Laud Mills

Part A: Using GPG

GNU Privacy Guard (GPG) can be used to encrypt, to sign or both at the same time. Encryption can be performed either with a single key (symmetric) or a key pair (asymmetric or public encryption).

Tutorial: https://medium.com/@VitaliyMatiyash/gpg-tutorial-bb6589c650ce

1. Log in as root. Long list all files in your home with “ls -la” and keep in mind no directory ~/.gnupg is listed. Then, create a new key pair by executing “gpg --gen-key”. Make sure your screenshot includes the last messages and information.

A computer screen shot of a computer

Description automatically generated

1. Prepare a file pretending to be a confidential document you want to keep for yourself. Execute “echo This is confidential > document.txt”. Encrypt the file and show the command you used.

gpg --encrypt --recipient laudmills@yahoo.com document.txt

1. Use the *cat* command to display the contents of both the original and encrypted files. Capture screenshots.

A computer screen shot of a computer screen

Description automatically generated

1. List all keys in your key ring.

A computer screen with white text

Description automatically generated

1. Delete the original unencrypted file, then use an appropriate gpg command to decrypt the encrypted file. Capture a screenshot.

A computer screen with text

Description automatically generated

1. Someone who works with you in a project sent you an encrypted file as an attachment in an email, along with the passphrase to decrypt it. What do you think of this method of communicating confidential information? What your recommendation would be in this case?

This practice would not be the best because anyone that gets access to this email would be able to access the information on the file with ease. It could be best to use different mediums to communicate the encrypted file and the passphrase. For instance, the encrypted file could be sent as an attachment through email and the passphrase through secured message like whatsapp or phone call. The reason here is that if the encrypted file and passphrase both end up in the same place for an unauthorized person to access then the need for encryption is defeated.