NAME

ecryptfs-setup-private − setup an eCryptfs private directory.

SYNOPSIS

ecryptfs-setup-private [−f|−−force] [−w|−−wrapping] [−b|−−bootstrap] [−n|−−no-fnek] [−−nopwcheck] [−u|−−username USER] [−l|−−loginpass LOGINPASS] [−m|−−mountpass MOUNTPASS]

OPTIONS

Options available for the **ecryptfs-setup-private** command:

−f, −−force
Force overwriting of an existing setup

−w, −−wrapping
Use an independent wrapping passphrase, different from the login passphrase

−u, −−username USER
User to setup, default is current user if omitted

−l, −−loginpass LOGINPASS
System passphrase for USER, used to wrap MOUNTPASS, will interactively prompt if omitted

−m, −−mountpass MOUNTPASS
Passphrase for mounting the ecryptfs directory, default is 16 bytes from /dev/random if omitted

−b, −−bootstrap
Bootstrap a new user's entire home directory

−−undo
Display instructions on how to undo an encrypted private setup

−n, −−no−fnek
Do not encrypt filenames; otherwise, filenames will be encrypted on systems which support filename encryption

−−nopwcheck
Do not check the validity of the specified login password (useful for LDAP user accounts)

−−noautomount
Setup this user such that the encrypted private directory is not automatically mounted on login

−−noautoumount
Setup this user such that the encrypted private directory is not automatically unmounted at logout

DESCRIPTION

ecryptfs-setup-private is a program that sets up a private cryptographic mountpoint for a non-root user.

Be sure to properly escape your parameters according to your shell's special character nuances, and also surround the parameters by double quotes, if necessary. Any of the parameters may be:

1) exported as environment variables
2) specified on the command line
3) left empty and interactively prompted

The user SHOULD ABSOLUTELY RECORD THE MOUNT PASSPHRASE AND STORE IN A SAFE LOCATION. If the mount passphase file is lost, or the mount passphrase is forgotten, THERE IS NO WAY TO RECOVER THE ENCRYPTED DATA.

Using the values of USER, MOUNTPASS, and LOGINPASS, **ecryptfs-setup-private** will:

- Create ~/.Private (permission 700)
- Create ~/Private (permission 500)
- Backup any existing wrapped passphrases
- Use LOGINPASS to wrap and encrypt MOUNTPASS
- Write to ~/.ecryptfs/wrapped-passphrase
- Add the passphrase to the current keyring
- Write the passphrase signature to ~/.ecryptfs/Private.sig
- Test the cryptographic mount with a few reads and writes

The system administrator can add the pam_ecryptfs.so module to the PAM stack which will automatically use the login passphrase to unwrap the mount passphrase, add the passphrase to the user's kernel keyring, and automatically perform the mount. See **pam_ecryptfs**(8).

FILES

~/.ecryptfs/auto-mount

~/.Private - underlying directory containing encrypted data

~/Private - mountpoint containing decrypted data (when mounted)

~/.ecryptfs/Private.sig - file containing signature of mountpoint passphrase

~/.ecryptfs/Private.mnt - file containing path of the private directory mountpoint

~/.ecryptfs/wrapped-passphrase - file containing the mount passphrase, wrapped with the login passphrase

~/.ecryptfs/wrapping-independent - this file exists if the wrapping passphrase is independent from login passphrase

SEE ALSO

ecryptfs-rewrap-passphrase(1), **mount.ecryptfs_private**(1), **pam_ecryptfs**(8), **umount.ecryptfs_private**(1)

/usr/share/doc/ecryptfs-utils/ecryptfs-faq.html

http://ecryptfs.org/

AUTHOR

This manpage and the **ecryptfs-setup-private** utility was written by Dustin Kirkland <kirkland@ubuntu.com> for Ubuntu systems (but may be used by others). Permission is granted to copy, distribute and/or modify this document under the terms of the GNU General Public License, Version 2 or any later version published by the Free Software Foundation.

On Debian and Ubuntu systems, the complete text of the GNU General Public License can be found in /usr/share/common-licenses/GPL.