Labs

During the Red Hat exams, the tasks will be presented electronically. Therefore, this book presents most of the labs electronically as well. For more information, see the "Lab Questions" section toward the end of Chapter 10. Most of the labs for this chapter are straightforward and require a very few commands or changes to one or two configuration files.

Lab 1

In this lab, you'll set up a GPG key pair and test the result on the 19610-labs.txt file. This lab requires the use of two different systems. The server1.example.com and tester1.example.com systems are ideal for this purpose.

1. On the server1.example.com system, copy the 19610-labs.txt file to a regular user home directory.
2. On the tester1.example.com system, create a private-public key pair using RSA encryption.
3. Copy the public key to the server1.example.com system, and then import it.
4. Encrypt the 19610-labs.txt file.
5. Copy the encrypted file to the tester1.example.com system.
6. Decrypt the 19610-labs.txt file on the tester1.example.com system.

This lab verification is of course trivial (it either works or it does not), but there is no "answer," per se—it requires going back to the text. It might be worth putting in the full commands to generate the key, copy the key, import it, and then to encrypt, copy and decrypt the file.

Lab 2

Think about bastion systems. One implicit lesson of this chapter is to minimize what's installed on a system. In other words, if a service isn't installed, a black hat hacker can't take advantage of it. This lab uses the **rpm** commands discussed in Chapter 7. Although that is a RHCSA topic, these are fundamental skills for RHCEs. Strictly speaking, this lab does not directly address any RHCE objectives. So if you're pressed for time, skip this lab. But it does point to a fundamental way to keep a RHEL 7 system secure.

To that end, review the current output of the **systemctl list-unit-files --type=service** command. It's a handy reference point for currently installed services. Don't uninstall anything at this time. If you don't recognize a service, identify the associated daemon and RPM package. For example, to identify the package associated with the abrtd service, run the following command:

```
$ rpm -qf /usr/lib/systemd/system/abrtd.service
abrt-2.1.11-12.el7.x86_64
```

This identifies the abrt RPM package. To learn more about that package, run the following command:

```
$ rpm -qi abrt
```

For the scope of this lab, you don't need to repeat this process for all the services installed on the system. For example, look at the following services and get more information on the associated RPM packages, to establish whether they can be removed from the system:

```
abrtd.service
atd.service
avahi-daemon.service
bluetooth.service
```

Lab 3

You want to configure a firewall to accommodate a secure web server that supports inbound requests to both the regular and secure web server protocols. The system should also accept remote communications to the local SSH server. What do you do?

Lab 4

You want to set up an FTP service on your internal LAN, accessible only to one specific IP address. You want to block access from the LAN. Assume that your LAN's network address is 192.168.122.0/24, and the IP address of the computer that should get access is 192.168.122.150. For the purpose of this lab, feel free to substitute the IP address of a second Linux computer on the local network. What do you do?

Lab 5

You want to use TCP Wrappers to limit access to the local SSH server. First, what do you do to confirm that SSH can be protected by TCP Wrappers? Once you've done so, make the needed changes to ensure the SSH server on the server1.example.com system can be accessed only from the tester1.example.com system.

Lab 6

This lab assumes you've installed the vsFTP server, as discussed in Chapter 1. The focus of this lab starts with the /etc/vsftpd/ftpusers file. As noted in that file, it's a list of users who are not allowed to log in to the local vsFTP server. For security, it's an excellent idea to retain the /etc/vsftpd/ftpusers file.

The objective of this lab is to configure and limit access through the FTP server to one regular user account.