Exprimenting with PGP

Hui Chen
Computer Science
Virginia State University, Virginia 23806
E-mail: huichen (AT) ieee.org

Written on September 21, 2015
Lastly revised on November 4, 2015
Revision: 140:f7cf97098528

PGP uses a certificate-based key management infrastructure for users' public key. However, its certificates and key management structure differs from X.509 public key infrastructure in several important ways that include [1],

- unlike X.509, a certificate may contain multiple signatures;
- also unlike X.509, a notion of *trust* is embedded in each signature and the signatures for a single key may have different level of trust;
- and it allows an arbitrary arrangement of certifiers and relies on each individual's knowledge of the certifiers while in X.509 PKI the public key of the root is known out of band.

The objective is to help readers gain familiarity with concepts pertaining to the PGP certificate signature chains and the *Web of Trust*, and be able to use PGP in practice.

OpenPGP is a PGP specification standardized by the IETF OpenPGP working group [2–4]. In this lab, we use GnuPG, an implementation of OpenPGP specified in IETF RFC 4880 [2].

1 Experiment Environment

The instructor prepares a Debian Linux virtual machine for this lab. The virtual machine is smaller in size (about 200MB in a compressed archive) and use much less RAM (64 MB). The instructor refers this virtual machine to as *the Debian Linux VM*. The Debian Linux VM is the same virtual machine for the PKI lab. If you have already been using the Debian Linux VM in your PKI lab, you do not need to download it again. Otherwise, download the Debian Linux virtual machine from either Dropbox or OneDrive.

The Debian Linux VM is in a 7-zip compressed archive file. You can use the 7-zip file archiver to extract the virtual machine.

1.1 GnuPG

GnuPG is an implementation of OpenPGP specified in IETF RFC 4880 [2]. A few versions of GnuPG that are currently in use [5],

- *GnuPG stable (2.0)*. It is the modularized version of GnuPG supporting OpenPGP, S/MIME, and Secure Shell.
- *GnuPG modern (2.1)*. It is a new version with enhanced features like support for Elliptic Curve Cryptography. It will eventually replace the current stable (2.0)
- *GnuPG classic (1.4)*. It is the old, single binary version which may be build even on ancient Unix platforms. It has no dependencies on the above listed libraries or the Pinnetry. However, it lacks many modern features.
- *Pinentry*. It is a collection of passphrase entry dialogs which is required for almost all usages of GnuPG stable or modern (2.x).
- *GPGME*. It is the standard library to access GnuPG functions from programming languages.
- *GPA*. It is a graphical frontend to GnuPG.
- *Dirmngr*. It is an optional tool for use with GnuPG stable (2.0). It is already included in GnuPG modern (2.1)

In the Debian Linux VM, the GnuPG classic has already been installed. To verify that GnuPG is installed, the version, and the supported features of the installed GnuPG, we can run gnupg as follows,

```
debian@dVMsec:~$ gpg --version
gpg (GnuPG) 1.4.18
Copyright (C) 2014 Free Software Foundation, Inc.
License GPLv3+: GNU GPL version 3 or later <http://gnu.org/licenses/gpl.html>
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Home: ~/.gnupg
Supported algorithms:
Pubkey: RSA, RSA-E, RSA-S, ELG-E, DSA
Cipher: IDEA, 3DES, CAST5, BLOWFISH, AES, AES192, AES256, TWOFISH,
        CAMELLIA128, CAMELLIA192, CAMELLIA256
        Hash: MD5, SHA1, RIPEMD160, SHA256, SHA384, SHA512, SHA224
        Compression: Uncompressed, ZIP, ZLIB, BZIP2
debian@dVMsec:~$
```

2 Assignment

Students are required to complete the tasks below and write a concise report on the tasks.

For GnuPG, you may use [6] as a tutorial. We refer this reference as *the tutorial* in the following.

Note that the tutorial uses GnuPG 2, i.e., gpg2. However, most steps would work by replacing gpg2 by gpg in most examples in the tutorial. That is to say, it is not necessary to install *GnuPG 2* and you may skip section "Install GnuPG2".

2.1 Task 1. Explaining GnuPG Version Information

Run `gpg --version` and explain *concisely* the information displayed. Provide references if necessary, such as, URLs to web pages, citations and bibliography entries of books and research papers.

2.2 Task 2. Generating Key Pair

Use GnuPG to generate a public key and private key pair.

2.3 Task 3. Sending Public Key to Keyserver

Use GnuPG to send the public key generated in subsection 2.2.

2.4 Creating a Revocation Certificate

Use GnuPG to create a revocation certificate.

2.5 Signing Keys

Use GnuPG to sign another user's public key. You should obtain the key id from your classmate and sign her or his key.

2.6 Signing Keys with Trust Value

Do research. Base on your research,

- list possible values of *ownertrusts*;
- describe the steps that assign an *ownertrust* value to a key;
- and adjust your trust on the key owner's you are signing to "I trust marginally".

2.7 Encrypting Documents

Use GnuPG to encipher a document.

2.8 Signing and Verifying Documents

Use GnuPG to sign and verify documents.

2.9 Sending "Secrete" Document to Instructor

You will write a concise step to the instructor and send the instructor a "secrete" document. Following your instruction, the instructor will reply you if your instruction is well-written and the instructor can decrypt your secrete document. You may refer to the stories on Edward Snowden for motivation,

- Micah Lee, Ed Snowden Taught Me to Smuggle Secrets Past Incredible Danger. Now I Teach You., The Intercept, October 28, 2014

- Janet Reitman, Snowden and Greenwald: The Men Who Leaked the Secrets, Rolling Stone, December 4, 2013

Note that in this task, you may need additional research. The instructor has the following additional requirement.

- As an additional requirement, you should *not* configure an e-mail client to send the "secrete" document. Instead, you should send the document using a regular e-mail client, such as, your university e-mail client. This allows us to understand the actual mechanism.
- You should provide a means to the instructor to authenticate yourself and the instructor and describe in your instruction to the instructor what this mechanism is, how your instructor should follow your instruction to authenticate you and how you authenticate the instructor. Again refer to the Edward Snowden stories for motivation.

References

[1] Matt Bishop. *Introduction to Computer Security*. Addison-Wesley Professional, 2004.

[2] J. Callas, L. Donnerhacke, H. Finney, D. Shaw, and R. Thayer. Openpgp message format. RFC 4880, RFC Editor, November 2007. http://www.rfc-editor.org/rfc/rfc4880.txt.

[3] M. Elkins, D. Del Torto, R. Levien, and T. Roessler. Mime security with openpgp. RFC 3156, RFC Editor, August 2001. http://www.rfc-editor.org/rfc/rfc3156.txt.

[4] IETF OpenPGP Working Group. Open specification for Pretty Good Privacy (openpgp). https://datatracker.ietf.org/wg/openpgp/charter/, retrieved October 15, 2015.

[5] The GnuPG Project. Download gnupg. https://www.gnupg.org/download/index.html, retrieved on October 15, 2015.

[6] Zachary Voase. Openpgp for complete beginners. http://zacharyvoase.com/2009/08/20/openpgp/, retrieved on October 15, 2015.