GNU PGP

Remco Hobo

October 4, 2004

1. Download the source and install it with ./configure, make and make install Generate a key:gpg –gen-key Export the key for publishing: gpg –export -a -o root.key Submit it to a key server i.e. http://keyserver.veridis.com/en/ To revoke a key: gpg –import revokekey gpg –keyserver www.keyserver.net –send-keys Remco

2. The second excercise was to make trusted certificates

   - Generate a key: ssh-keygen -t rsa
   - Upload the key to the remote ssh server: scp /.ssh/id_rsa.pub remote:
   - Move the key to the appropriate place: Log into the remote host cat /id_rsa.pub ¿¿ /.ssh/authorized_keys2
   - Logout and test.

3. To set up using a remote X program, edit the /ec/ssh/sshd_config and edit the following lines to these values:

   - X11Forwarding yes
   - X11DisplayOffset 10
   - X11UseLocalhost yes
   - When you log in, you can take over the X server

4. Add the following line to the crontab to securely run a backup script:

   0 5 * * * username /script/backup 1¿/dev/null

   This will run the script under the appropriate user, preferably this is not root, but a user, without login rights, and has only rights, to the appropriate files.

5. Kmail setup

   The following packages must be installed to let Kmail send encrypted messages:

- Pth (¿= 1.3.7), usually packaged as libpth-devel (libpth-dev on Debian)
- gpg-error (¿= 0.7), usually packaged as libgpg-error-devel (libgpg-error-dev on Debian)
- libgcrypt (¿= 1.1.94), usually packaged as libgcrypt-devel (libgcrypt11-dev in Debian)
- libassuan (¿= 0.6.6), usually packaged as libassuan-devel (libassuan-dev in Debian)
- After this, set the appropriate certificates on in the KMail-config settings.

6. Encrypted filesystems One way to encrypt a filesystem, is to run a daemon, which encrypts data, using a private key. This has two disadvantages:
   - Random loss of data can occur, when a user loses his/her private key.
   - Integrity checks are much more difficult.