

NS Linux SIG - April 2009

Random Bytes

md5 checksums.

Used to check the integrity of a file using an MD5 (128 bit) checksum. The sums are computed as described in RFC 1321.

eg/.

```
$ md5sum test.txt
9b2e0f300cb90d1f6e8ed056e4261010 test.txt
```

Use an editor to add only 1 space at beginning of file, and run md5sum again. The generated checksum will be quite different.

We can make a checksum file quite easily.

```
$ md5sum test2.txt > test2.txt.md5sum
```

This gives a one line file with contents something like ..

```
672fe2f40d328cd3686878a97b299616 test2.txt
```

then use the command

```
$ md5sum -c test.txt.md5sum
test.txt: OK
```

Usefull for checking large files, eg/. isos of linux distros, where checksum files are usually available from the download sections of the distro's website.

Tips and tricks with .rpm packages. (Fedora, CentOS etc)

GPG keys

Determine which GPG public keys are installed ...

```
$ rpm -qa gpg-pubkey*
gpg-pubkey-e8562897-459f07a4
```

To get details about this key, use

```
$ rpm -qi gpg-pubkey-e8562897-459f07a4
Name           : gpg-pubkey                Relocations: (not relocatable)
Version        : e8562897                Vendor: (none)
Release        : 459f07a4                Build Date: Thu 16 Apr 2009 08:52:53 PM EST
Install Date:  Thu 16 Apr 2009 08:52:53 PM EST      Build Host: localhost
Group          : Public Keys              Source RPM: (none)
Size           : 0                        License: pubkey
Signature      : (none)
Summary        : gpg(CentOS-5 Key (CentOS 5 Official Signing Key) <centos-5-key@centos.org>)
Description    :
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: rpm-4.4.2.3 (NSS-3)
```

```
mQGiBEWfB6MRBACrnYW6yKMT+MwJlChoyTxGf3mAxmnAiDEy6HcYN8rivssVTJk
CFtQBIBOPvL/OW2YtKrCO2xHn46eNfnMri8FGT8g+9JF3MUVi7kiV1He4ijynHXB
+F2ZzqlvHf3laUj1ys+p8TK64FDfXDQDrGQfIsD/+pkSGx53/877lrvdwiwCguQcr
loip5TH0Fj0OLUY4asYVZ8EAAlqFHEqsY+9ziP+2R3/FyxSllKkjwcMLrBug+cYO
LYDD6eQXE9Mq8XKGFdj9ZB/0+JzK/XQeStheeFG75q3noq5oCPVFO4czuKErIRAB
qKbDBhaTj3JhOgM12XsUYn+rI6NeMV2ZogoQCC2tWmDETFrYp2moo53NuFWHbAy
XjETA/sHEeQT9huHzdi/lebNBj0L8nBGfLN1nSRP1GtvagBvkR4RZ6DTQyI0UzOj
RA3ywWlrL9V9mrpb1Fmn60I2jTMMcc7J6LacmPK906N+FcN/Docj1M4s/4CNanQ
NhzcFhAftQL56SNyLTck1XzhssGZ/jwGnNbU/aaj4wOj0Uef5LRGQ2VudE9TLTug
S2V5lChDZW50T1MgNSBPZmZpY2lhbCBTaWduaW5nlEtleSkGPGnlbnRvcy01LWtl
eUBjZW50b3Mub3JnPohkBBMRAGAkBQJFfwekAhsDBQkSzAMABGsjCacDAgMVAgMD
FgIBAh4BAheAAAoJEKikR9zoViiXKIEAmwSoZDvZo+WChcg3s/SpNoWCKhMAAJwI
E2aXpZVrpsQnInUQWwkdrTiL5YhMBBMRAGAMBQJFfwisBYMSzAIRAAoJEDjCFhY5
bKCK0hAAn134blx3wSbq58E6P6U5RT7Z2Zx4AJ9VxnVkoGHkVlgSdsxHUgRjo27N
F7kBDQRfnwezEAQA/Hnj5yiozwtg6jt+kii8iua+WnjqBKomPHOQ8moxbWdv5Ks
4e1DPHzRqxhshjmub4Suj93sgMSAF2ayC9t51mSJV33KfzPF2glahcMqfABe/2hj
aMzcQZHRGjCEX6ek8l8SFkou7vlCzyajRSIK8gxWKBuQknP/9LKsoczV+xsAAwUD
/idXPkk4vRRHsCwc6l23fdI0ur52bzEqHiAlswNfO521YgLk2W1xyCLc2aYjc8Ni
nrMX1tCnEx0/gK7lCyjoWH1Vc7//79sWftX2EaTO+Q07xjFX4E66WxjICo9lOjos
Vk5qc7R+xzLDoLGFtbzaTRQFzf6yr7QTu+BebWLoPwNtIE8EGBECAAA8FAkWfB7MC
GwwFCRLMAwAACgkQqKRH3OhWKJfvvACfbsF1WK193zM7vSc4uq51XsclWgAol0/
9GxdNhGQEAweSIQfhPa3yYXH
=o/Mx-----END PGP PUBLIC KEY BLOCK-----
```

```
$
```

Using rpm to check the integrity of packages

For example, you want to check the integrity of the cups package

```
$ rpm -qa | grep cups
cups-1.3.7-8.el5
cups-libs-1.3.7-8.el5
hal-cups-utils-0.6.2-5.2.el5
libgnomecups-0.2.2-8
```

Having confirmed the name of the cups package, use the verify option

```
$ rpm --verify cups-1.3.7-8.el5
..?..... c /etc/cups/classes.conf
..?..... c /etc/cups/printers.conf
..?..... /usr/lib/cups/backend/ipp
..?..... /usr/lib/cups/backend/lpd
..?..... /usr/lib/cups/backend/serial
$
```

The format of the verify output is a string of 8 characters, a possible attribute marker from the package header, followed by the file name.

This output is interpreted as follows ..

1) The 8 character string:

Each of the 8 characters denotes the result of a comparison of attribute(s) of the file to the value of those attribute(s) recorded in the database. A single "." (period) means the test passed, while a single "?" (question mark) indicates the test could not be performed (e.g. file permissions prevent reading). Otherwise, the (mnemonically emBoldered) character denotes failure of the corresponding --verify test:

```
S file Size differs
M Mode differs (includes permissions and file type)
5 MD5 sum differs
D Device major/minor number mismatch
L readLink(2) path mismatch
U User ownership differs
G Group ownership differs
T mTime differs
```

2) The attribute marker:

```
c %config configuration file.
d %doc documentation file.
g %ghost file (i.e. the file contents are not included in the package pay      load).
l %license license file.
r %readme readme file.
```

3) The full file name.

You will usually need to run this command with root privileges for the results to be accurate.

```
$ su -
Password:
# rpm --verify cups-1.3.7-8.el5
#
```

Freeing up disk space.

You can often use the package manager to remove superfluous packages stored in cache, freeing up disk space)

rpm does not appear to have this facility, but other package managers do.

For example, the yum package manager stores downloaded package files (*.rpm) in /var/cache/yum directory tree, and the command

```
# yum clean packages
```

will remove downloaded packages that have been installed or updated.

Tips and tricks with .deb packages. (Ubuntu, Debian etc)

Checking package integrity

If we run synaptic, there is no diagnostic option to check the package integrity, although there is the menu item

```
Edit -> Fix broken packages.
```

Note that there is a status line at the bottom of the synaptic screen which show the number and status of available/installed packages, eg/.

```
25142 packages listed, 1274 installed, 0 broken, 0 install/upgrade, 0 remove
```

Note: You cannot run synaptic and an apt-get shell command which accesses the package database at the same time - you will get error message something like ...

```
E: Could not get lock /var/lib/dpkg/lock - open (11 Resource temporarily unavailable)
E: Unable to lock the administration directory (/var/lib/dpkg/), is another process using it?
```

To get a list of installed packages, use dpkg

```
$ dpkg -l | more
```

Note: use a wide console for ease of reading.

To check for package problems ..

```
# apt-get check [package name]
```

To reclaim some disk space - check /var/cache/apt/archives. If there are many packages and you have no package problems listed, and updates have successfully completed, then use

```
# apt-get clean
```

to delete the downloaded package archive contents. Note that there is a subdirectory /var/cache/apt/archives/partial. This holds any partially downloaded package files - when you next update the download will commence from where the partial download stopped.