Global Information Assurance Certification Paper

Marc Duperrier
GSEC ver. 1.4b
Option 1

A corporate implementation of PGP

Introduction

In this short paper I will try to provide some information in order to help plan a corporate deployment of PGP. A brief explanation about how a standard PKI infrastructure is designed will be made, followed by a comparison between PKI and the PGP Web of Trust. A set of corporate requirements will then be analyzed and mapped to some PGP functionality. I will then try to provide some key elements that need to be taken into account for a successful deployment of PGP.

About the importance of certificates

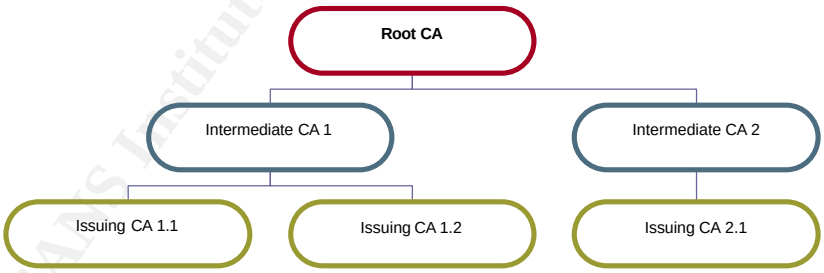
This paper will not discuss any cryptographic protocols details, nor will it discuss the basis of public key encryption. I still want to emphasis on an important concept: *certificates*. Please note that *certificate* is a term used in PKI world, and it usually refers to an X.509 V3 certificate. On the other hand, *signature* is a PGP term that in this paper refers to a Public Key signature. PGP Corporation defines a signature as "A digital code created with a private key. Signatures allow authentication of information by the process of signature verification. […]" in its PGP 8.0 for Windows User's Guide [PGP8ug] p.179

Why is certificate such an important topic? Because it's the base element on which we can build a trusted communication.

Cryptographers have done an incredible job in the past years to create strong algorithm and good implementation of it. In most cases breaking AES-256 or an RSA 2048 bits key is usually not the problem, but knowing WHO to trust when sending a mail through the Internet is still a challenge. And this trust is based on certificates or signatures.

The PKI trust model

The PKI answer to this certificate trust problem is to use a somewhat easy to understand hierarchy. Every end-entity, either user or computer, is certified by an issuing CA. The issuing CA is in turn, either certified by an intermediate CA or directly by a root CA.



We can quickly see that trusting the Root CA implies trusting the whole hierarchy as well. As a consequence, we can easily see that the Root CA needs to be highly protected, in order to be trusted.

For the protection, physical security measure, such as offline CA and hardware devices such as nCipher's nShield or AEP's Keyper are often used.

- http://www.ncipher.com/nshield/index.html
- http://www.aep.ie/products_sure_keyper.htm

The Certificate Practice Statement (CPS) is a formal statement made by the CA organization about how it issues, maintains and revokes certificates. Its goal is to provide enough information to end users so that they can trust this specific CA

A few example of CPS from major CA are referenced below:

- http://www.verisign.com/repository/CPS2.1/cps2-1.pdf
- http://www.globalsign.net/repository/CPSv4_1.pdf
- http://www.thawte.com/downloads/Thawte_CPS_2.pdf

The main elements contained are usually:

- What controls are in place to verify a subject identity?
- The revocation conditions, by the user or by the issuing authority.
- How frequently will Certificate Revocation List (CRL) be issued?
- Administrative and technical procedures to protect private keys.
- And, if applicable, the cost and liability associated with the certificate.

In a corporate deployment, a similar set of rules can usually be implemented as part of the Security Policy document.

So, in theory, any user should read the CPS related to a specific certificate to decide, it is worth any confidence. On top of that, users should evaluate if the CA's company itself is worth confidence!

It's obviously not realistic to assume that it can work that way. Netscape, Microsoft and others have clearly understood the problem and they pre-configure their software with a list of trusted Root certificates. This is a practical approach, but do you really trust any of those hundreds of default certificate? Even when doing your online banking?

I personally do not, by any means.

Let's see now how PGP tries to solve the problem.

The PGP Web of Trust

PGP uses the notion of "web of trust" where any user can sign the certificate of any other users and then export this signature. If Alice sign Bob's public key and export it, it *should* indicate to the rest of the world that Alice as verified the relation between Bob and his private key. Does this help? Are you now confident enough to send confidential information to Bob even if you never meet him?

I would certainly not, especially if I have never heard about Alice in the real world.

On the other hand, let's assume that you personally know Alice and you therefore have already a signed and trusted copy of her public key. You also know that Alice always take great care to verify users identity before signing any key. You should now have enough confidence in Bob's identity to send him confidential -but encrypted- information.

This is basically the Web of Trust. It relies on quite some assumptions about knowing people and understanding how they sign other people's keys; this is therefore not easily translated in a corporate environment.

Some very good analyses are available on the web about the mathematics behind this web of trust like the one from Drew Streib: http://www.dtype.org/keyanalyze/. Unfortunately this does not (and cannot) take into account the "Personal CPS" used by each individual.

In the PGP community, some users already have some sort of "personal CPS"! This can be as simple as "I do exchange signature only in face to face meeting"

Or it can be more detailed as in the following two examples:

> "I would like to see a government-issued photo ID, preferably a passport, and one additional piece of ID such as a student ID or credit card with a signature that I can compare to the signature on the photo ID" Nathan Lutchansky http://www.litech.org/~lutchann/signinginfo.html January 2003
>
> "[…] we can arrange to meet, verify identities, and exchange key fingerprints. You should bring a photo ID with you." Stefan Gmoser http://bau2.uibk.ac.at/sg/pgp.html March 2003

What we can see here, again, is the fact that we need a sort of CPS, a formal statement which needs to provide the necessary information in order to trust a certificate or a key signature.

Why chose a PGP solution

With all that have been said on the easier trust model of standard PKI, why would a corporation prefer a PGP deployment?

Well, there are many reasons which can be listed here, but the main ones in my opinion are:

1. PGP is a de facto standard on the internet, it is even a formal standard as per [RFC2440], Open PGP. Most security newsletter and alert service are PGP signed and many serious software are now signed to prove their unmodified origin.
2. Easy integration in virtually any applications, any file can be signed and/or encrypted, as can any text within most application.
3. Proven security, PGP is widely recognized for its elegant design and good coding practice. A large number of very renowned cryptographers have scrutinized PGP sources code and algorithm.
4. Users can, is most cases, retain a strong control of his/her private key. This can add to the user's confidence in the solution, especially when hardware token are in use.
5. PGP can be used seamlessly inside the corporate network or for any correspondences to or from the Internet.

I'm not including here cost as an argument because both the PKI and PGP solution can be implemented with a highly variable budget, ranging from GnuPG to outsourced PKI.

Corporate needs

Of course, I will not try to summarize here all IT security concerns that corporate organization are facing in today's world! Not even all these can be solved by encryption. Obviously it depends on the corporate size, business sector, and so on. For the scope of this paper, I will focus only on those issues that a PGP deployment can address.

- Ability to exchange encrypted emails, both inside the corporation and with external partners.
- Non-repudiation of digital communication
- Secure storage of information, either single files or a whole set of information, as with laptop.
- Recoverability, most corporations will not accept the risk of being unable to open critical document if the owner is unavailable.

Other usage of encryption, as in SSL for web and intra server communication, SSH, VPN and more will not be discussed here.

What solution can a PGP deployment provide to those needs?

Many solutions exist to exchange confidential mails within a corporate network, but PGP is certainly the best solution when you need to communicate with external business partners. If your external correspondents do not yet have any email encryption solution, they will certainly be less reluctant to implement a PGP solution (or GnuPG if they are really short on money) than a solution based on PKI, S/MIME and X.509 certificates.

Non-repudiation is not yet fully achievable by cryptographic means, but the closest we can get do rely on: 1) a strong relation between a physical user and its key (a good signature) and 2) a properly protected private key. An interesting discussion on this subject, by Bruce Schneier can be found at: http://www.counterpane.com/crypto-gram-0011.html

Secure storage of single file can be easily achieved with PGP. Any sort of files can be encrypted, for one or many users. A very elegant solution for storing many files is to use PGPdisk. Again either a single user or many of them can read the set of files, not writing simultaneously. A hardware token granting access to a PGPdisk on a laptop is an excellent protection for your data. Of course leaving the token in the laptop bag and the PIN code written on a post-it will not achieve the desired goal. Not a lot of technical solution here, education and policy are your only hope.

Recoverability needs have to be balanced with any added risk. ADK (Additional Decryption Key) is a good solution to insure that you can always recover your corporate data. Such solution need to be well planned and documented. The keys themselves need the strongest procedural and technical protection. Another solution to consider is the protection of the *private keys* instead, or in complement, of the *data* protection provided by implementing ADK. This can be achieved by implementing a Key Recovery Server. Standard backup of the private keys is not a recommended solution, and do not help, anyway, in case of lost passphrase.

Preparing the deployment

Now that we have built the framework, we should be able to go one step further and really plan our PGP deployment.

First, we need a CPS! No surprise here, I believe a good CPS will go a long way to help a smooth deployment. This document must be an integral part of the corporate security policy, and should clearly state how company signatures are issued and revoked. The full CPS or, at least, part of it should be published on the corporate web site for review by business partners. Providing enough information about the key protection and issuing process is the best way to convince partner to trust the *Company Signing Key* as a trusted introducer, and therefore all employee keys. The *Signing Key* should also be verifiable, by providing, at least, the fingerprint on a secure web page.

Some question need to be answered and will help to create a good document:

1. How will the corporate signing key be managed? The recommended solution could be to use key splitting and requesting at least 2 trusted individuals –let say security personnel- to verify and sign any new keys.
2. Who owns the key? Of course the user will in any case be responsible for it's key, but a corporate created key can enforce a fixed format, and include ADK, such a "company owned" key will be fully revoked when the employee leaves. Another option would be to "take on board" users existing key (if applicable) and simply sign it with a corporate signing key. This solution could be preferred by existing PGP users but do not seems to be a good solution for large deployment.
3. If ADK are to be used, again a formal process has to be established in advance. Again key splitting is the option to follow, but 3 senior executives could then be granting such an access. Remember that the use of the ADK should be extremely rare and the consequences are important.
4. Where will the keys be stored? Many possibilities here, but implementing a PGP key server is an easy win, if no other directory is readily available. Defining the keys availability for external partner also need to be clearly defined.
5. How will the private keys be stored? Where practical the use of hardware cryptographic devices is highly recommended. Both Smartcard and USB token provide very secure storage.

Two factors is better than one

Of course, using a Two Factors authentication system is a huge security improvement over password or even passphrase. Two quite common solutions can easily be used with PGP: Smartcards or USB Token. Both provide safe storage of the private key and are equally easy to carry on and to use. USB token has the added advantage not to require a reader on each PC.

Good smartcards, readers and USB token can be purchased from:

- Schlumberger http://www.slb.com/smartcards
- GemPlus http://www.gemplus.com
- DataKey http://www.datakey.com/cardpage
- Rainbow http://www.rainbow.com
- Aladdin http://www.eAladdin.com/eToken
- Spyrus http://www.spyrus.com

6. Should a key recovery server be used? The loss of a private key prevents access to all information encrypted for this key. If ADK is not implemented, and even if it is, a key recovery server may be a good safety net to implement. Users have to provide 5 questions and the 5 related answers; the private key is then split and securely stored. At least three correct answers are required to reconstruct the key.
7. Define if PGPdisks need to be used and how. A PGPdisk allows encrypted storage of numbers of files in a very easy way. Once "mounted", a PGPdisk file behave as another drive, which can contain folders and files. ADK can be enforced and this is recommended.

You now need a repository for your public keys. If you do not yet have an X.509 directory service available, like Microsoft AD, installing PGP Key server is a good solution. This product is easy to install and to operate, but a careful reading of the documentation is still highly recommended.

You also will need to publish some keys on the Internet. A minimalist solution can simply be to use public PGP servers, and a secure web page containing the details of your corporate signing key. More evolved design includes an external key server, replicating with others key server on the Internet.

Creating your corporate signing key is the next step. Take great care to do this operation on a trusted machine and immediately split the key as defined earlier in the physical presence of the share owners. A properly secured backup copy of this key is important. I could also be wise to add one or two revokers.

Business Partner signing key?

You have done a great job to ensure that all keys signed by your "Corporate Signing Key" are properly authenticated, and stored on your key server. You have told your External partners (through your published CPS) that they can trust all such signed keys. Now how will your internal users recognize the valid keys of your external business partners?

An elegant solution would be for your skilled security staff to do the verification *by any appropriate means*, then sign those key with a "Business Partner Signing key" and store them on your key server.

It would certainly help your user to recognize those validated business partners.

Is your security policy document growing?

The PGP admin program

Now that the CPS is written, and the corporate keys created and securely stored, we can prepare the PGP software deployment with the PGP admin program.

This program is used to pre-configure PGP in a very detailed way. Reading your CPS and keeping an eye open on your security policy documentation should make this configuration very easy.

The main setting to configure includes:

1. The ADK settings
2. The passphrase, if you do not use hardware devices to store your private keys.
3. The type of key that you want to use. As already said, the key and encryption protocols discussion is out of the scope of this paper.
4. The default key which have to be present in the user's keyrings
5. The key revocation setting.
6. Interface and parameters locking, so that users cannot change setting which are parts of the CPS or security policy.
7. The plug-in and others installation option.
8. As the above defined settings can change over time, an update mechanism can be implemented at initial installation time. This is highly recommended, as it can save a lot a time latter on.

PGPadmin can now be used to generate a custom installation of PGP that you can now deploy using your favorite method.

Depending on your user's technical skills, you may want to write a small PGP users guide. Not replacing the PGP documentation, but it's important that they understand which keys should be trusted and which one should not. How they should validate any new contact or ask Security staff to do so, and any such practical issues.

That should be it; you are now ready to install your pre-configured client.

Do we really need to choose between PGP and a PKI?

No, indeed. A very attractive new capability of PGP 8.0 is its support for X.509 certificate and CA. It can therefore neatly integrate in a PKI environment; to quote PGP Corporation:

> *There is a common misconception that an organization must choose between an X.509 PKI and PGP. The thought is that if one has invested in Entrust, Verisign OnSite, Microsoft's PKI, or others, that it precludes using PGP. Nothing could be further from the truth!*
>
> Using an X.509 PKI with PGP 8, PGP Corporation December 2002
> http://www.pgp.com/products/whitepapers/PGP8X509.pdf

So if a PKI is already available, the PGP installation can be easily integrated. Of course this possibility adds quite some new and interesting scenarios.

Conclusion

I hope information presented in this paper was useful. I want to stress once more that a published CPS is very important to give people the trust needed to start using encryption technologies. The encryption products available on the market are good, not all, but PGP certainly is a very good one. Unfortunately too many times the lack of information about the process used to issue a certificates or signature result in an uncomfortable posture. A corporate environment is, in a sense, an easier place to start building a good "web of trust" than the Internet is.

Glossary

CA Certificate Authority,
Certificate or **key signature** is a digital code that allows authenticating information based on a mathematical computation.
CPS Certificate Practice Statement, formal document published by an issuing authority, and detailing how certificates are issued, maintained and revoked.
PGP The encryption software originally written by Phil Zimmermann and that provide a "Pretty Good Privacy"
Root CA The CA at the top of a Certificate hierarchy. Technically it's a self signed X.509 certificate.
Web of Trust: The notion in the PGP community of users which represents the "web" of signatures between users. See reference below for much more detailed explanations.
Public Key: Public part of the key for a so called "Public key" system. This key is used to encrypt messages and verify signatures.
Private Key: This is the private part of a "public key" system, used to decrypt messages and sign documents. This key need strong protection.
Encryption: The mathematical process by which a clear text (readable) is transformed in a cipher text using en encryption key.

Decryption: Operation to retrieve the clear text from the cipher text. Again a key is needed for the operation to succeed.
Electronic Signature Electronic signatures provide a similar proof of authenticity as a traditional paper signature. It add another important feature, it's can validate that the information has not been altered since it has been signed.

References:

[PGP8ug] PGP Corporation
PGP 8.0 For Windows User's Guide, November 2002

[RFC2440] OpenPGP Message Format, Novembre 1998
http://www.ietf.org/rfc/rfc2440.txt

PGP Corporation, PGP Enterprise Configuration & Deployment, 2003
http://www.pgp.com/products/enterpriseconfig.html

The GNU Privacy Guard
http://www.gnupg.org/

Phil Zimmermann's Home Page
http://www.philzimmermann.com/

Drew Streib, Key Analyze,
http://www.dtype.org/keyanalyze/

Neal McBurnett, PGP Web of Trust Statistics, July 2001
http://bcn.boulder.co.us/~neal/pgpstat/

Keith Parkins, PGP - the Web of Trust, September 1997
http://www.heureka.clara.net/sunrise/pgpweb.htm

PGP Corporation, PGP White papers
http://www.pgp.com/display.php?pageID=105

Patrick Feisthammel, Explanation of the web of trust of PGP, June 2002
http://www.rubin.ch/pgp/weboftrust.en.html

Carl Ellison, SPKI/SDSI and the Web of Trust, April 2001
http://world.std.com/~cme/html/web.html

Bruce Schneier, Crypto-Gram: November 15, 2000
http://www.counterpane.com/crypto-gram-0011.html