0  0.1  1332  424 pts/1    R    14:39   0:00 grep sshd
```

## 1.3 Lab

### 1.3.1 Exercise

## 1.4 Questions





# Objective 109.2

## Customize or write simple scripts

### 2.1 Overview

#### 2.1.1 Weight: []

#### 2.1.2 Statement of Objective:

Candidate should be able to customize existing scripts, or write simple new (ba)sh scripts. This objective includes using standard sh syntax (loops, tests), using command substitution, testing command return values, testing of file status, and conditional mailing to the superuser. This objective also includes making sure the correct interpreter is called on the first (!) line of scripts. This objective also includes managing location, ownership, execution and suid-rights of scripts.

#### 2.1.3 Key files, terms, and utilities:

```
while  
for  
test  
chmod
```

#### 2.1.4 Resources of Interest:

1. Bash Programming Introduction (LDP HOWTO)

<http://www.linux.org/docs/ldp/howto/Bash-Prog-Intro-HOWTO.html>

2. Linux Shells by Example  
*by Ellie Quigley*  
ISBN 0-13-0141711-7  
Prentice Hall

3. LINUX & UNIX Shell Programming

*David Tansley*

ISBN 0-201-67472-6

ADDISON-WESLEY

**2.2 Notes**

**2.3 Lab**

**2.4 Questions**



## **Topic 111**

### **Administrative Tasks**



# Objective 111.1

## Manage users and group accounts and related system files

### 1.1 Overview

#### 1.1.1 Weight: []

#### 1.1.2 Statement of Objective:

Candidate should be able to add, remove, suspend and change user accounts. Tasks include to add and remove groups, to change user/group info in passwd/group databases. The objective also includes creating special purpose and limited accounts.

#### 1.1.3 Key files, terms, and utilities include:

```
chageg passwd groupadd groupdel groupmod grpconv grpunconv passwd  
pwconv pwunconv useradd userdel usermod /etc/passwd /etc/shadow  
/etc/group /etc/gshadow
```

#### 1.1.4 Resources of Interest:

**The Linux System Administrators' Guide** Chapter 9 - Managing User Accounts

**Manpages** useradd usermod userdel groupadd groupmod groupdel useradd  
passwd chage

**1.2 Notes**

**1.3 Lab**

**1.4 Questions**



# Objective 111.2

## Tune the user environment and system environment variables

### 2.1 Overview

#### 2.1.1 Weight: []

#### 2.1.2 Statement of Objective:

Candidate should be able to modify global and user profiles. This includes setting environment variables, maintaining skel directories for new user accounts and setting command search path with the proper directory.

#### 2.1.3 Key files, terms, and utilities include:

```
env export set unset /etc/profile /etc/skel
```

#### 2.1.4 Resources of Interest:

TBA

**2.2 Notes**

**2.3 Lab**

**2.4 Questions**

## **Objective 111.3**

# **Configure and use system log files to meet administrative and security needs**

### **3.1 Overview**

#### **3.1.1 Weight: []**

#### **3.1.2 Statement of Objective:**

Candidate should be able to configure system logs. This objective includes managing the type and level of information logged, manually scanning log files for notable activity, monitoring log files, arranging for automatic rotation and archiving of logs and tracking down problems noted in logs.

#### **3.1.3 Key files, terms, and utilities include:**

#### **3.1.4 Resources of Interest:**

TBA logrotate tail -f /etc/syslog.conf /var/log/\*

**3.2 Notes**

**3.3 Lab**

**3.4 Questions**

# Objective 111.4

## Automate system administration tasks by scheduling jobs to run in the future

### 4.1 Overview

#### 4.1.1 Weight: []

#### 4.1.2 Statement of Objective:

Candidate should be able to use `cron` or `anacron` to run jobs at regular intervals and to use `at` to run jobs at a specific time. Task include managing `cron` and `at` jobs and configuring user access to `cron` and `at` services.

#### 4.1.3 Key files, terms, and utilities include:

```
at atq crontab /etc/anacrontab /etc/at.deny /etc/at.allow
    /etc/crontab /etc/cron.allow /etc/cron.deny /var/spool/cron/*
```

#### 4.1.4 Resources of Interest:

TBA

**4.2 Notes****4.3 Lab****4.4 Questions**

# Objective 111.5

## Maintain an effective data backup strategy

### 5.1 Overview

#### 5.1.1 Weight: []

#### 5.1.2 Statement of Objective:

Candidate should be able to plan a backup strategy and backup filesystems automatically to various media. Tasks include dumping a raw device to a file or vice versa, performing partial and manual backups, verifying the integrity of backup files and partially or fully restoring backups.

#### 5.1.3 Key files, terms, and utilities include:

`cpio dd dump restore tar`

#### 5.1.4 Resources of Interest:

TBA

## 5.2 Notes

### 5.2.1 Backup Overview

Prepared by Grant Parnell

Decide what data is important and how long you can do without it.

- Is this used 24 x 7 or just business hours?
- During business hours how long can you do without it? 4 hours, 30 minutes, 5 minutes?
- How up-to-date is it required to get you running in an emergency?
- Are you backing up for archival or high availability or espionage?

#### Examples of Data

**Static:** Configurations of running servers. You need these 24x7 but they don't change much.

**Databases / Transactions - financial & otherwise:** These are updated frequently and need to balance. Associated with these are logs and duplication and other means of rollback and integrity checking. With databases it's often a good idea to dump them in a good portable format, especially if the inbuilt format is not cross platform or cross version compatible. EG `'mysqldump mydata >mydata.dump'` will give you a text file which can be used on most mysql versions and possibly adapted to other database packages.

**Logs:** People don't tend to read them unless something goes wrong in which case they're valuable. These need to be kept but don't need to be restored in a hurry.

**Home directories:** This is a mixed bag of everything but some policies could be instated to make the admin's life easier. EG Making specific sub-directories for things and assigning them different backup/restore priorities. Often the existence of a home directory is more important than the rest of the contents as it may make a user unable to login without it.

**Code repositories:** Programmers should be accustomed to doing regular backups anyway, they often need to revert to an old version to figure out what they broke. Any tools used such as CVS that have a central repository should be backed up almost as often as programmers commit code, at least once a day but they could probably cope with it being missing for half a day.

**High availability - read only:** Websites frequently used by your clients. They can contain dynamic data but customers don't update it. This sort of scenario lends itself to frequent replication to a backup server.



**High availability - interactive:** Taking a website again, this one might allow the customer to do such things as place orders. The website maintains some state information to allow building of an order. This is the most difficult, the state information can be stored in a replicated database. In the event of web server failure the other one comes into play and the customer may have to login again but the information is kept. (Otherwise complex designs and expensive hardware can be used to seamlessly migrate the state to the other webserver).

### Important Linux directories

```

/var/spool/mail - daily backup
/var/lib/mysql - databases - backup the dumps, and possibly
                 the binary.
/var/log ?      - from "don't care" to "backup daily"
/etc           - backup config changes
/home         - be selective, but if you can't backup daily.
/home/<user>/mail - contains the user's mail folders
                 (may also be 'Mail' or 'Maildir')
/home/<user>/.ssh - If you login using ssh keys only, this is a
                 must have.
/usr/local    - locally installed apps & data
                 Application specifics

```

## 5.2.2 Backup & Restore methods

### Copy the files to another directory

This is the poor mans backup and does not offer much peace of mind. It does protect against accidental deletion and corruption by users. One advantage is that it can be very quick for things such as log files. You can also keep multiple copies, one for every day of the week for example. See `/etc/logrotate.conf`.

### Backup to a standby partition

This has about the same level of peace of mind as the above. The backup partition can be left un-mounted after the backup. The backup is slower than the above but the restore operation can be quick. See also "Broken Mirror" method below.

### Backup to tape

This is probably the most common backup used in the commercial world. It's easy to backup the lot every day provided you have the tape capacity. If you don't, you become more selective as to what to backup. There's a variety of software to do this but there's 3 main basic systems. Tar, cpio and dump. Often commercial software uses these basic systems and provide for labelling and indexing as well as multi-server capability from a simple GUI. The reason for using the basic systems is you can restore from them if you have to.

### **Backup to standby disk**

This can offer peace of mind and a fairly cheap backup for people that don't require 24x7 service. Basically a removable drive bay houses another hard disk of similar capacity and the entire system is backed up. This can be done partition by partition or file by file using `dd`, `cpio` or `rsync`. Additional steps can be taken to ensure that the backup is also bootable. The backup drive should be removed once done and treated like a tape. The disadvantage here is that you most likely will need to power down the system twice for one backup. Alternately, if you have an external USB or fire-wire storage medium it becomes possible to do this without downtime.

### **Backup to CDROM/DVD**

Under linux (as far as I know) there's no software to directly write data without creating an image first. This means there must be sufficient space available. It would be possible to create a bootable CD with restore software and a compressed filesystem but I haven't seen this. It may be OK if you don't have a large filesystem or you have a DVD writer or you're not backing up everything.

### **RAID System**

Not strictly a backup but a RAID system can protect against hard drive failure by providing redundancy. Data is written simultaneously to 2 or more hard drives and can include parity information. It does not protect against corrupt databases and people removing files. It will corrupt and remove files equally well on all disks. Linux can do RAID in software very well but the ideal is a hardware solution involving hot swappable disks so they can be replaced whilst the system is fully running. A RAID system can mean the difference between going on-site at 3am and saying "Oh dear, we'll replace that first thing in the morning". Just ensure that you do have a replacement readily available and do not have to wait a week.

### **RAID Tape array**

In a similar manner to RAID 5 disks, data is written in parallel to 5 tape drives which increases throughput and data integrity.

### **Backup Server**

All of the methods discussed so far involve direct transfer from server to backup medium. If you have a number of servers it may not be practical to install backup devices on each. Another way is to remotely access the required medium directly (`/dev/rmt0`) but arbitration of access can be an issue. An increasingly popular way is to provide a super-server with a huge amount of disk space capable of holding everything required by the other servers. Transferring the data can happen at any time in either a batch or continuous process. A batch would be say backup a whole directory at once whereas a continuous operation might be transmitting log information or database updates. The backup server itself

may then employ any one or more methods to perform backups of itself, possibly based on some statistical analysis. An example of this is a system called ADSM which employs RAID arrays, multiple tape drives, a tape robot with barcode reader and intelligent software that tells the operators which tapes are to go off-site and which ones it wants back. It essentially is a huge cache that stores frequently changing data locally and stores old data off-site.

### Broken Mirror

If you've got about 100Gb of data on a mirrored pair of disks and only have a 10 minute backup window this may be for you. Basically you bring the system down, unhook one of the mirrors and replace it with another set of drives and bring the system up again. Mirroring starts from scratch during quiet time and should be finished before load picks up again. With the drive set you just un-hooked this can then be loaded into the standby server and backed up to tape over the course of many hours. Some high end servers can perform this operation without downtime as the hooking up can be done using inbuilt hardware or such things as dual-port fire-wire drive bays. All that is required in this case is an application shutdown, sync, dismount, remount, application start type operation.

### 5.2.3 Software

dd - can be used to copy raw disk blocks, even to tape (yuk). eg `dd if=/dev/hda1 of=/dev/hdb1`

tar - Tape ARchive - you all know how to unpack `tgz` files, and maybe even create them. Just remove the `'f'` option. It also can be an advantage not to use compression as some drives have this built in. Also, a portion of the tape being corrupt can ruin the rest of the data, whereas you can skip corrupt bits and pickup the next file if not compressed. eg `tar -c /home cd /tmp; tar -x`

cpio - cp I/O - Similar capabilities of tar but different methodology. EG `find /home — cpio -oB ;/dev/tape cd /tmp; cpio -idB ;/dev/tape`

rsync - remote sync - can sync a directory or whole filesystem by only transferring the changes between them. Be careful about trailing slashes. `rsync -a /home /backup/ rsync -a -e ssh /home backup@backup:/serverA/`

Arkeia - commercial package BRU - commercial package Amanda - Open source? Thousands more, some are client/server model and can backup multiple operating systems which is great.

See <http://www.linuxhelp.com.au/free.shtml> for our generic CPIO backup script.

### 5.2.4 Rotation & off-site strategies

It's no good having a backup if it's sitting next to the computer when there's a fire. You've got to have some off-site backups for really important stuff. On a small scale a friend of mine has a backup of all my music CD's I couldn't live without.

You could use this example strategy with any bulk medium but typically people refer to tapes or a set of tapes and for convenience I'll refer to a tape. If you can fit everything on one tape good for you, life is easy, backup the lot

daily. If you don't you'll have to do an incremental backup (ie what's changed) daily and do a whole backup with multiple tapes weekly. Take the weekly backup off-site home from work or over to a trustworthy friend's place. Once a month take a weekly backup to long term storage and keep it for 7 years or something if it's got all your tax info on it. It goes without saying the tapes should be labelled full/incremental and a date, hostname and what sequence in the set they are. Daily backup tapes may be rotated once a week with a new tape supplied once a week for a specific day of the week. Eg week1 will be all new tapes with one shipped off on Monday morning. week2 it'll be a new tape for Sunday morning, week3 it'll be Saturday morning's tape that's new. Alternately, some people believe the weekly or monthly should be on a fresh tape that's never been used.

With this strategy you get reasonable rotation of the tapes keeping costs down and for archival purposes, if you keep at least a months worth of data on the server you'll be able to go back to any point over the last few years and pull out a file. If you keep at least 3 months on hard disk you'll have 3 copies of this on 3 separate tapes because believe it or not they do fail and it will happen to you. To explain this more fully lets look at the following table and assume we have some wages data every week and the company's just started and there's 4 weeks per month.

| server | has     | weekly tape       | has | monthly tape     | has |
|--------|---------|-------------------|-----|------------------|-----|
| wk1    | wk1     | wk1               |     | -                |     |
| wk2    | wk1-2   | wk1-2             |     | -                |     |
| wk3    | wk1-3   | wk1-3             |     | -                |     |
| wk4    | wk1-4   | wk1-4, month1     |     | wk1-4, month1    |     |
| wk5    | wk1-5   | wk1-5, month1     |     | -                |     |
| wk6    | wk1-6   | wk1-6, month1     |     | -                |     |
| wk7    | wk1-7   | wk1-7, month1     |     | -                |     |
| wk8    | wk1-8   | wk1-8, month1-2   |     | wk1-8, month1-2  |     |
| wk9    | wk1-9   | wk1-9, month1-2   |     | -                |     |
| wk10   | wk1-10  | wk1-10, month1-2  |     | -                |     |
| wk11   | wk1-11  | wk1-11, month1-2  |     | -                |     |
| wk12   | wk1-12  | wk1-12, month1-3  |     | wk1-12, month1-3 |     |
| wk13   | wk2-13  | wk2-13, month1-3  |     | -                |     |
| wk14   | wk3-14  | wk3-14, month1-3  |     | -                |     |
| wk15   | wk4-15  | wk4-15, month1-3  |     | -                |     |
| wk16   | wk5-16  | wk5-16, month2-4  |     | wk5-16, month2-4 |     |
| wk17   | wk6-17  | wk6-17, month2-4  |     | -                |     |
| wk18   | wk7-18  | wk7-18, month2-4  |     | -                |     |
| wk19   | wk8-19  | wk8-19, month2-4  |     | -                |     |
| wk20   | wk9-20  | wk9-20, month3-5  |     | wk9-20, month3-5 |     |
| wk21   | wk10-21 | wk10-21, month3-5 |     | -                |     |
| ....   |         |                   |     |                  |     |

A complete backup and archive strategy should provide a means of going back to any point in time for critical data. Sometimes keeping the whole lot of data is not required. For example you could drop the weekly information and keep the monthly summary information and do a dedicated monthly backup for this data. The monthly data may be optimised and arranged for searching and an index provided but essentially contain all the information from the weekly data.

### 5.3 Lab

### 5.4 Questions

# Objective 111.6

## Maintain system time

### 6.1 Overview

#### 6.1.1 Weight: []

#### 6.1.2 Statement of Objective:

Candidate should be able to properly maintain the system time and synchronize the clock over NTP. Tasks include setting the system date and time, setting the BIOS clock to the correct time in UTC, configuring the correct timezone for the system and configuring the system to correct clock drift to match NTP clock.

#### 6.1.3 Key files, terms, and utilities include:

```
date hwclock ntpd ntpdate /usr/share/zoneinfo /etc/timezone  
/etc/localtime /etc/ntp.conf /etc/ntp.drift
```

#### 6.1.4 Resources of Interest:

**LPI Linux Certification in a Nutshell .**

by Jeffrey Dean  
O'Reilly

**LPIC 1 Certification Bible .**

*Angie Nash and Jason Nash*  
Hungry Minds

TBA

## 6.2 Notes

Prepared by Andrew Eager

### 6.2.1 Display or Set System Date & Time: **date**

The **date** command without any options will print the current date and time. The date will be relative to any timezone set for the machine.

```
[andy@Node4]$ date ↵
Tue May 21 09:57:51 EST 2002
```

#### Options to the **Date** command

**-I** Output an ISO-8601 compliant date (YYYY-MM-DD)

```
$ date -I ↵
2002-05-21
```

**-R** Output an RFC-822 compliant date (Local time + GMT Offset)

```
$ date -R ↵
Tue, 21 May 2002 10:14:09 +1000
```

**-r** **<file>** Display the last modification time of file

```
$ date -r ~/ivrr/va/src/va.c ↵
Mon May 20 12:55:48 EST 2002
```

**-d** **<STRING>** Display date described by string instead of now

```
$ date -d "last Monday 4 years ago" ↵
Mon May 18 00:00:00 EST 1998
```

**-u** Display UTC time & date instead of localtime

```
$ date ↵
Tue May 21 10:55:34 EST 2002
$ date -u ↵
Tue May 21 00:55:34 UTC 2002
```

**-s** **<date>** Set the system time (must be superuser)

```
# date -s "Tue May 21 10:03:06 EST 2002" ↵
Tue May 21 10:03:06 EST 2002
```

**+FORMAT** Display date in user defined format

```
$ date +"Today is %A, %d %B, %Y" ↵
Today is Tuesday, 21 May, 2002
```

## 6.2.2 The Hardware Clock: `hwclock`

### RTC <==>System clock

`hwclock` is used to do the following:

- Set the system clock from the Hardware clock
- Set the hardware clock from the system clock
- Show the time/date held by the RTC
- Adjust the RTC to account for clock drift

The *Real Time Clock (RTC)* is the hardware clock and is located on the motherboard of the system. This is what keeps track of the time when the system is not powered up. The *system clock* is maintained in the Linux kernel and is used while the system is running.

### Set System clock to Hardware clock

- To set the system time from the RTC, use the following option to `hwclock`:

```
hwclock -s (or hwclock --hctosys)
```

- To set the RTC from the system time, use this option:

```
hwclock -w (or hwclock --systohc)
```

- To display the contents of the RTC, use this option:

```
hwclock -r (or hwclock --show)
```

- To adjust the RTC for clock drift, use this option:

```
hwclock -a (or hwclock --adjust)
```

Note that the file `/etc/adjtime` is used to hold information about the extent to which (and direction) your RTC drifts

## 6.2.3 NTP - Network Time Protocol

### `ntp`

NTP is a time protocol used to synchronise a systems clock to master time source. For example, the CSIRO maintains a nationwide time source with atomic clock accuracy. As a user I can synchronise my system to that time source by sending a request to the CSIRO's `ntp` server.

Features and properties of NTP include:

- NTP takes into account the time taken to send/receive NTP packets
- Uses the UDP protocol

- Uses Port 123 plus one other unprivileged port (1024:65535)
- Can operate in both client & server modes
- There are 3 versions of the protocol (ntp1, ntp2 & ntp3)
- Available for Unix & Windows machines.

### The suite of tools

NTP normally comes in a package and contains the following binaries:

- `ntpd` - Network Time Protocol (NTP) daemon
- `ntpq` - standard NTP query program
- `ntpdc` - special NTP query program
- `ntpdate` - set the date and time via NTP
- `ntptrace` - trace a chain of NTP servers to the primary source
- `tickadj` - set time-related kernel variables
- `ntp_time` - read kernel time variables
- `ntp-genkeys` - generate public and private keys

### Quick install guide

For anyone new to NTP, here's a quick guide to installing & setting up NTP.

- Install NTP package (`rpm -Uvh ntp-4.1.0-4.rpm`)  
or  
`apt-get install ntp`
- Modify `/etc/ntp.conf` to reflect time servers
- Start the service: `service ntpd start`
- Confirm operation using `ntpq` (command `pe`)

That's all there is to it! The hardest part is deciding which public time servers to use.

### `ntpdate` - Set system time & date

- `Ntpdate` is a command line utility that will set the local machines time & date from the indicated remote time server(s).
- More than one server can be specified in order for `ntp` to get a better idea of the transit time and overall server accuracy.
- Running as a cron job is a simple way to maintain system time

Usage: `ntpdate [options] server ...`



```
# ntpdate ntp.nml.csiro.au
21 May 14:01:13 ntpdate[4002]: adjust time server 10.27.1.10
offset -0.000804 sec
```

This will set the local machines system time using server `ntp.nml.csiro.au`

### **ntpd - The NTP daemon**

- NTPD is a better way to maintain the system time on a permanent basis.
- NTPD acts as both a client & server (Linux only).
- In server mode, other machines on the local network can use the server to set their own system clocks
- For Windows machines, automachron is available.
- NTPD also keeps track of RTC drift.

The NTP daemon is normally started up by the system initialisation scripts. On a Red-Hat system you can start the service by:

```
service ntpd start
```

### **ntpd usage & configuration**

Usage: `ntpd [options] &`  
(normally done in the `/etc/init.d` scripts)  
NTPD is configured using these files:

- `/etc/ntp.conf` - Configuration file
- `/etc/ntp.drift` - RTC drift file
- `/etc/ntp.keys` - Key file (for authentication mode)

The only file of concern to the user is `ntp.conf`. The other files are all written to and read by the `ntp` applications.

### **Sample ntp.conf file**

```
# Disable authentication mode
disable auth
restrict default ignore      # ignore all requests by default
server ntp.cs.mu.OZ.AU      # 128.250.36.2
server apphys16.mst.csiro.au # 138.194.21.154
server ntp.nml.csiro.au     # 130.155.98.1
server 127.0.0.1            # localhost
# Lift restrictions on time servers
restrict 128.250.36.2 nomodify # time service only, no rt mods
restrict 138.194.21.154 nomodify
restrict 130.155.98.1 nomodify
# All local addresses are unrestricted
restrict 127.0.0.1
restrict 10.27.1.0 mask 255.255.255.0
# Set the default drift file
driftfile /etc/ntp/drift
```

### Public Time Servers

A (partial) list of public time servers is shown below. When using these servers, it is considered polite to advise the administrator of the service that you intend on using it.

- ntp.cs.mu.OZ.AU (128.250.36.2)
- apphys16.mst.csiro.au (138.194.21.154)
- ntp.nml.csiro.au (130.155.98.1)

### Testing NTP

Once you have the NTP daemon up & running, the easiest way of testing it is to use the ntpq utility.

```
# ntpq
ntpq> pe
      remote           refid      ...      delay    offset  jitter
=====
localhost.local 0.0.0.0    ...      0.000    0.000 4000.00
xmurgon.cs.mu.OZ .GPS.      ...      526.202  -206.43 208.270
+apphys16.mst.cs .ATOM.     ...      169.956   -5.576  87.828
*tictoc.tip.CSIR .ATOM.     ...      149.988  -24.328   6.761
ntpq> q
#
```

## 6.3 Lab

## 6.4 Questions

## **Topic 112**

# **Networking Fundamentals**



# Objective 112.1

## Fundamentals of TCP/IP

### 1.1 Overview

#### 1.1.1 Weight: []

#### 1.1.2 Statement of Objective:

Candidates should demonstrate a proper understanding of network fundamentals. This objective includes the understanding of IP-addresses, network masks and what they mean (i.e. determine a network and broadcast address for a host based on its subnet mask in "dotted quad" or abbreviated notation or determine the network address, broadcast address and netmask when given an IP-address and number of bits). It also covers the understanding of the network classes and classless subnets (CIDR) and the reserved addresses for private network use. It includes the understanding of the function and application of a default route. It also includes the understanding of basic internet protocols (IP, ICMP, TCP, UDP) and the more common TCP and UDP ports (20, 21, 23, 25, 53, 80, 110, 119, 139, 143, 161).

#### 1.1.3 Key files, terms, and utilities include:

```
/etc/services  
ftp  
telnet  
host  
ping  
dig  
traceroute  
whois
```

#### 1.1.4 Resources of Interest:

**Linux Networking HOWTO - Joshua Drake :**

<http://www.linuxdoc.org/HOWTO/Net-HOWTO/index.html>

**The Linux Networking Overview HOWTO by Daniel Lopez Ridruejo :**

<http://www.linuxdoc.org/HOWTO/Networking-Overview-HOWTO.html>

**Linux Network Administrators Guide :**

<http://www.linuxdoc.org/LDP/nag2/index.html>

**1.2 Notes**

**1.3 Lab**

**1.4 Questions**

# Objective 112.3

## TCP/IP configuration and troubleshooting

### 3.1 Overview

#### 3.1.1 Weight: []

#### 3.1.2 Statement of Objective:

Candidates should be able to view, change and verify configuration settings and operational status for various network interfaces. This objective includes manual and automatic configuration of interfaces and routing tables. This especially means to add, start, stop, restart, delete or reconfigure network interfaces. It also means to change, view or configure the routing table and to correct an improperly set default route manually. Candidates should be able to configure Linux as a DHCP client and a TCP/IP host and to debug problems associated with the network configuration.

#### 3.1.3 Key files, terms, and utilities include:

```
/etc/HOSTNAME or /etc/hostname
/etc/hosts
/etc/networks
/etc/host.conf
/etc/resolv.conf
/etc/nsswitch.conf
ifconfig
route
dhcpcd, dhcpclient, pump
host
hostname (domainname, dnsdomainname)
netstat
ping
traceroute
tcpdump
the network scripts run during system initialization.
```

#### 3.1.4 Resources of Interest:

Linux Networking HOWTO by Joshua Drake :

<http://www.linuxdoc.org/HOWTO/Net-HOWTO/index.html>

**Linux Ethernet-Howto by Paul Gortmaker :**

<http://www.linuxdoc.org/HOWTO/Ethernet-HOWTO.html>



**3.2 Notes**

**3.3 Lab**

**3.4 Questions**



# Objective 112.4

## Configure Linux as a PPP client

### 4.1 Overview

#### 4.1.1 Weight: []

#### 4.1.2 Statement of Objective:

Candidates should understand the basics of the PPP protocol and be able to configure and use PPP for outbound connections. This objective includes the definition of the chat sequence to connect (given a login example) and the setup commands to be run automatically when a PPP connection is made. It also includes initialisation and termination of a PPP connection, with a modem, ISDN or ADSL and setting PPP to automatically reconnect if disconnected.

#### 4.1.3 Key files, terms, and utilities include:

```
/etc/ppp/options.*  
/etc/ppp/peers/*  
/etc/wvdial.conf  
/etc/ppp/ip-up  
/etc/ppp/ip-down  
wvdial  
pppd
```

#### 4.1.4 Resources of Interest:

Linux PPP HOWTO Corwin Light-Williams and Joshua Drake :

<http://www.linuxdoc.org/HOWTO/PPP-HOWTO/index.html>

**4.2 Notes**

**4.3 Lab**

**4.4 Questions**

## **Topic 113**

# **Networking Services**



# Objective 113.1

## Configure and manage inetd, xinetd, and related services

### 1.1 Overview

#### 1.1.1 Weight: []

#### 1.1.2 Statement of Objective:

Candidates should be able to configure which services are available through inetd, use tcpwrappers to allow or deny services on a host-by-host basis, manually start, stop, and restart internet services, configure basic network services including telnet and ftp. Set a service to run as another user instead of the default in inetd.conf.

#### 1.1.3 Key files, terms, and utilities include:

```
/etc/inetd.conf  
/etc/hosts.allow  
/etc/hosts.deny  
/etc/services  
/etc/xinetd.conf  
/etc/xinetd.log
```

#### 1.1.4 Resources of Interest:

TBA

**1.2 Notes**

**1.3 Lab**

**1.4 Questions**



# Objective 113.2

## Operate and perform basic configuration of sendmail

### 2.1 Overview

#### 2.1.1 Weight: []

#### 2.1.2 Statement of Objective:

Candidate should be able to modify simple parameters in sendmail configuration files (including the "Smart Host" parameter, if necessary), create mail aliases, manage the mail queue, start and stop sendmail, configure mail forwarding and perform basic troubleshooting of sendmail. The objective includes checking for and closing open relay on the mailserv. It does not include advanced custom configuration of Sendmail.

#### 2.1.3 Key files, terms, and utilities include:

```
/etc/sendmail.cf  
/etc/aliases or /etc/mail/aliases  
/etc/mail/*  
~/forward  
mailq  
sendmail  
newaliases
```

#### 2.1.4 Resources of Interest:

TBA

**2.2 Notes**

**2.3 Lab**

**2.4 Questions**

# Objective 113.3

## Operate and perform basic configuration of Apache

### 3.1 Overview

#### 3.1.1 Weight: []

#### 3.1.2 Statement of Objective:

Candidates should be able to modify simple parameters in Apache configuration files, start, stop, and restart httpd, arrange for automatic restarting of httpd upon boot. Does not include advanced custom configuration of Apache.

#### 3.1.3 Key files, terms, and utilities include:

apachectl  
httpd  
httpd.conf

#### 3.1.4 Resources of Interest:

Apache home page :

<http://www.apache.org>

**3.2 Notes**

**3.3 Lab**

**3.4 Questions**

# Objective 113.4

## Properly manage the NFS, smb, and nmb daemons

### 4.1 Overview

#### 4.1.1 Weight: []

#### 4.1.2 Statement of Objective:

Candidate should know how to mount remote filesystems using NFS, configure NFS for exporting local filesystems, start, stop, and restart the NFS server. Install and configure Samba using the included GUI tools or direct edit of the `/etc/smb.conf` file (Note: this deliberately excludes advanced NT domain issues but includes simple sharing of home directories and printers, as well as correctly setting the `nmbd` as a WINS client).

#### 4.1.3 Key files, terms, and utilities include:

```
/etc/exports  
/etc/fstab  
/etc/smb.conf  
mount  
umount
```

#### 4.1.4 Resources of Interest:

TBA

**4.2 Notes**

**4.3 Lab**

**4.4 Questions**

# Objective 113.5

## Setup and Configure Basic DNS Services

### 5.1 Overview

#### 5.1.1 Weight: []

#### 5.1.2 Statement of Objective:

Candidate should be able to configure hostname lookups and troubleshoot problems with local caching-only name server. Requires an understanding of the domain registration and DNS translation process. Requires understanding key differences in configuration files for bind 4 and bind 8.

#### 5.1.3 Key files, terms, and utilities include:

```
/etc/hosts  
/etc/resolv.conf  
/etc/nsswitch.conf  
/etc/named.boot (v.4) or /etc/named.conf (v.8)  
named
```

#### 5.1.4 Resources of Interest:

TBA

## 5.2 Notes

### 5.2.1 Setup and Configure basic DNS services

#### Objective

Candidate should be able to configure hostname lookups and troubleshoot problems with local caching-only name server. Requires an understanding of the domain registration and DNS translation process. Requires understanding key differences in configuration files for bind 4 and bind 8.

### 5.2.2 Setup and Configure basic DNS services

#### Key files, terms, and utilities

```
/etc/hosts
/etc/resolv.conf
/etc/nsswitch.conf
/etc/named.boot (v.4) or /etc/named.conf (v.8)
named
```

### 5.2.3 DNS - DOMAIN NAME SERVICE

- The internet works with numbers not names.

```
www.abc.gov.au is really 203.2.218.61
```

- DNS namespace is made up of a tree of domain names.
- At the top is root (.)
- Below this is the Top Level Domain (TLD)
- Below the TLD is the Second Level Domain.
- The Second level domain is handled by whoever 'owns' that domain
- Third & lower level domains are handled by the domain owner.

### 5.2.4 DNS - DOMAIN NAME SERVICE

- Example:

```
node1.office.my-domain.com
^         ^         ^         ^
|         |         |         |
|         |         |         -- Top level domain
|         |         -- Second level domain
|         - Subdomain
-- Hostname
```

- Domain names are fully qualified (FQDN) when a name is specified all the way down to the hostname.



### 5.2.5 RESOLVING A NAME

- A name is resolved using the following steps:
  - `/etc/nsswitch.conf` is checked to see what resolution method to use (eg: read `/etc/hosts`, use `dns`, use `nis`...)
  - `nsswitch` says USE DNS:
    - \* Read `resolv.conf` to see what nameserver to use
    - \* Send request to nameserver and wait for response
  - `nsswitch` says USE HOSTS
    - \* Lookup `/etc/hosts` for a matching hostname

### 5.2.6 The `nsswitch.conf` file

- This is a file that determines what mechanisms are used by the hostname library calls to resolve names.
- The file contains lines with an identifier followed by a list of methods to use for name lookups.
- An example:
 

```
passwd: files nisplus nis
shadow: files nisplus nis
group: files nisplus nis
hosts: db files dns
```
- Note that the other entries like `passwd`, `shadow` and `group` are used for other applications like `login` and have nothing to do with DNS.

### 5.2.7 The `nsswitch.conf` file

- In the `hosts` line, we see that any hostname to be looked up will be done in the following order:
  1. Use local databases file (`.db` files in `/var/db`)
  2. Read `/etc/hosts`
  3. Search DNS
- The Search options can be one of:
 

```
nisplus (or nis+) - Consult NIS+ (Yellow Pages)
nis (or yp)       - Consult NIS
dns               - Use a DNS server
files             - Use local files like /etc/hosts
db               - Use local database files
compat           - Use NIS in compat mode
[NOTFOUND=return] - Stop searching and return host notfound
```

### 5.2.8 An example nsswitch file:

```

nisplus (or nis
passwd:    db files nisplus nis
shadow:    nisplus
group      db files nisplus nis

hosts:     db files nis dns

# Example - obey only what nisplus tells us...
#services: nisplus [NOTFOUND=return] files
#networks:  nisplus [NOTFOUND=return] files
#protocols: nisplus [NOTFOUND=return] files
#rpc:       nisplus [NOTFOUND=return] files
#ethers:    nisplus [NOTFOUND=return] files
#netmasks: nisplus [NOTFOUND=return] files

bootparams: nisplus [NOTFOUND=return] files

ethers:     files
netmasks:   files
networks:   files nis
protocols:  files nisplus
rpc:        files
services:   files nisplus

netgroup:   files nisplus

publickey:  nisplus

automount:  files nisplus
aliases:    files nisplus

```

### 5.2.9 The resolv.conf file

- This file configures how the system uses DNS. An example:

```

search aes
nameserver 10.27.1.10
nameserver 10.27.1.254

```

- The 'search' line says what to append to a non-fully qualified name:  
eg: ping node10 -> ping node10.aes
- The nameserver lines tell the hostname routines which dns server to send requests to. (If first lookup fails, use the second, third etc)

### 5.2.10 BIND - Berkley Internet Name Domain

- Bind is just one implementation of a DNS. Bind is to DNS what Apache is to http.
- Bind is configured with:

```

/etc/named.conf - For BIND V8
/etc/named.boot - For BIND V4

```

- Know that there is a difference between V4 & V8.
- Know how to configure V8 but not V4. (Different syntax)

### 5.2.11 BIND Configuration

- The configuration file contains subsections as follows:
  - Options → How named will operate
  - logging → What/how named will log information
  - Access Lists → Who can use named & what they can do
  - Remote Servers → Characteristics of remote servers
  - zones → Information about our defined domains

### 5.2.12 An Example Config file:

```
options {
    directory "/var/named/";
    forward only;
    forwarders {
        203.2.75.132;
        203.2.75.108;
    };
    query-source address * port 53;
    listen-on {
        10.27.1.10;
        127.0.0.1;
    };
    notify no;
};

#### The root zone ###
zone "." {
    type hint;
    file "named.ca";
};

#### A zone for localhost ###
zone "0.0.127.in-addr.arpa" {
    type master;
    file "0.0.127.in-addr.arpa.zone";
};

zone "localhost" {
    type master;
    file "localhost.zone";
};

### A local domain ###
zone "1.27.10.in-addr.arpa" {
    type master;
    file "1.27.10.in-addr.arpa.zone";
};

zone "aes" {
    type master;
    file "aes.zone";
};

key "key" {
    algorithm hmac-md5;
    secret "JoqlFqtncqurkhMOrrbQLYRcxSYXoNRovNTZBqWJFumleNkzOvEvTAbqpbMV";
};
```

### 5.2.13 Zone files:

- Each zone uses a file for:

- Hostname to IP address translations (Forward lookups)
- IP to Hostname translations (Reverse lookups)
- The names can be anything, but usually:
  - Forward file `-> /domain/<zone>`
  - Reverse file `-> /Net-IP/<in-addr.arpa>`
- Where the Net-IP is the network part of the IP address.

### 5.2.14 Zone Records:

**SOA record** Marks the start of a zone.

**NS record** Defines the name server for a zone or subdomain

**MX record** Define mail servers for domain

**CNAME record** Defines an alias for a hostname

**LOC record** Defines the physical location of the server

**SRV record** Defines what services are found where (eg ftp, http etc)

**A record** Defines hostname to IP address translations (forward file)

**PTR record** Defines IP address to hostname translations (reverse file)

### 5.2.15 Example Forward file `/var/named/aes.zone`

```
@      IN      SOA      nodel0.aes.  root.localhost (
                2 ; serial
                28800 ; refresh
                7200 ; retry
                604800 ; expire
                86400 ; ttl
                )

@      IN      NS       nodel0.aes.

node5  IN      MX       10      mail
node6  IN      MX       10      mail
node4  IN      MX       10      mail
node2  IN      MX       10      mail
nodel0 IN      MX       10      mail
gw     IN      MX       10      mail

node10 IN      A        10.27.1.10
node2  IN      A        10.27.1.2
node4  IN      A        10.27.1.4
node5  IN      A        10.27.1.5
node6  IN      A        10.27.1.6
cds    IN      A        10.27.1.99
gw     IN      A        10.27.1.254

ns     IN      CNAME    nodel0
mail   IN      CNAME    nodel0
node-4 IN      CNAME    node4
```

### 5.2.16 Example reverse file /var/named/1.27.10.in-addr.arpa.zone

```
@      IN      SOA      @      root.localhost (
                2 ; serial
                28800 ; refresh
                7200 ; retry
                604800 ; expire
                86400 ; ttk
                )

@      IN      NS      ns.aes.

2      IN      PTR      node2.aes.
4      IN      PTR      node4.aes.
5      IN      PTR      node5.aes.
6      IN      PTR      node6.aes.
10     IN      PTR      node10.aes.
99     IN      PTR      cds.aes.
254   IN      PTR      gw.aes.
```

### 5.2.17 Configuring a Caching only Nameserver

- A caching only nameserver is simple to setup.  
The first time a name is needed, a normal lookup occurs (Authoritative)  
The next time that name is needed, it is returned from cache (Non-authoritative)
- Under /etc/named.conf in the options section, just make sure you have the following directives set:

```
options {
    directory "/var/named/";
    forward only;
    forwarders {
        <First DNS to query>;
        <Second DNS to query>;
    };
    listen-on {
        <Your local IP address>;
        127.0.0.1;
    };
};
```

- Leave the root zone (.) and localhost entries as they are.

### 5.2.18 Testing DNS

- To test DNS, use one of the following tools:
  - nslookup (deprecated)
  - dig
  - host
- To use in their simplest form, just add the hostname you wish to query as the first option to the command:

```
nslookup node16.c222
dig node16.c222
host node16.c222
```

### 5.2.19 nslookup

- Usage: nslookup [option] host-to-find [-name-server]  
Example:

```
$ nslookup node2.aes -10.27.1.10 ↵
```

- Note: nslookup is deprecated and may be removed from future releases. Consider using the 'dig' or 'host' programs instead. Run nslookup with the -sil[ent] option to prevent this message from appearing.

```
Server:          10.27.1.10
Address:         10.27.1.10#53
```

```
Name:   node2.aes
Address: 10.27.1.2
```

### 5.2.20 dig

- Usage: dig [@name-server] host-to-find [query-type]
- Example:

```
$ dig @10.27.1.10 node2.aes ↵

; <<>> DiG 9.2.0 <<>> @10.27.1.10 node2.aes
;; global options: printcmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 43860
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;node2.aes.                IN      A

;; ANSWER SECTION:
node2.aes.                 86400  IN      A      10.27.1.2

;; AUTHORITY SECTION:
aes.                       86400  IN      NS     node10.aes.

;; ADDITIONAL SECTION:
node10.aes.                86400  IN      A      10.27.1.10

;; Query time: 5 msec
;; SERVER: 10.27.1.10#53(10.27.1.10)
;; WHEN: Mon Sep  2 13:48:38 2002
;; MSG SIZE rcvd: 80
```

### 5.2.21 host

- Usage: host [option] host-to-find [name-server]
- Example:

```
$ host node2.aes ↵
node2.aes has address 10.27.1.2
```

### 5.2.22 Exercise:

1. Install bind on your machine:

```
# rpm -Uvh bind-9*.rpm
```

2. Configure a Caching only nameserver on your machine. (Make all queries forward to 192.168.222.254)
3. Make changes to resolv.conf & nsswitch.conf as required (Default domain to use is c222)
4. Start the named.

```
# service named start
```

5. Test it out with the host node16.c222 using:

- nslookup
- dig
- host

6. Test again this time with the host box16

7. (For those who want a DNS challenge)

- (a) Setup a set of zones for the .c222 domain.
- (b) Insert the new zone into the main configuration file
- (c) Restart the named and test it.

## 5.3 Lab

## 5.4 Questions





# Objective 113.7

## Set up secure shell (OpenSSH)

### 7.1 Overview

#### 7.1.1 Weight: []

#### 7.1.2 Statement of Objective:

The candidate should be able to obtain and configure OpenSSH. This objective includes basic OpenSSH installation and troubleshooting, as well as configuring sshd to start at system boot.

#### 7.1.3 Key files, terms, and utilities include:

```
/etc/hosts.allow  
/etc/hosts.deny  
/etc/nologin  
/etc/ssh/sshd_config  
/etc/ssh_known_hosts  
/etc/sshr  
sshd  
ssh-keygen
```

#### 7.1.4 Resources of Interest:

TBA

**7.2 Notes****7.3 Lab****7.4 Questions**

# **Topic 114**

## **Security**



# Objective 114.1

## Perform security administration tasks

### 1.1 Overview

#### 1.1.1 Weight: []

#### 1.1.2 Statement of Objective:

Candidates should know how to review system configuration to ensure host security in accordance with local security policies. This objective includes how to configure TCP wrappers, find files with SUID/SGID bit set, verify packages, set or change user passwords and password aging information, update binaries as recommended by CERT, BUGTRAQ, and/or distribution's security alerts. Includes basic knowledge of ipchains and iptables.

#### 1.1.3 Key files, terms, and utilities include:

```
/proc/net/ip_fwchains  
/proc/net/ip_fwnames  
/proc/net/ip_masquerade  
find  
ipchains  
passwd  
socket  
iptables
```

#### 1.1.4 Resources of Interest:

TBA

**1.2 Notes****1.3 Lab****1.4 Questions**

# Objective 114.2

## Setup host security

### 2.1 Overview

#### 2.1.1 Weight: []

#### 2.1.2 Statement of Objective:

Candidate should know how to set up a basic level of host security. Tasks include syslog configuration, shadowed passwords, set up of a mail alias for root's mail and turning of all network services not in use.

#### 2.1.3 Key files, terms, and utilities include:

```
/etc/inetd.conf or /etc/inet.d/*  
/etc/nologin  
/etc/passwd  
/etc/shadow  
/etc/syslog.conf
```

#### 2.1.4 Resources of Interest:

TBA

**2.2 Notes**

**2.3 Lab**

**2.4 Questions**



# Objective 114.3

## Setup user level security

### 3.1 Overview

#### 3.1.1 Weight: []

#### 3.1.2 Statement of Objective:

Candidate should be able to configure user level security. Tasks include limits on user logins, processes, and memory usage.

#### 3.1.3 Key files, terms, and utilities include:

quota  
usermod

#### 3.1.4 Resources of Interest:

TBA

**3.2 Notes****3.3 Lab****3.4 Questions**

# List of topics

- 105 Kernel
  - Manage/Query kernel and kernel modules at runtime
  - Reconfigure, build, and install a custom kernel and kernel modules
- 106 Boot, Initialization, Shutdown and Runlevels
  - Boot the system
  - Change runlevels and shutdown or reboot system
- 107 Printing
  - Manage printers and print queues
  - Print files
  - Install and configure local and remote printers
- 108 Documentation
  - Use and manage local system documentation
  - Find Linux documentation on the Internet
  - Notify users on system-related issues
- 109 Shells, Scripting, Programming, Compiling
  - Customize and use the shell environment
  - Customize or write simple scripts
- 111 Administrative Tasks
  - Manage users and group accounts and related system files
  - Tune the user environment and system environment variables
  - Configure and use system log files to meet administrative and security needs
  - Automate system administration tasks by scheduling jobs to run in the future
  - Maintain an effective data backup strategy
  - Maintain system time
- 112 Networking Fundamentals

- Fundamentals of TCP/IP
- TCP/IP configuration and troubleshooting
- Configure Linux as a PPP client
- 
- 113 Networking Services
  - Configure and manage inetd, xinetd, and related services
  - Operate and perform basic configuration of sendmail
  - Operate and perform basic configuration of Apache
  - Properly manage the NFS, smb, and nmb daemons
  - Setup and configure basic DNS services
  - Configure ntp.conf and ntp.drift to be used by xntpd
- Security
  - Perform security administration tasks
  - Setup host security
  - Setup user level security
- 

## **.1 Topics moved to General Linux 1**

- -
- -
- -
- -
- 
-