eCryptfs: A stacked cryptographic filesystem for Linux

eCryptfs is free software. Please see the file COPYING for details. For documentation, please see the files in the doc/ subdirectory. For building and installation instructions please see the INSTALL file.

Maintainer: Phillip Hellewell
Lead developer: Michael A. Halcrow <mhalcrow@us.ibm.com>
Developers: Michael C. Thompson Kent Yoder
Web Site: http://ecryptfs.sf.net

This software is currently undergoing development. Make sure to maintain a backup copy of any data you write into eCryptfs.

eCryptfs requires the userspace tools downloadable from the SourceForge site:

http://sourceforge.net/projects/ecryptfs/

Userspace requirements include:

- David Howells' userspace keyring headers and libraries (version 1.0 or higher), obtainable from http://people.redhat.com/~dhowells/keyutils/
- Libgcrypt

> **Note**
>
> In the beta/experimental releases of eCryptfs, when you upgrade eCryptfs, you should copy the files to an unencrypted location and then copy the files back into the new eCryptfs mount to migrate the files.

Mount-wide Passphrase

Create a new directory into which eCryptfs will write its encrypted files (i.e., /root/crypt). Then, create the mount point directory (i.e., /mnt/crypt). Now it's time to mount eCryptfs:

```
mount -t ecryptfs /root/crypt /mnt/crypt
```

You should be prompted for a passphrase and a salt (the salt may be blank).

Try writing a new file:

```
echo "Hello, World" > /mnt/crypt/hello.txt
```

The operation will complete. Notice that there is a new file in /root/crypt that is at least 12288 bytes in size (depending on your host page size). This is the encrypted underlying file for what you just wrote. To test reading, from start to finish, you need to clear the user session keyring:

keyctl clear @u

Then umount /mnt/crypt and mount again per the instructions given above.

```
cat /mnt/crypt/hello.txt
```

Notes

eCryptfs version 0.1 should only be mounted on (1) empty directories or (2) directories containing files only created by eCryptfs. If you mount a directory that has pre-existing files not created by eCryptfs, then behavior is undefined. Do not run eCryptfs in higher verbosity levels unless you are doing so for the sole purpose of debugging or development, since secret values will be written out to the system log in that case.

Mike Halcrow mhalcrow@us.ibm.com