Hands-on Projects

These projects should be completed in the order given. The hands-on projects presented in this chapter should take a total of three hours to complete. The requirements for this lab include:

* A computer with Fedora Linux installed according to Hands-on Project 2-1 and Ubuntu Server Linux installed according to Hands-On Project 6-1.

# Project 13-1

In this hands-on project, you install and configure the Very Secure FTP daemon on your Fedora Linux virtual machine.

1. Boot your **Fedora** Linux virtual machine. Login to your chosen desktop environment as **user1** using password **LNXrocks!** and open up a terminal window.
2. At the terminal, become **root** by typing **su -** and press enter and provide **LNXrocks!** as the password.
3. Install the **vsftpd** package to install the Very Secure FTP daemon.
4. Use the grep command for the string anonymous in the **/etc/vsftpd/vsftpd.conf**  file to review the anonymous settings for our FTP server.
5. Start the **vsftpd.service** and enable it to start at system boot.
6. Install the **ftp** package to install the command line ftp client.
7. At the command prompt, type **ftp localhost** and press Enter. When prompted for name, enter **anonymous** with no password (press enter when prompted for password).
8. At the command prompt, type **ls** and press Enter. You should see the pub directory. Next type **cd pub** to switch to the pub directory. Next type **ls** again to view the contents of this directory. Currently, no files exist here. Next type **exit** to logout of the ftp server.
9. At the command prompt, type **cp /usr/share/pixmaps/faces/\* /var/ftp/pub/** and press Enter to populate the pub directory. Follow steps 7 and 8 to view the files we just copied to the directory.
10. At the command prompt, type **get flower.jpg** and press Enter to download this image to our current directory. At the command prompt, type **quit** to logout of the FTP server. Verify the file was downloaded by typing **ll flower.jpg** and press Enter.
11. **Provide screenshot(s) of steps 3 through 10.**

# Project 13-2

In this hands-on project, you install and explore the NTP daemon on your Ubuntu Server Linux virtual machine as well as explore the Chrony NTP daemon on your Fedora Linux virtual machine.

1. Boot your **Ubuntu** Linux virtual machine. Login to your chosen desktop environment as **root** using password **LNXrocks!** and open up a terminal window.
2. Install the **ntp** package using the Ubuntu package manager.
3. Next, type **ps -ef | grep ntpd** and press Enter to verify that the NTP daemon is running.
4. Review the contents of **/etc/ntp.conf** file. What are the default ntp servers? **0.ubuntu.pool.ntp.org(continues from 0 – 3 at the front)**
5. At the command prompt, type **service ntp stop** and press Enter to stop the NTP daemon.
6. At the command prompt, type **ntpdate -u 0.ubuntu.pool.ntp.org** and press Enter to synchronize your clock with the first time server listed in /etc/ntp.conf.
7. At the command prompt, type **service ntp start** and press Enter to start the NTP daemon.
8. At the command prompt, type **ntpq -p** and press Enter to view information about the time servers that you are synchronizing with (peers).
9. Boot your **Fedora** Linux virtual machine. Login to your chosen desktop environment as **user1** using password **LNXrocks!** and open up a terminal window.
10. At the terminal, become **root** by typing **su -** and press enter and provide **LNXrocks!** as the password.
11. Review the **/etc/chrony.conf** file used for updating our time in Fedora.
12. At the command prompt, type **chronyc sources -v** and press Enter to view information about the time servers that you are synchronizing with (peers).
13. **Provide screenshot(s) of steps 2 through 12.**

# Project 13-3

In this hands-on project, you configure the Apache Web server on your Fedora Linux virtual machine and test daemon permissions to files on the system.

1. Boot your **Fedora** Linux virtual machine. Login to your chosen desktop environment as **user1** using password **LNXrocks!** and open up a terminal window.
2. At the terminal, become **root** by typing **su -** and press enter and provide **LNXrocks!** as the password.
3. Use grep and the case insensitivity option for **documentroot** in **/etc/httpd/conf/httpd.conf** What is the document root directory? **/var/www/html**
4. Use grep and the case insensitivity option for **directoryindex** in **/etc/httpd/conf/httpd.conf** What file(s) will automatically be handed out by the Apache daemon from the document root directory? **Index.html**
5. Use grep for **"User "** in **/etc/httpd/conf/httpd.conf** What user does the Apache daemon run as locally? **User**
6. Use grep for **"Group "** in **/etc/httpd/conf/httpd.conf** What user does the Apache daemon run as locally? **Group**
7. At the command prompt, type **apachectl configtest** and press Enter. Are there any syntax errors within your /etc/httpd/conf/httpd.conf file? **No**
8. Start and enable the **httpd.service** . Review the status of the **httpd.service** What PID of this service? **6304**
9. Open up a web browser and got to **http://localhost** to access the Fedora test page.
10. Next attempt to access **http://localhost/manual** to view the documentation for the Apache HTTP server. Note that you are unable to find this page. Install the **httpd-manual** package to install the manuals for Apache.
11. At the command prompt, type **apachectl reload** and press Enter. Now refresh your web browser to access the manual pages.
12. Create the following file **/var/www/html/index.html** with a text editor. Add the following lines:

<html>

<body>

<h1>My sample website</h1>

</body>

</html>

When finished, save your changes and quit the editor.

1. Change permissions on **/var/www/html/index.html** to **640**.
2. Open up a web browser (most likely firefox) and got to **http://localhost** to access our index.html file. Notice that you are unable to access the page since we removed the read permission from the others group.
3. Change permissions on **/var/www/html/index.html** to **644**.
4. Open up a web browser (most likely firefox) and got to **http://localhost** to access our index.html file.
5. At the command prompt, type **curl http://127.0.0.1/** and press Enter to display the HTML to standard out. At the command prompt, type **curl -o index.html http://127.0.0.1/** and press Enter to download the HTML to index.html.
6. **Provide screenshot(s) of steps 3 through 17.**
7. Install the **elinks** package to install a terminal based web browser.
8. Once installed type **elinks http://localhost/manual** to access the manuals in the terminal. Type q to exit elinks.
9. At the command prompt, type **curl http://127.0.0.1/** and press Enter to display the HTML to standard out.
10. At the command prompt, type **ab -n 10000 http://127.0.0.1/** and press Enter to use the Apache benchmarking tool against your web server.
11. Create a symbolic link from **/var/ftp/pub/** to **/var/www/html/** to link the contents of the FTP pub directory for use with Apache.
12. Modify **/var/www/html/index.html** file with a text editor. Append the file to look like the following:

<html>

<body>

<h1>My sample website</h1>

See additional files in the <a href="pub">pub</a> directory.

</body>

</html>

1. Open up a web browser and got to **http://localhost** to access your index.html. Click on the link to pub to view the image files in the directory.
2. Install the **mod\_ssl** package to add TLS/SSL support for Apache.
3. At the command prompt, type **apachectl restart** and press Enter to restart the Apache server.
4. Open up a web browser and got to **https://localhost** to access your site securely. When you see the This Connection is Untrusted page, click I Understand the Risks, then click Add Exception. From this page you can view the certificate. We will want to click on Confirm Security Exception to access our page encrypting the traffic.
5. **Provide screenshot(s) of steps 19 through 28.**

# Project 13-4

In this hands-on project, you configure apache user directories and apply a password to a directory.

1. Boot your **Fedora** Linux virtual machine. Login to your chosen desktop environment as **user1** using password **LNXrocks!** and open up a terminal window.
2. At the command prompt, type **mkdir -p /home/user1/public\_html/photos** and press Enter to create the directory we will utilize for this lab.
3. At the terminal, become **root** by typing **su -** and press enter and provide **LNXrocks!** as the password.
4. Modify **/etc/httpd/conf.d/userdir.conf** Find the line that reads UserDir disabled and changed to UserDir enabled. Also find the line that reads #UserDir public\_html and removed the # to read UserDir public\_html. Save and exit the editor.
5. Change the group owner to **:apache** to on the following directories:

**/home/user1 /home/user1/public\_html /home/user1/public\_html/photos** We have changed the group on our home directory, public\_html directory, and photos directory to allow the Apache webserver to access our public\_html directory.

1. Change the permissions of the following directories **/home/user1 /home/user1/public\_html /home/user1/public\_html/photos** to **750** We have changed the permissions to allow Apache to read the public\_html and photos directory.
2. At the command prompt, type **apachectl restart** and press Enter to restart the httpd service.
3. At the command prompt, type **setsebool -P httpd\_enable\_homedirs on** and press Enter to enable access to your public\_html for SELinux.
4. At the command prompt, type **setsebool -P httpd\_read\_user\_content on** and press Enter to allow access to the photos directory.
5. At the command prompt, type **semanage fcontext -a -t public\_content\_t '/home/user1/public\_html/photos'** and press Enter to change the label on the photos directory so we can access it.
6. At the command prompt, type **restorecon -FRv /home/user1/public\_html/** and press Enter to change the label on the photos directory to public\_content\_t.
7. Open a new terminal window to become **user1** and create **~/public\_html/index.html** Append the file to look like the following:

<html>

<body>

<h1>User1’s Personal Website</h1>

See my pictures in the <a href="photos">photos</a> directory.

</body>

</html>

1. Copy **/usr/share/pixmaps/faces/\*** to **~/public\_html/photos** to copy the pictures to the photos directory.
2. Create **~/public\_html/photos/.htaccess** and enter the following:

AuthUserFile /home/user1/public\_html/photos/.htpasswd

AuthName "Private Photos"

AuthType basic

require valid-user

1. At the command prompt, type **htpasswd -c ~/public\_html/photos/.htpasswd user1** and press Enter. Enter the password **LNXrocks!** And press Enter.
2. Open up Firefox and go to the following URL **http://localhost/~user1** and press Enter. You should now see user1’s personal webserver located in the /home/user1/public\_html/ directory.
3. Click on the photos link. You should be prompted for username and password to access the photos directory. Use the username **user1** and the password **LNXrocks!** and press enter.
4. **Provide screenshot(s) of steps 2 through 17.**

# Project 13-5

In this hands-on project, you configure and test Samba file sharing on your Ubuntu Server Linux virtual machine.

1. Boot your **Ubuntu** Linux virtual machine. Login to your chosen desktop environment as **root** using password **LNXrocks!** and open up a terminal window.
2. At the command prompt, type **ps –ef |grep mbd** and press Enter. Are the Samba daemons installed and started by default?
3. Review **/etc/samba/smb.conf** file with the vi text editor. Spend a few minutes examining the comments within this file to understand the available Samba configuration options. Under the Share Definitions section, notice that the only two shares configured by default are the [printers] share (which shares all printers to Windows hosts) and the hidden [print$] share (which shares print drivers to Windows hosts).
4. Add the following line underneath the [global] line in this file**:**

netbios name = *ubuntu-server*

***Where ubuntu-server is the hostname of your Ubuntu server***

1. Uncomment/modify the following section that shares out all home directories to users who authenticate successfully:

[homes]

comment = Home Directories

browseable = no

read only = no

1. Next, add the following share definition to the bottom of the file to share out the /etc directory to all users as read-only:

[etc]

comment = The etc directory

path = /etc

guest ok = yes

read only = yes

1. When finished, save your changes and quit the editor.
2. At the command prompt, type **testparm** and press Enter. Were any syntax errors reported within /etc/samba/smb.conf?  **Yes** Press Enter to view your Samba configuration.
3. At the command prompt, type **service samba restart** and press Enter to restart the Samba daemons.
4. At the command prompt, type **smbpasswd -a root** and press Enter. When prompted, supply the password LNXrocks!. Repeat the same password when prompted a second time.
5. At the command prompt, type **smbclient -L 127.0.0.1** and press Enter. Supply your Samba password of LNXrocks! when prompted. Do you see your shared home directory? **Yes** Do you see any printer shares? **No**
6. At the command prompt, type **smbclient //127.0.0.1/root** and press Enter. Supply your Samba password of LNXrocks! when prompted.
7. At the smb:\> prompt, type **ls** and press Enter. Are you in your home directory?
8. At the smb:\> prompt, type **lcd /etc** and press Enter to change your local directory to /etc.
9. At the smb:\> prompt, type **put issue** and press Enter to copy the /etc/issue file to our home directory via SMB.
10. At the smb:\> prompt, type **ls** and press Enter and ensure the issue file is in your home directory.
11. At the smb:\> prompt, type **get issue** and press Enter to download the issue file.
12. At the smb:\> prompt, type exit and press Enter.
13. **Provide screenshot(s) of steps 2 through 18.**

# Project 13-6

In this hands-on project, you explore the Postfix e-mail daemon on your Ubuntu Server Linux virtual machine.

1. Boot your **Ubuntu** Linux virtual machine. Login to your chosen desktop environment as **root** using password **LNXrocks!** and open up a terminal window.
2. At the command prompt, type **ps –ef | grep postfix** and press Enter. Is the Postfix daemon running?
3. Edit the /etc/aliases file with a text editor and add the following line:

webmaster: user1

When finished, save your changes and quit the editor.

1. At the command prompt, type **newaliases** and press Enter to update the aliases database using the information within the /etc/aliases file.
2. At the command prompt, type **mail webmaster** and press Enter to compose a new e-mail to the webmaster. When prompted for a subject, type **Test email** and press Enter. Next, type **This is a test email that will be delivered using the Postfix daemon** and press Enter. Next, type **.** (a period) and press Enter. Press Enter again to complete and send the e-mail.
3. At the command prompt, type **su - user1** to switch to a new shell as user1.
4. At the command prompt, type **mail** to check your mailbox for e-mail messages. Press Enter to view the body of the message. The last e-mail should have a subject line of Test e-mail. If you don’t see this message, type z to advance to the next screen of messages. Note the number of the e-mail message that has the subject line of Test e-mail and type this number at the & prompt to read your e-mail message. Type q when finished to exit the mail program.
5. At the command prompt, type **exit** and press Enter to return to your root shell.
6. At the command prompt, type **telnet localhost 25** and press Enter. Can you tell that you are interacting with the Postfix daemon? **No**
7. Type **EHLO localhost** and press Enter. Does your Postfix daemon support 8-bit MIME?**No** Type quit and press Enter to quit the telnet session.
8. **Provide screenshot(s) of steps 2 through 10.**

# Project 13-7

In this hands-on project, you export the /etc directory using NFS on your Ubuntu Server Linux virtual machine and access it from your Fedora Linux virtual machine

1. Boot your **Ubuntu** Linux virtual machine. Login to your chosen desktop environment as **root** using password **LNXrocks!** and open up a terminal window.
2. Install the **nfs-kernel-server** package to install NFS.
3. Edit the **/etc/exports** file with a text editor and add a line that reads:

/etc \*(rw)

When finished, save your changes and quit the editor.

1. At the command prompt, type **exportfs -a** and press Enter.
2. At the command prompt, type **service nfs-kernel-server restart** and press Enter to restart NFS.
3. Boot your **Fedora** Linux virtual machine. Login to your chosen desktop environment as **user1** using password **LNXrocks!** and open up a terminal window.
4. At the terminal, become **root** by typing **su -** and press enter and provide **LNXrocks!** as the password.
5. At the command prompt, type **dnf -y install nfs-utils** and press Enter.
6. At the command prompt, type **mount -t nfs IPaddress:/etc /mnt** (where IPaddress is the IP address of your Ubuntu Server Linux virtual machine hint: ip addr) and press Enter.
7. At the command prompt, type **df -h** and press Enter. What is mounted to the /mnt directory?
8. At the command prompt, type **ll /mnt** and press Enter. What directory are you observing? Type **cat /mnt/issue** at the command prompt and press Enter. Can you tell that the issue file is on your Ubuntu Server Linux virtual machine?
9. At the command prompt, type **umount /mnt** and press Enter to unmount the NFS filesystem.
10. **Provide screenshot(s) of steps 2 through 12.**