

Network Security Workshop

Cryptography Applications / PGP



CONFERENCE & APNIC REGIONAL MEETING

Track 2: Network Security



Fakrul (Pappu) Alam
bdHUB Limited
fakrul@bdhub.com

Security issues for E-mail on the

Confidentiality	Network admin can read your e-mail. Webmail provider can read your e-mail. LAN user may read your e-mail by monitoring tool. Even in some hotel, I could have chance to read
Integrity	E-mail contents may be changed by some attacker on the network.
Authenticity	Easy to set any e-mail headers like "From". Any other e-mail headers can be set anything you want. Difficult to know it is true.

Targeted Attack

- Attacks on information security which seek to affect a specific organization
- or group, rather than indiscriminately. Some may be customized for a specific target organization or group.
 - An e-mail with suspicious file attached
 - Executable binary
 - Word document file
 - Database application file

Targeted Attack

To: your e-mail address

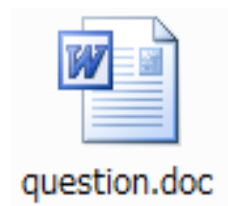
From: Fakrul Alam fakrul@dhakacom.com

Subject: my request

Hello,

I have been looking for someone who can answer questions of the attached file. I hope you can do that and reply me.

Thanks !



Example of Spoof Mail

```
by spamwall.dhakacom.com (Postfix) with ESMTP id 70EF3BA13F8
for <fakrul@dhakacom.com>; Tue, 12 Jun 2012 04:39:04 +0600 (BDT)
Received: (qmail 62722 invoked from network); 12 Jun 2012 05:34:59 +0700
X-Spam-Level: *****
X-Spam-Status: No, hits=7.6 required=8.0
tests=MISSING_HEADERS,RAZOR2_CF_RANGE_51_100,RAZOR2_CF_RANGE_E8_51_100,RAZOR2_CHECK,SUBJ_ALI
X-Spam-Check-By: smtp3.dnet.net.id
Received: from smtp3.dnet.net.id (HELO newwebmail.dnet.net.id) (202.148.1.233)
by smtp3.dnet.net.id (qpsmtpd/0.84) with ESMTP; Tue, 12 Jun 2012 05:34:54 +0700
Received: from 94.41.250.182
(SquirrelMail authenticated user raphael@dnet.net.id)
by newwebmail.dnet.net.id with HTTP;
Tue, 12 Jun 2012 05:34:54 +0700 (WIT)
Message-ID: <751e890ca8b32f6b6ec8b26dd484fb71.squirrel@newwebmail.dnet.net.id>
Date: Tue, 12 Jun 2012 05:34:54 +0700 (WIT)
Subject: ACCOUNT TERMINATION
From: "Dhakacom.com Mail Manager" <webadmin@dhakacom.com>
User-Agent: SquirrelMail/1.4.16
MIME-Version: 1.0
Content-Type: text/plain; charset=iso-8859-1
Content-Transfer-Encoding: 8bit
X-Priority: 3 (Normal)
Importance: Normal
X-Virus-Checked: Checked by ClamAV on smtp3.dnet.net.id
X-dhakacom-MailScanner-ID: 70EF3BA13F8.7A916
X-dhakacom-MailScanner: Found to be clean
X-dhakacom-MailScanner-From: webadmin@dhakacom.com
CC: undisclosed-recipients;
```

Cryptography:

The Two Basic Encryption Techniques

- Symmetric and Asymmetric (public-key)
- The latter is widely accepted
- PGP is based on Asymmetric (Public-Key) Encryption

Symmetric Encryption

- Involves only one key, which is used by both the sender for encrypting and the recipient for decrypting
- Symmetric algorithms: blowfish, Triple-DES, AES (Advanced Encryption Standard), CAST (Carlisle Adams and Stafford Tavares), IDEA (International Data Encryption Algorithm, legally restricted, but the other algorithms may be freely used)
- Problem: the means of distributing the key

Asymmetric (Public-Key) Encryption

- Solves the problem of distributing keys by using one pair of complimentary keys, one public and the other private.
- Public: freely exchanged to others without fear of compromising security.
- Private: only you have access, should be carefully protected.
- A message is encrypted to a recipient using the recipient's public key, and it can only be decrypted using the corresponding private key.

Asymmetric Encryption Refresher

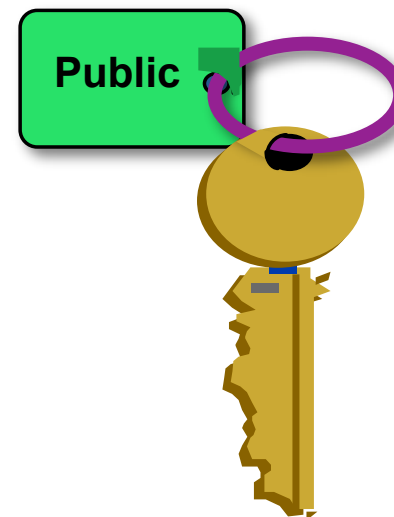
- One key mathematically related to the other.
- Public key can be generated from private key. But NOT vice versa.
- If you encrypt data with the public key, you need to private key to decrypt
- You can sign data with the private key and verify the signature using the public key

Keys

- Private key is kept SECRET.
- You should encrypt your private key with a symmetric passphrase.



- Public key is distributed.
- Anyone who needs to send you confidential data can use your public key



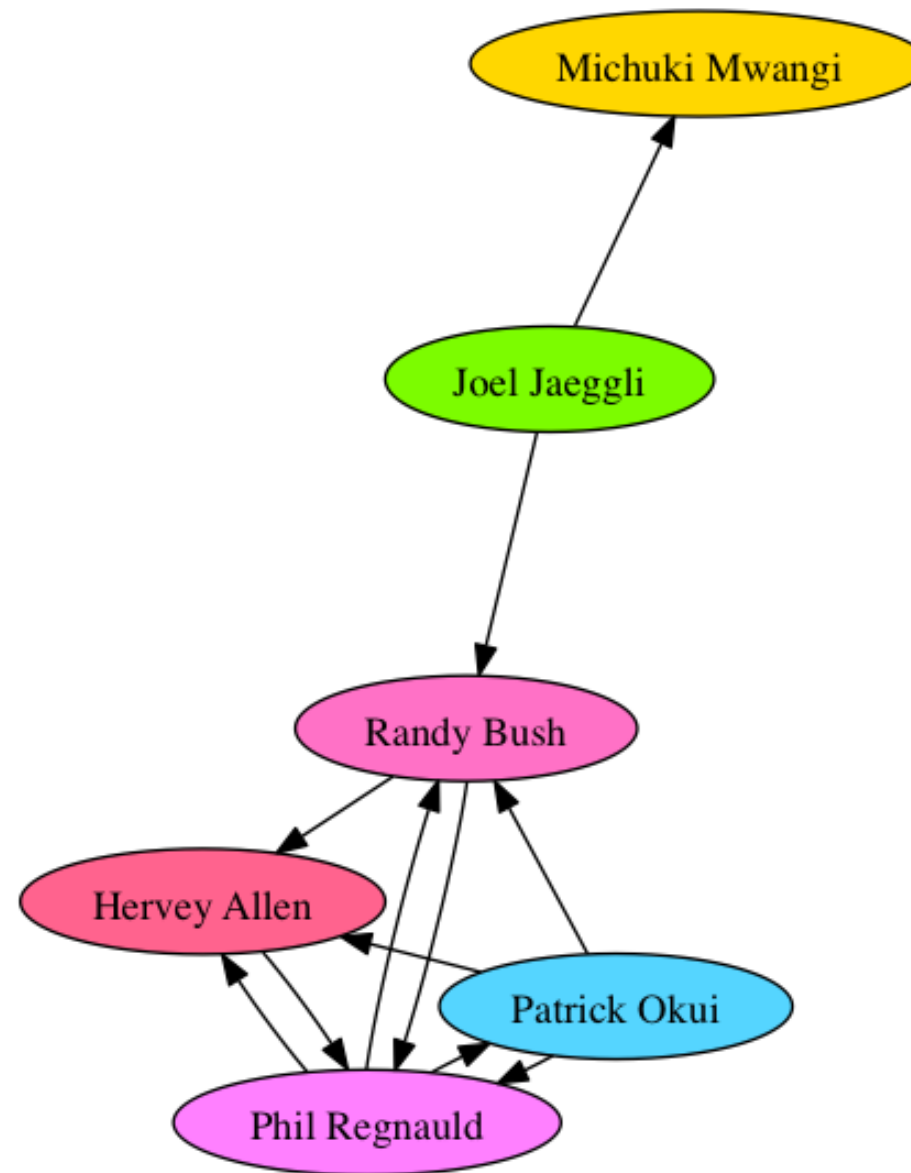
Signing & Encrypting

- Data is encrypted with a public key to be decrypted with the corresponding private key.
- Data can be signed with the private key to be verified by anyone who has the corresponding public key.
- Since public keys are data they can be signed too.

Trust

- Centralized / hierarchal trust – where certain globally trusted bodies sign keys for every one else.
- Decentralized webs of trust – where you pick who you trust yourself, and decide if you trust who those people trust in turn.
- Which works better for what reasons?

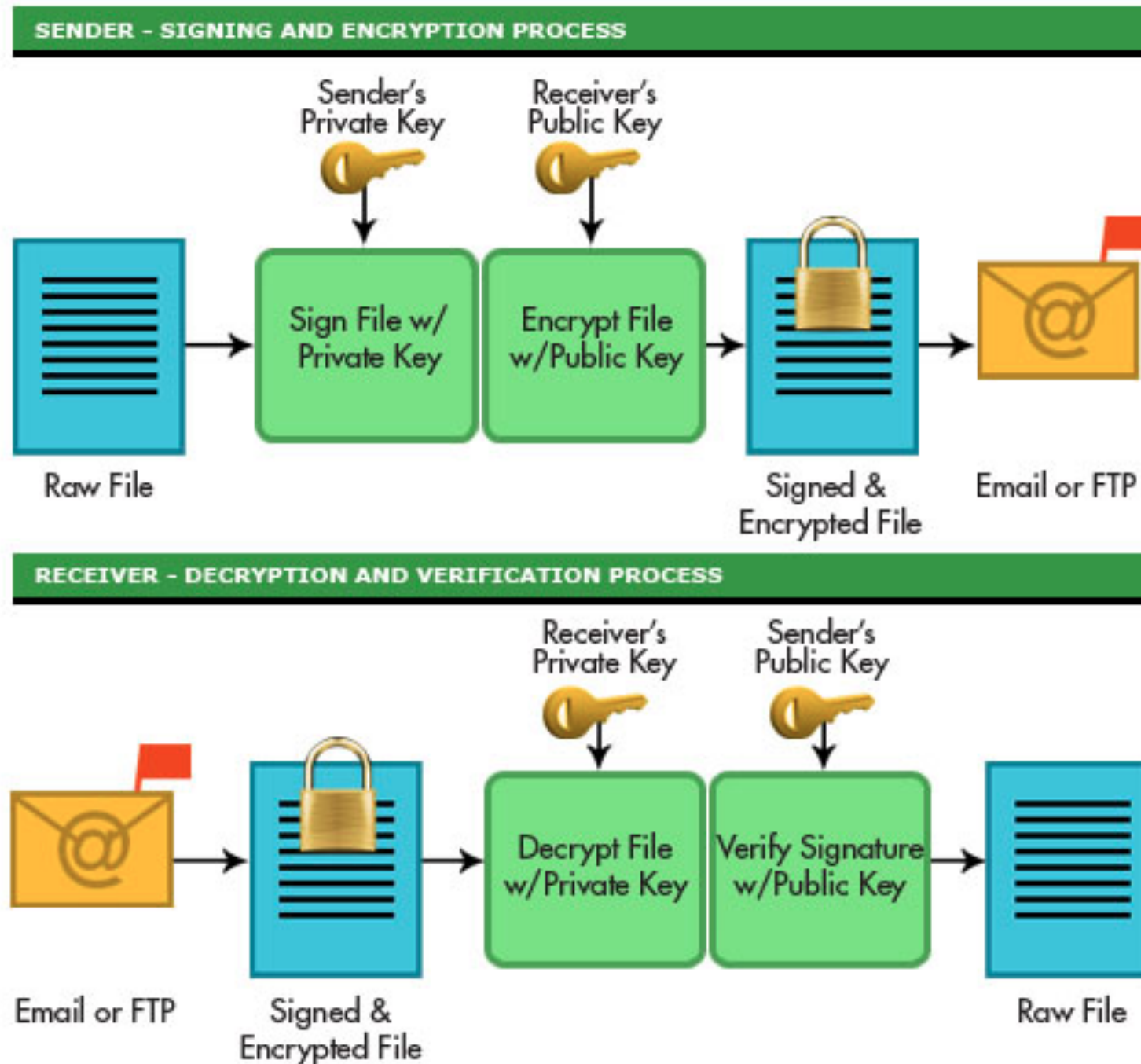
Sample Web of Trust



PGP by GnuPG

- Create your keys
 - Public key
 - Private key (secret key)
- Identify key by
 - Key ID (like 0x23AD8EF6)
- Verify others' public key by
 - Key fingerprint
- Find keys on PGP key servers
 - Like <http://pgp.mit.edu>

How PGP Works



Installing GnuPG Software

- Core software either commercial from pgp or opensource from gnupg.
 - <https://www.gpg4win.org/> for windows
 - <https://www.gpgtools.org/> for OS X
- Your package manager for Linux/UNIX
 - Source code from <https://www.gnupg.org/>

Key Management

- Using graphical tools based on what you installed above:
 - GPG Keychain Access for OS X
 - Kleopatra or GPA for windows
- Using the command line:
 - `gpg --gen-key`
- Generate a key – use your email address. The comment field can be left blank.

Key Management

- On printed media: published book or business cards:
- Digitally in email or using sneaker-net
- Online using the openpgp key servers.
- Still does not tell you if you trust the key.


Key Management

- Expiry dates ensure that if your private key is compromised they can only be used till they expire.
- Can be changed after creating the key.
- Before expiry, you need to create a new key, sign it with the old one, send the signed new one to everyone in your web of trust asking them to sign your new key.

Key Management - Revocation

- Used to mark a key as invalid before its expiry date.
- Always generate a revocation certificate as soon as you create your key.
- Do not keep your revocation certificate with your private key.
- `gpg --gen-revoke IDENTITY`

Key Management - Partying

- Key signing parties are ways to build webs of trust.
- Each participant carries identification, as well as a copy of their key fingerprint. (maybe some \$ as well )
- Each participant decides if they're going to sign another key based on their personal policy.
- Keys are easiest kept in a keyring on an openpgp keyserver in the aftermath of the party.

Interesting gpg commands

- Get help for gpg options
 - `gpg --help` AND `man gpg`
- Print the fingerprint of a particular key
 - `gpg --fingerprint IDENTITY`
- `IDENTITY` = email or PGP key ID
- Export a public key to an ASCII armored file.
 - `gpg -a --output my-public-key.asc --export IDENTITY`

Interesting gpg commands

- Import a key from a file into your keyring
 - `gpg --import public.asc`
- Import a key from a keyserver
 - `gpg --recv-keys --keyserver hkp://keys.gnupg.net`
- Send your key to a keyserver
 - `gpg --send-keys --keyserver hkp://keys.gnupg.net`
- Sign a key
 - `gpg --sign-key IDENTITY`