Published in the September 2008 issue of Linux for You

How to build a fortress

S. Parthasarathy
drpartha@gmail.com

This article is about two powerful utilities, to ensure security of your data/files. Let us assume that you have a whole lot of files which you want to protect aganst any mischievous tampering or alteration. In an article published in LFY (August 2008), we saw the use of md5 checksums (or hash digests). We will combine md5 with a concept called digital signature, to protect your data.

The principle of our scheme is very simple :

- With the GPG program (included in all Linux distros), you can digitally "sign" files, and ensure tamper detection easily. But, the catch is, with GPG, you can "sign" only one file at a time.

- With MD5 you can create checksums of several files at a time (using the md5sum program bundled with your Linux distro). But, the catch is that if an intruder can tamper the file, he can also tamper the md5 checksum, and destroy all traces of his mischief.

Any modification to any of the files is detected by verifying the md5 checksum. Any modification to the md5 checksum file gets detected by verifying, using GPG, the digital signature of the checksum file.

These are fairly sophisticated tools. We will make some simplfying assumptions, and demonstrate the concepts involved.

Create a test directory called, say md5lfy. Put all the files you want to protect, into this test directory. Let us say that this test directory contains:

-rw-r--r-- 1 root root 11 Jul 23 14:39 a

-rw-r--r-- 1 root root 21 Jul 23 14:39 b

-rw-r--r-- 1 root root 32 Jul 23 14:40 c

Step 1: *Create the md5sums of files in md5lfy*

cd to md5lfy and run md5sum, like this:

md5sum * >> md5sums

The md5sums file will now contain md5sum of all the files. This is what it contains now:

e96a190b5b0300e551c9863b80d76bdd a

addf31477cd03c11529ec909112a3881 b

01cabf3243cc874cafd552d7e4d890c2 c

The first column is the md5sum of the file named in the second column.

Step 2: *Check md5sums of all files in md5lfy*

Now, let us see if the sums in the md5sums file conform to the files. Run md5sum command to check, like this:

md5sum -c md5sums

This is what you should see:

a: OK

b: OK

c: OK

Now, change one of the files (say a). And run md5sum -c md5sums again. This is what you will see:

a: FAILED

b: OK

c: OK

So, alterations made to any file can be spotted easily, by using md5sum in the checking mode (-c option): md5sum -c md5sums.

Step 3: *Protect this md5sums file*

The cunning criminal who tampered your files can also tamper the md5sums file. So, now we use GPG (an encryption tool) to digitally sign the md5sums file. The GPG signature is done

using a secret key known only to you. You can learn about GPG here: http://www.gnupg.org/ . The digital signature consists of an encrypted form of the hash digest of the file which is signed. Any change to the file, will render its digital signature ineffective. You can verify the digital signature, using the public key of the person who signed the file (using his own secret key). Thus any change to the md5sums will get noticed on verification of its digital signature.

You have now created a fortress for the files in your md5lfy directory. You can of course use your imagination and ingenuity, to extend this approach, to suit our specific needs. For instance, md5sum does not go into directories. Use the following one line script (called md5sigma) to give the md5sums of all files recursively starting from a root directory of your choice. You pass the root directory as the only parameter in the command line like this:

```
md5sigma  myrootdirectory
```

md5sigma is a one-line script containing :

```
find $1 -type f -print0 | xargs -0 md5sum -b > md5sigma.md5
```

To verify that this md5 checksums are okay, use the command:

```
md5sum -c md5sigma.md5
```

Your comments and queries, are most welcome. Send them to drpartha@gmail.com.

partha

** * * **