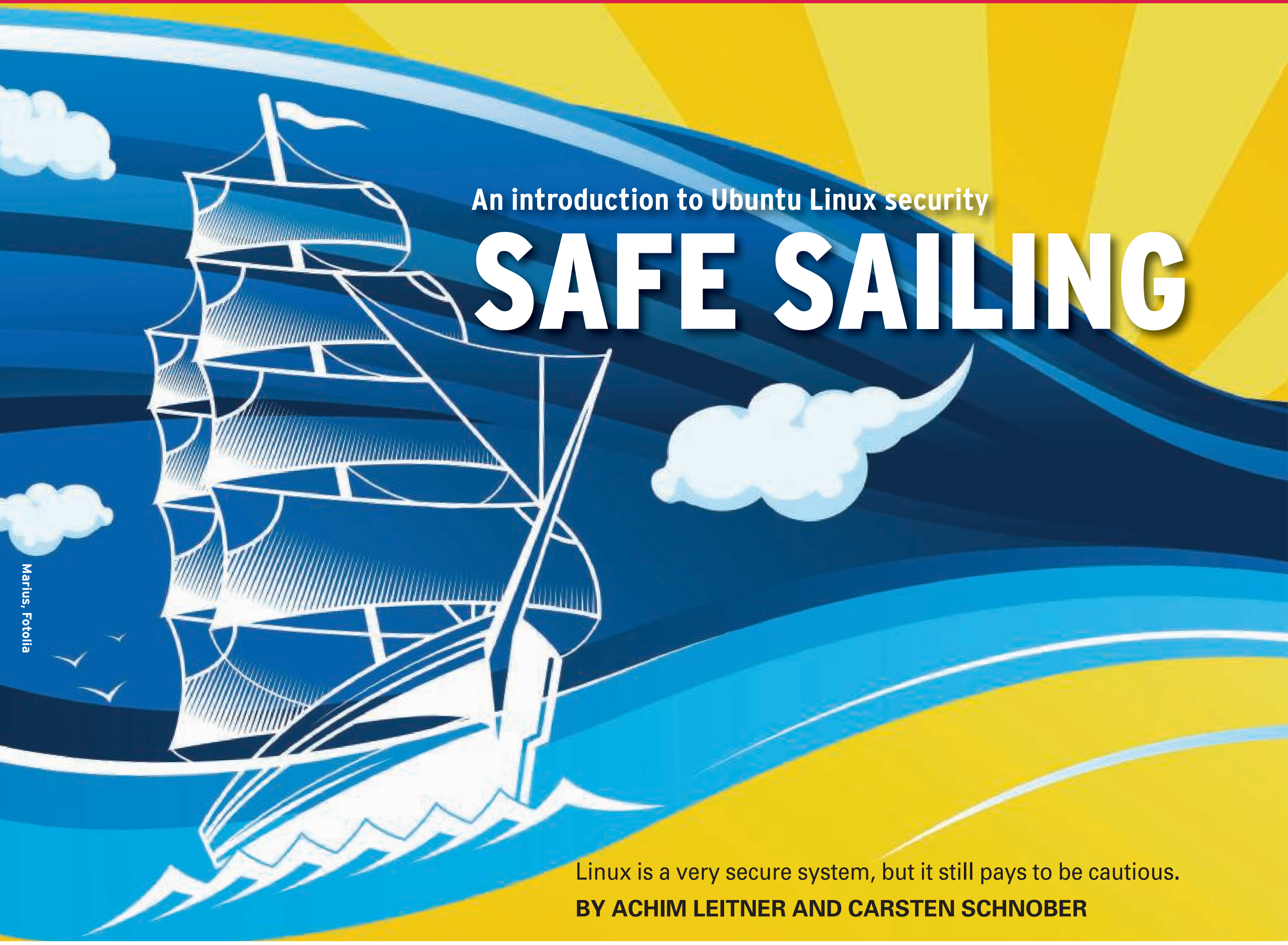


Marius, Fotolia

An introduction to Ubuntu Linux security

SAFE SAILING

Linux is a very secure system, but it still pays to be cautious.

BY ACHIM LEITNER AND CARSTEN SCHNOBER

Security is one reason many users switch to Linux from Windows. With the rising importance of computers in daily life, more and more private data are stored on desktop systems. The hunt for bank account access keys has become big business, and many users also store private messages, business plans, and other sensitive information on home PCs.

A fresh installation of Ubuntu is much less vulnerable than a comparable Windows system. Spyware, trojans, viruses, and other attack vectors have not settled in the world of Linux as they have with other systems, but it is important to understand some basic security if you want to keep your Linux system safe.

Linux Security Basics

Every discussion of computer administration revolves around finding the balance point between convenience and security.

If you're careful, you can have a system that is both convenient and secure. The best approach is to let each user do what is needed, and nothing more.

This basic idea is most obvious in the multiuser system, which includes Linux and recent Windows and Mac OS systems. Every user stores documents, email, and other personal data in a private area. In Linux, this area is known as the user's *home folder*. If a user accidentally chooses something like *Delete all*, other users' data are not affected. The operating system itself is in a place where an individual user can't delete or edit the files.

Keep in mind that misbehaving users are not the only danger for a computer system; malfunctioning software or an outside intruder who has breached security can cause considerable damage.

Root

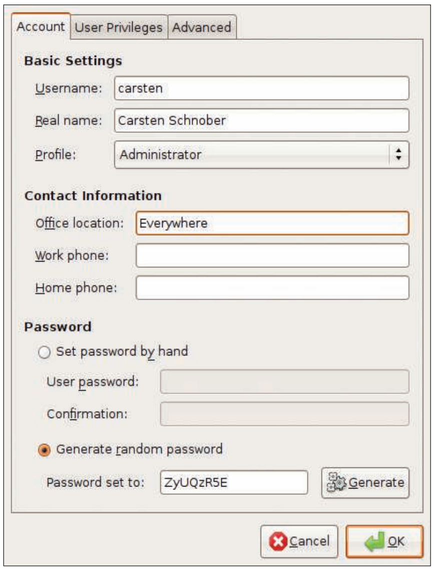
Users can take care of their daily work without the need for write access to critical system files. Sooner or later, though, you'll need to make changes to the system. For instance, new software or hardware might require changes to the system configuration.

Every operating system has a special user account with the elevated privileges needed to change the system; in Linux, this special account is called *root*. The root account, which is also sometimes called the superuser account, corresponds to what Windows users call the *Administrator* account.

When you log into a Linux system as the root user, you are allowed to do virtually everything: You may read every document, delete every file, remove every directory, and create new directories wherever you want.

Risky Business

Windows users who do their daily work as *Administrator* to avoid annoying privilege restrictions are taking a serious security risk. This warning also applies to the root user account in Linux. The root account should only be used for administrative work when it is required. Even if you are charged with the task of ad-



Basic Settings

Username: carsten

Real name: Carsten Schnober

Profile: Administrator

Contact Information

Office location: Everywhere

Work phone:

Home phone:

Password

○ Set password by hand

User password:

Confirmation:

● Generate random password

Password set to: ZyUQzR5E

Figure 1: Adding a new user account.

ministering the computer and are given the root password, you should still give yourself an ordinary user account for daily user tasks.

The traditional way to use the root account is to work as a normal user and, when necessary, switch to root during single administrative actions. Ubuntu further reduces the temptation to permanently work with superuser privileges by completely deactivating the root account by default. That means you will be able neither to log in as *root* nor to switch to the root account during your session.

Sudo

Of course, Ubuntu does not bar you from performing administrative tasks. Instead, the user account created during installation can gain superuser privileges with the command-line program sudo.

One of the advantages of *sudo* is that you do not have to remember another password. Before an administrator can execute an operation that requires privileged permissions – like changing network settings, setting the time, installing new software, or adding or removing software – Ubuntu asks for the user's normal password.

In one sense, this approach has the effect of reducing security for greater convenience because an intruder only needs to know your user password to gain full access to the system. Therefore, it is especially important that every person using the computer has a separate account. On the other hand, this approach forces the intruder to guess both a username and a password, whereas the classic root account always has the same name: *root*.

Managing Users

When you create a new account, you can choose whether the new user is a normal desktop user or an administrator who is qualified to perform system configuration tasks.

To create a new user in Ubuntu, open the Users and Groups dialog from the *System* | *Administration* menu, and you will see a list of existing users. First, click the *Unlock* button and enter your password; if you have administrative rights, you may now add and delete users and manage groups.

If you choose *Add User*, a New User dialog (Figure 1) pops up. In the *Account* tab, enter the *Username*, *Real name*, and optional *Contact Information* (e.g., telephone numbers).

In the *Password* section, you can either enter a temporary password yourself or make the system generate one. The new user is expected to change the password after the first login.

Granting Rights

In the *Profile* list, you can grant rights according to three levels of access. As described before, if you choose *Administrator*, the new user is allowed to gain superuser privileges and thereby can operate on the system without limits. However, if he or she is not required or allowed to do so, choose *Desktop user*. This lets the new user work in a home folder and gives full access to external media but forbids tasks that result in changes to the system settings. *Unprivileged* creates a user with minimal rights. A possible use for this kind of account would be to support users who log in via a network and therefore do not need access to local devices.

The *User Privileges* tab reveals more details about the chosen profile. Here, you'll find a list of specific actions, such as *Administer the system*, *Access local storage devices automatically*, or *Use audio devices*. All these actions can be activated separately.

If you choose the *Administrator* profile, you'll see a check mark next to every entry. By clicking the checkboxes, you can adjust each new user's privileges individually. Be aware that users who are allowed to administer the system can change their own permissions.

Linux employs the concept of *groups* to manage access to files and other resources. In Unix and Linux, a group is a collection of users with certain common permissions. Linux grants different write and read permissions to the owner of the file and to the members of the group.

In the User Settings main window, the *Manage Groups* button shows a list of all groups in your system. Double-click an entry to see the group's current members. If you want, you can add users to the group manually, which has the same effect as activating the corresponding checkbox in the user properties.

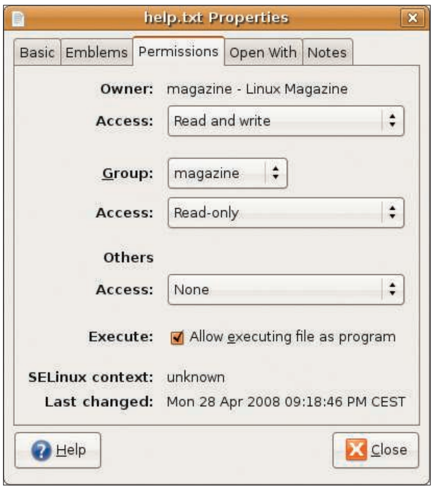
Read, Write, and Execute

User permissions are implemented as a system of access levels. This even applies to the access control on hardware devices because Linux manages these devices through special files residing in the */dev* directory.

File permissions are divided into three sections that specify what the owner, the members of the owner group, and all other users are allowed to do with the file. The basic access categories are read, write, and execute.

When you right-click on any file in the file browser and choose *Properties*, the *Permissions* tab provides a view of access permissions (Figure 2). The upper section shows the file owner and the owner's access level: *Read and write*, *Read-only*, or *None*.

Execution permissions are handled separately with the checkbox labeled *Allow executing file as program*.



Owner: magazine - Linux Magazine

Access: Read and write

Group: magazine

Access: Read-only

Others

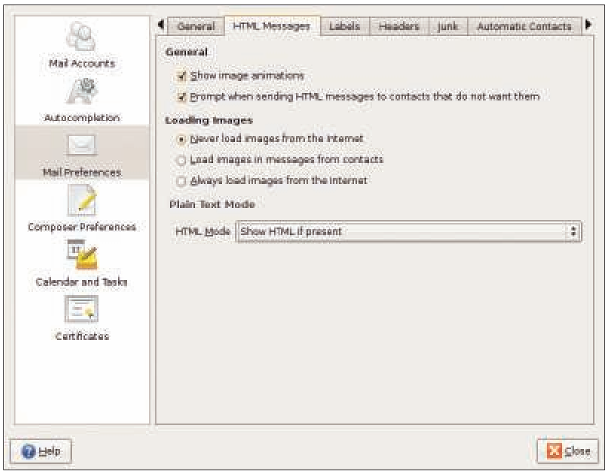
Access: None

Execute: ☑ Allow executing file as program

SELinux context: unknown

Last changed: Mon 28 Apr 2008 09:18:46 PM CEST

Figure 2: Use the Permissions tab to set file permissions.



General

☑ Show image animations

☑ Prompt when sending HTML messages to contacts that do not want them

Loading Images

● Never load images from the Internet

○ Load images in messages from contacts

○ Always load images from the Internet

Plain Text Mode

HTML Mode: Show HTML if present

Figure 3: Make sure you know how your mail client is processing HTML email.

Whether you are able to change these settings depends on your access permissions.

Directories

Directories are handled in the same way as files except for execution. (Folders can have execution access as well, but in that case, it means that users are allowed to browse the directory content.) The Properties dialog of a directory looks like that of a file, but you'll also find a button labeled *Apply permissions to enclosed files*. If you click this button, the permission settings affect files and subdirectories in this folder.

Filesystem settings can override file permissions. Ubuntu makes use of this feature for external storage devices. By default, external storage devices are mounted with the *noexec* flag, which means that enclosed programs cannot be executed. This protects the system from uncontrollable software brought in by CDs or other external sources.

In a terminal window, the command *ls -l* gives information about file permissions. This command shows every entry in the current directory in a single line:

```
-rw-r--r-- 1 carsten users ↩
43142 2008-04-28 20:17 data.txt
```

For normal files, the first position is a dash; otherwise, *d* points to a directory. Device files and other special files show *c*, for example. The nine symbols that follow relate to permissions. These symbols are interpreted in three sections of three letters each. The first section refers to the file owner, the next refers to the group, and the last three positions are relevant for the remaining users. Each one of these nine positions can show either a dash or one of the letters *r* (readable), *w* (writable), or *x* (executable).

Viruses in Linux

Theoretically, viruses could threaten Linux, but in practical terms, this hardly ever happens. The reasons for this natural protection are mainly rooted in the technology. To propagate, the virus needs to affect other programs it locates on the file system. Older Windows versions give viruses a free hand, whereas Linux slams the door shut. As long as the program has a normal user account, rather than one with root privileges, it is not permitted to overwrite other program files. This leaves the virus without a propagation vector.

On top of this, Linux users rarely send each other pre-compiled binaries. Because any software you need is available for free, users are more likely to obtain the program from a website or distribution DVD.

Several vendors provide virus scanners for Linux [1], but the primary purpose of these tools is to protect email messages or files on file servers that might one day find their way to a Windows system.

Email Worms

In contrast to viruses, worms do not need a host program but run autonomously. A typical email worm sends itself to all the contacts in an address book. But it only causes damage to the receiving machine if the recipient carelessly opens the attachment or if the worm can exploit a vulnerability in the mailer.

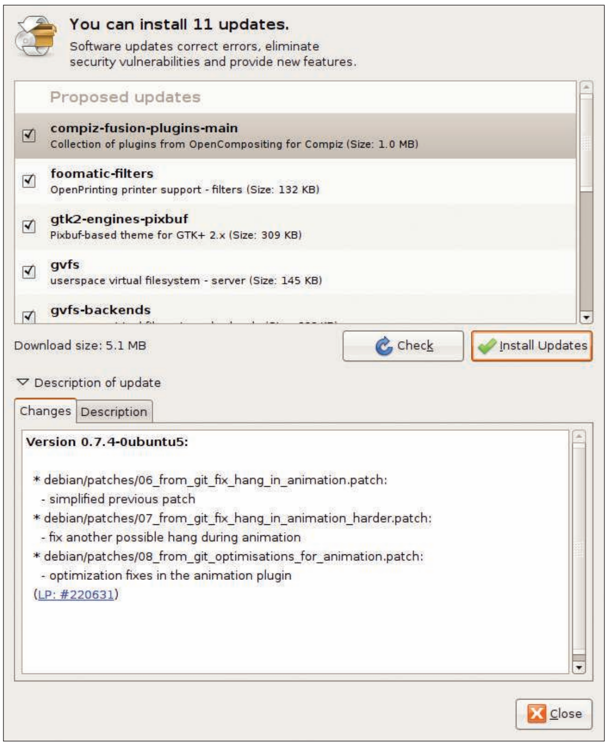
HTML-based email is also a risk. Many email clients ignore HTML attachments or deploy secure program modules to display HTML code. Check to see how your web browser treats HTML-based email. In Linux – just as in any other operating system – don't open attachments with untrusted messages and, above all, never launch any programs sent to you unexpectedly.

Evolution

Ubuntu's default mailer Evolution does display HTML mail by default. If you prefer to avoid this, go the preferences dialog via the menu entry *Edit* | *Preferences*. In the *Mail Preferences* section, you will find the *HTML Messages* tab (Figure 3).

In the Plain Text Mode section, you can disable HTML by choosing *Only ever show PLAIN*. Some email messages consist of two parts: One part contains an HTML version, and the other part contains pure text. If you prefer the colorful HTML version, select *Show HTML if present*.

HTML email also provides the possibility of embedding direct links to Internet images. Because it is generally better not to load files from the Internet unless you know what they are, the default setting is *Never load images from the Inter-*



You can install 11 updates.

Software updates correct errors, eliminate security vulnerabilities and provide new features.

Proposed updates

☑ compiz-fusion-plugins-main
Collection of plugins from OpenCompositing for Compiz (Size: 1.0 MB)

☑ foomatic-filters
OpenPrinting printer support - filters (Size: 132 KB)

☑ gtk2-engines-pixbuf
Pixbuf-based theme for GTK+ 2.x (Size: 309 KB)

☑ gvfs
userspace virtual filesystem - server (Size: 145 KB)

☑ gvfs-backends

Download size: 5.1 MB

▽ Description of update

```
Version 0.7.4-0ubuntu5:

 * debian/patches/06_from_git_fix_hang_in_animation.patch:
   - simplified previous patch
 * debian/patches/07_from_git_fix_hang_in_animation_harder.patch:
   - fix another possible hang during animation
 * debian/patches/08_from_git_optimisations_for_animation.patch:
   - optimization fixes in the animation plugin
 (LP: #220631)
```

Figure 4: The Update Manager helps you keep your system up to date.

net. If you indeed want to see images from a known secure source, change this option in the HTML preferences to *Load images in messages from contacts* or even to *Always load images from the Internet*.

Your own mail is written in plain text by default. If you prefer HTML, go to the *Composer Preferences* section in the Preferences window.

To enable HTML formatting, activate the checkbox next to *Format messages in HTML*. Alternatively, you can enable it for a specific message in the composing window.

When you are writing a message, the Format menu provides the option *HTML*, which you can can check or uncheck depending on current needs.

Be aware that the receiver's mail software also has to be able to display HTML email, so if in doubt, you should stay with plain text.

Don't Forget to Update

When a vulnerability is discovered, the correct response is to fix the program, preferably by updating the vulnerable program package. The Ubuntu Update Manager automatically locates patches for installed packages.

Vulnerabilities in server programs are particularly critical. The server just waits for a client to connect, and, of course, an attacker who gets inside can hitch up a server to the Internet and wait for victims. In fact, an attacker doesn't even need to know a victim's address: The intruder can simply search large sections of the network looking for low-hanging fruit. What makes this even worse is that servers often run with root privileges – following an attack, the attacker assumes the privileges of the victim, thus controlling the whole computer.

By default, Ubuntu checks for available updates every day. Ubuntu will indicate the presence of a new software package for your system with a message and put the update manager icon into the upper panel.

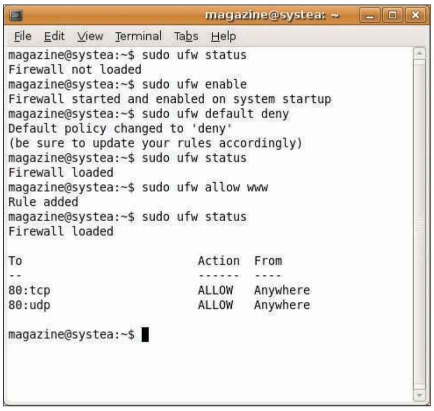
If you click the icon with the left mouse button, the Update Manager window opens (Figure 4). Different update types might be available; when security updates are listed, you should install them immediately.

Recommended updates usually fix bugs in applications that could cause application crashes but are not considered to be related to security.

After you click on the *Install Updates* button, the system asks for your password then downloads and installs the updates.

Firewalls

Many networks deploy firewalls to protect against outside attack (Figure 5). Firewalls only allow data packets to pass if they comply with specific criteria. A firewall can only give you limited protection: If you permit a connection, you are leaving it up to the client and server programs to defend themselves. In other words, the best protection is to install and launch only those services that you really need.



```
magazine@systea:~$ sudo ufw status
Firewall not loaded
magazine@systea:~$ sudo ufw enable
Firewall started and enabled on system startup
magazine@systea:~$ sudo ufw default deny
Default policy changed to 'deny'
(be sure to update your rules accordingly)
magazine@systea:~$ sudo ufw status
Firewall loaded
magazine@systea:~$ sudo ufw allow www
Rule added
magazine@systea:~$ sudo ufw status
Firewall loaded

To                         Action  From
--                         ------  ----
80:tcp                     ALLOW   Anywhere
80:udp                     ALLOW   Anywhere

magazine@systea:~$
```

Figure 5: The ufw firewall configuration utility is useful if you are comfortable at the command line.

In addition to server programs, which must be available to outside access and, therefore, are vulnerable to attack, some client programs are also endangered. A client application theoretically opens itself up to attacks by servers. For example, a web browser surfing the web exposes itself to the dangers of spoofing or phishing attacks.

As a protection against outside attack, Ubuntu installs no server software in the default configuration. However, many users might want to explore the possibility of running network services in Ubuntu.

If you are looking for some additional control, Ubuntu 8.04 comes with a new firewall configuration tool called Uncomplicated Firewall (ufw). Configure ufw at the command line by entering the command *sudo ufw* followed by some arguments. The prepending of *sudo* is necessary because controlling network traffic requires administrator privileges. Complete documentation for ufw is at the Ubuntu wiki [2].

Solid Security

Linux does a good job of protecting itself against attackers, but you need to be careful and always install the updates. Although a little attention is warranted, the natural defenses of Linux are extremely successful at keeping the intruders away. ■

Secure Passwords

Even Linux is powerless if users inadvertently or negligently endanger its security. Passwords play a key role in the security of a system. If you want a safe system, don't use passwords that are easy to guess. Using a friend's birthday or your pet's name is definitely not a good idea. An attacker who is looking to break passwords will deploy programs to automate the guesswork. These programs use comprehensive dictionaries in multiple languages and possess rules on how to compile words and replace letters with figures or nonstandard characters.

To make the guesswork as difficult as possible, a password should not be too short. Unix and Linux systems often demand eight-character passwords. These length restrictions are no longer valid for systems that use MD5-based password encryption techniques.

The number of different characters is particularly important: A mixture of uppercase and lowercase letters, numbers, and nonstandard characters, such as # < > _ = ( ), makes a password more effective.

Despite all the versatility and randomness, a user needs to be able to remember the password. Mnemonic aids can help, but they have to be just as secret as the password itself. Also of importance is that each password be used for one task only. For example, you should never use your login password for a web forum as your password for the root account on another computer. And you should change your password at regular intervals.

INFO

[1] ClamAV open source antivirus tool: *http://www.clamav.org/*

[2] Uncomplicated Firewall documentation: *https://wiki.ubuntu.com/UbuntuFirewall*