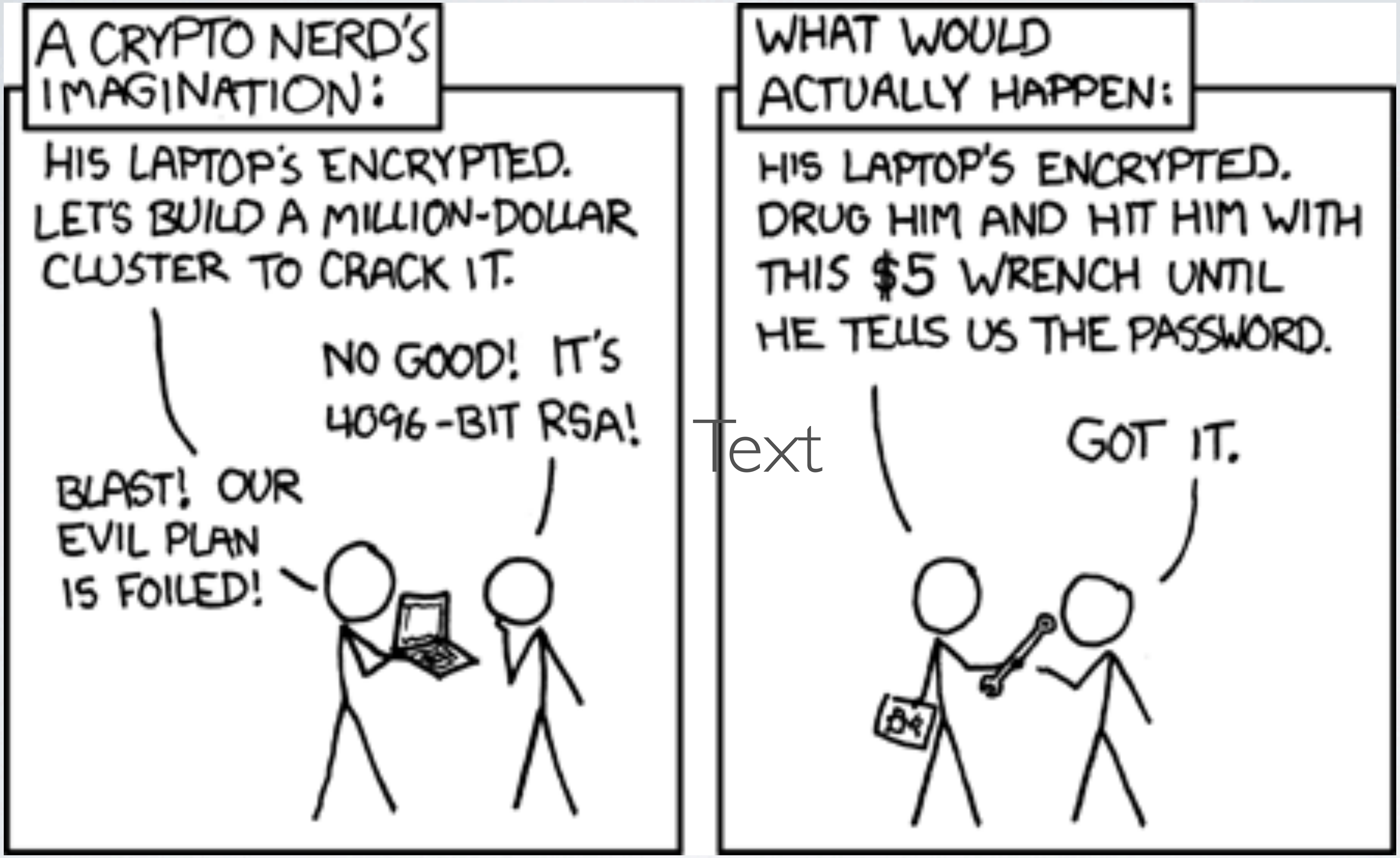
GPG BASICS

Carlos Perez carlos_perez@darkoperator.com

WHAT IS ENCRYPTION?

- Encryption encodes and scrambles data so it is difficult to obtain the original content unless a known secret is used to decipher it.
- The 2 main schemes of encryption are:
  - Symmetric - The same cryptographic key is used for both encryption and decryption of the data. It is the simplest form of encryption.
  - Public Key - Requires two separate keys, a secret key and a public key. Although different, the two parts of the key pair are mathematically linked. One key locks or encrypts the data , and the other unlocks or decrypts the data.


A CRYPTO NERD'S IMAGINATION:
HIS LAPTOP'S ENCRYPTED. LET'S BUILD A MILLION-DOLLAR CLUSTER TO CRACK IT.
NO GOOD! IT'S 4096-BIT RSA!
BLAST! OUR EVIL PLAN IS FOILED!
WHAT WOULD ACTUALLY HAPPEN:
HIS LAPTOP'S ENCRYPTED. DRUG HIM AND HIT HIM WITH THIS $5 WRENCH UNTIL HE TELLS US THE PASSWORD.
GOT IT.

Text

PGP

- PGP Stands for Pretty Good Privacy.
- It was initially created by Phil Zimmerman in 1991
- In 1997 OpenPGP was proposed to the IETF and in 2007 and accepted. It is currently RFC4880 http://tools.ietf.org/html/rfc4880 and it is fo

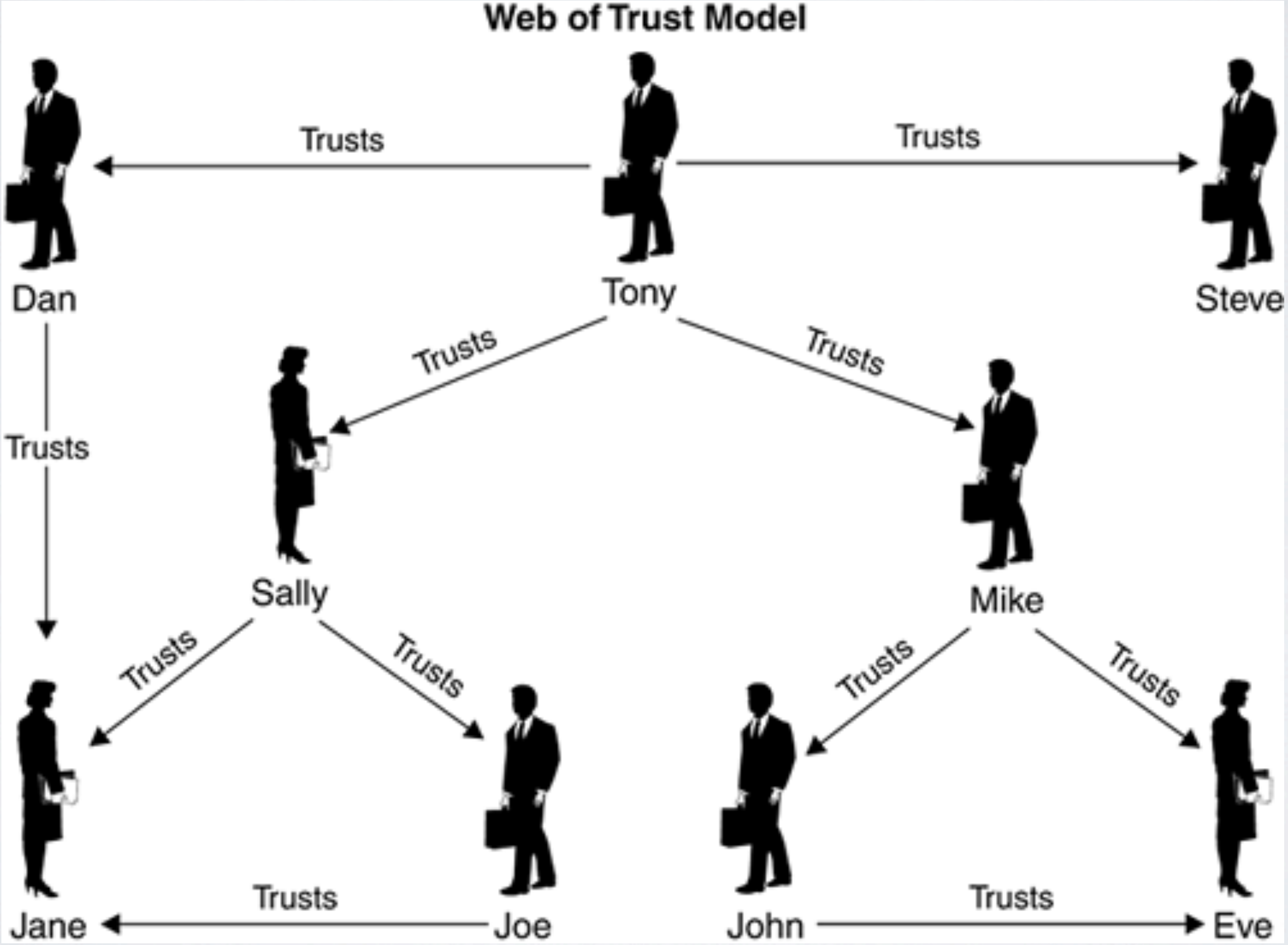
OPENPGP

- The standard covers strong public-key and symmetric cryptography to provide security services for electronic communications and data storage.
- These services are:
  - Confidentiality
  - Key management
  - Authentication
  - Digital signatures

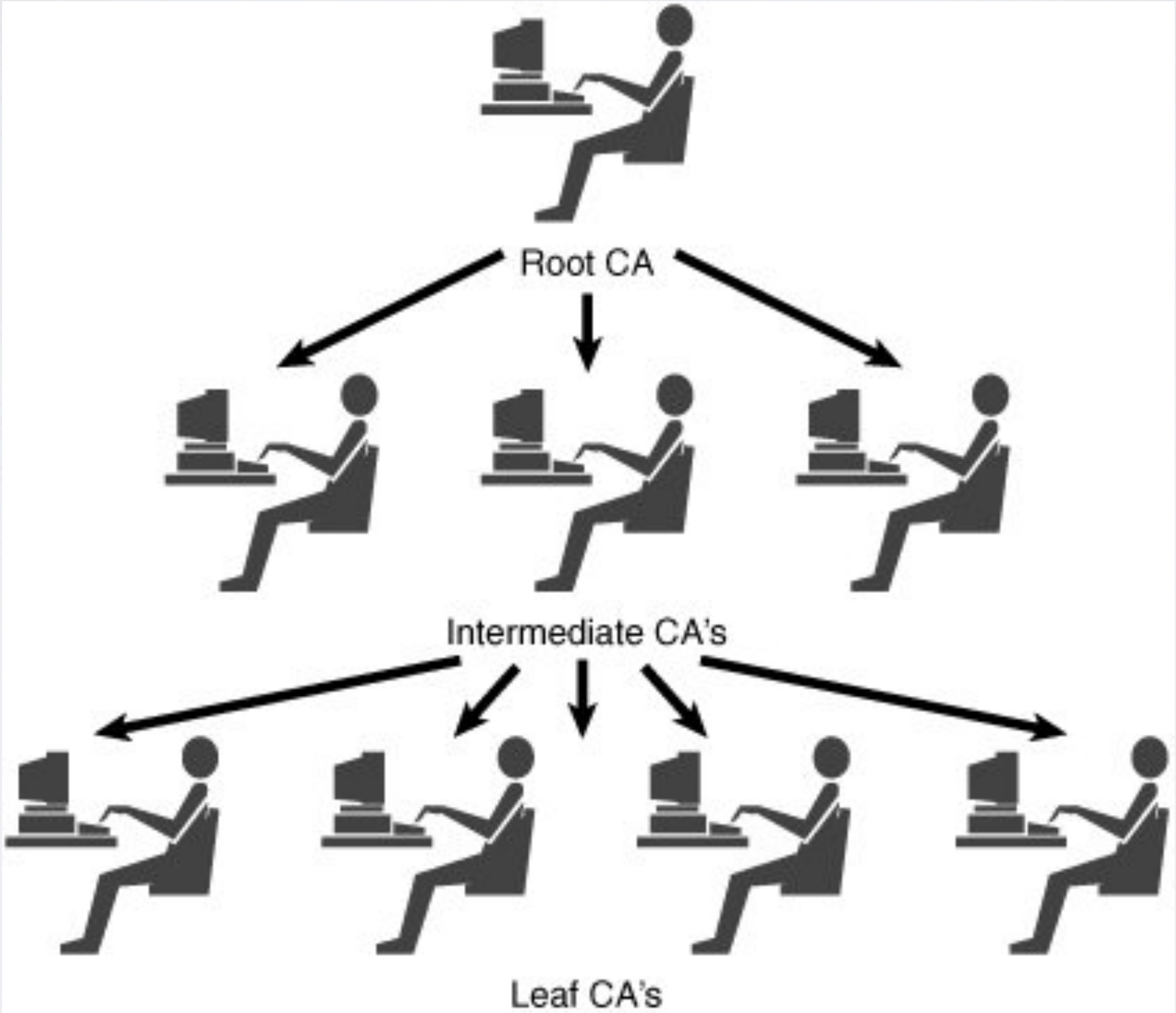
GNUPG

- Stands for GNU Privacy Guard http://www.gnupg.org/
- Is a Free (open-source) implementation of the OpenPGP standard.
- The package is separate from any GUI and refers to the Library and Binary tools.
  - Linux - comes with all distributions
  - Windows - http://www.gpg4win.org/ (Do NOT use the outlook plugin)
  - OS X - https://gpgtools.org/

PGP WEB OF TRUST



CERTIFICATE AUTHORITY WEB OF TRUST



WHAT PGP/GPG DOES PROVIDE

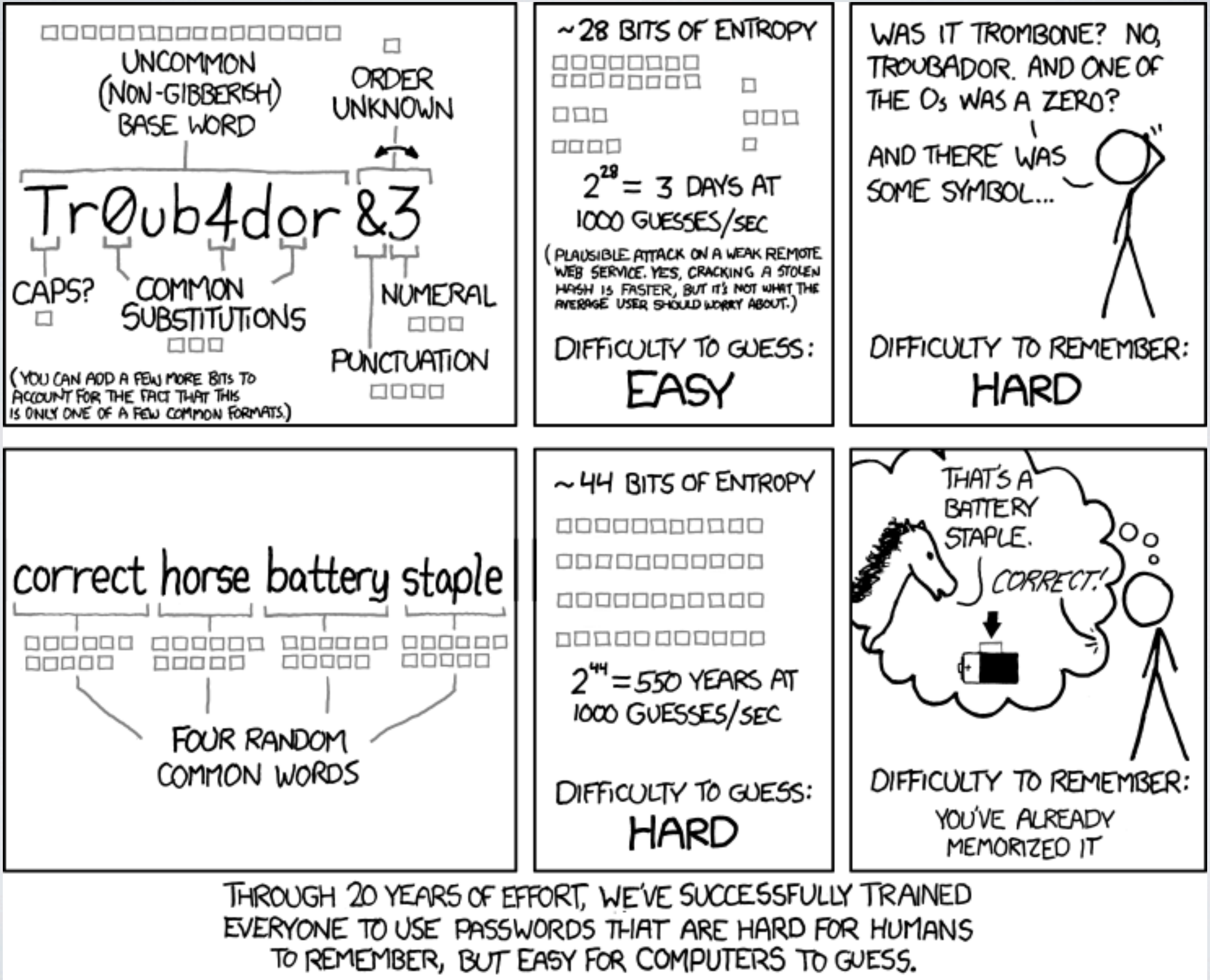
- Verification of sender.
- Encryption of data being sent.
- Trust relationship based on reputation of known persons.
- Strong protection of offline data or data at rest at other location as long as private key is protected.

WHAT PGP/GPG DOES NOT PROVIDE

- Anonymity
- Enumeration of Metadata (Subject, Source, Destination, Possible software version)
- Enumeration of Relations (People that trust the parties)

GENERATING KEYS

- The command to generate the keys is: **gpg --gen-key**
- Choose key sizes larger than 1024.
- Set an expiration date for the key.
- Set a good passphrase to protect the key.
- To list the key **gpg --list-keys "<your name|Email>"**


UNCOMMON (NON-GIBBERISH) BASE WORD
ORDER UNKNOWN
Tr0ub4dor&3
CAPS?
COMMON SUBSTITUTIONS
NUMERAL
PUNCTUATION
(YOU CAN ADD A FEW MORE BITS TO ACCOUNT FOR THE FACT THAT THIS IS ONLY ONE OF A FEW COMMON FORMATS.)
~28 BITS OF ENTROPY
2^28 = 3 DAYS AT 1000 GUESSES/SEC
(PLAUSIBLE ATTACK ON A WEAK REMOTE WEB SERVICE. YES, CRACKING A STOLEN HASH IS FASTER, BUT IT'S NOT WHAT THE AVERAGE USER SHOULD WORRY ABOUT.)
DIFFICULTY TO GUESS: EASY
WAS IT TROMBONE? NO, TROUBADOR. AND ONE OF THE Os WAS A ZERO?
AND THERE WAS SOME SYMBOL...
DIFFICULTY TO REMEMBER: HARD
correct horse battery staple
FOUR RANDOM COMMON WORDS
~44 BITS OF ENTROPY
2^44 = 550 YEARS AT 1000 GUESSES/SEC
DIFFICULTY TO GUESS: HARD
THAT'S A BATTERY STAPLE.
CORRECT!
DIFFICULTY TO REMEMBER: YOU'VE ALREADY MEMORIZED IT
THROUGH 20 YEARS OF EFFORT, WE'VE SUCCESSFULLY TRAINED EVERYONE TO USE PASSWORDS THAT ARE HARD FOR HUMANS TO REMEMBER, BUT EASY FOR COMPUTERS TO GUESS.

GENERATING KEYS

- After generating a key pair create a revocation certificate and save it in a safe place with **gpg --output revoke.asc --gen-revoke <keyid>**

- Revocation certificate is use to revoke your key from key servers in the case you lost your passphrase.

- A revoked key can still be used to verify old signatures, or decrypt data, but it cannot be used to encrypt new messages to you.

GENERATING KEYS

- To list secret keys **gpg --list-secret-keys**
- Create a backup of your private key **gpg --export-secret-key -a "[name|email]" > private.key**
- placed the backed up public and private keys in a safe place.
- To restore a private key on another machine:
  - **gpg --import public.key**
  - **gpg --allow-secret-key-import --import private.key**

UPLOAD YOUR KEY TO A KEYSERVER

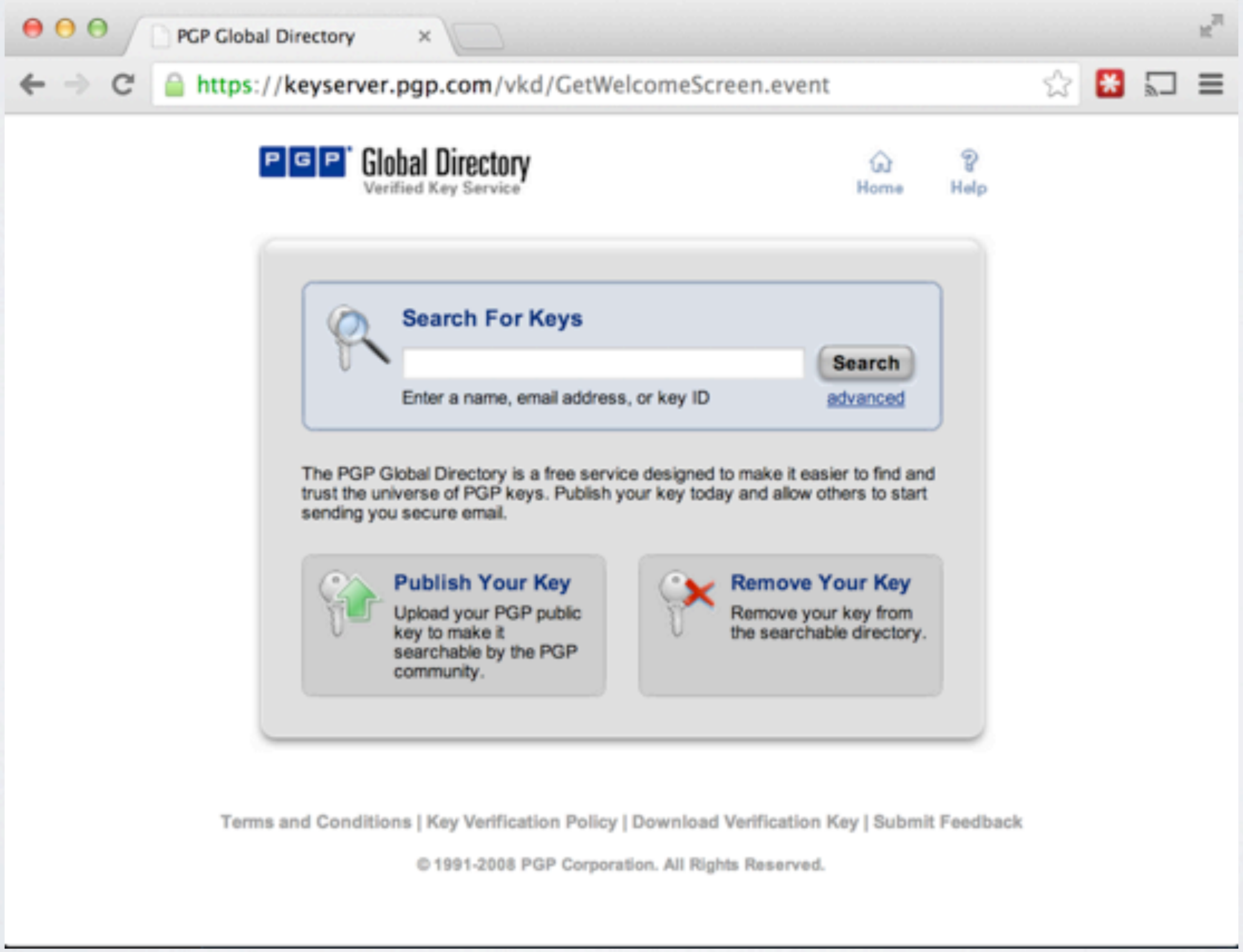
- For first time keys use a key server that verifies the email, this applies to you and anyone you ask to generate a new key to communicate with.

- To export a key to a server **gpg --keyserver <keyserver> --send-keys <key ID>**

- To export an individual Public key for sharing **gpg --armor --export [email|name] > pubkey.asc**

UPLOAD YOUR KEY TO A KEY SERVER

- A recommended server is https://keyserver.pgp.com server will validate the key via the email message in the key and will ask for periodic confirmation.



PGP Global Directory
https://keyserver.pgp.com/vkd/GetWelcomeScreen.event
PGP Global Directory
Verified Key Service
Home
Help
Search For Keys
Search
Enter a name, email address, or key ID
advanced
Publish Your Key
Remove Your Key
Terms and Conditions | Key Verification Policy | Download Verification Key | Submit Feedback

The PGP Global Directory is a free service designed to make it easier to find and trust the universe of PGP keys. Publish your key today and allow others to start sending you secure email.

Upload your PGP public key to make it searchable by the PGP community.

Remove your key from the searchable directory.

© 1991-2008 PGP Corporation. All Rights Reserved.

IMPORTING AND VERIFYING A KEY

- To download a key from a key server **gpg --keyserver <keyserver> --recv-keys <key id>**
- To import an exported key **gpg --import <key file>**
- After we import a key the fingerprint should verified to know if its the one we expected **gpg --fingerprint "[email|name]"**

IMPORTING AND VERIFYING A KEY

- To download a key from a key server **gpg --keyserver <keyserver> --recv-keys <key id>**
- To import an exported key **gpg --import <key file>**
- After we import a key the fingerprint should verified to know if its the one we expected **gpg --fingerprint "[email|name]"**

IMPORTING AND VERIFYING A KEY

- Once a key is verified you can sign it with our key, for this we have to edit the key
  - **gpg --edit-key "<u>[email|name]</u>"**
  - **gpg> sign**

REMOVING A KEY FROM THE KEYRING

- To remove a key a trusted source from the keyring trustdb.gpg **gpg --delete-key "[name|email]"**

- To remove a secret key from secring **gpg --delete-secret-key "[name|email]"**

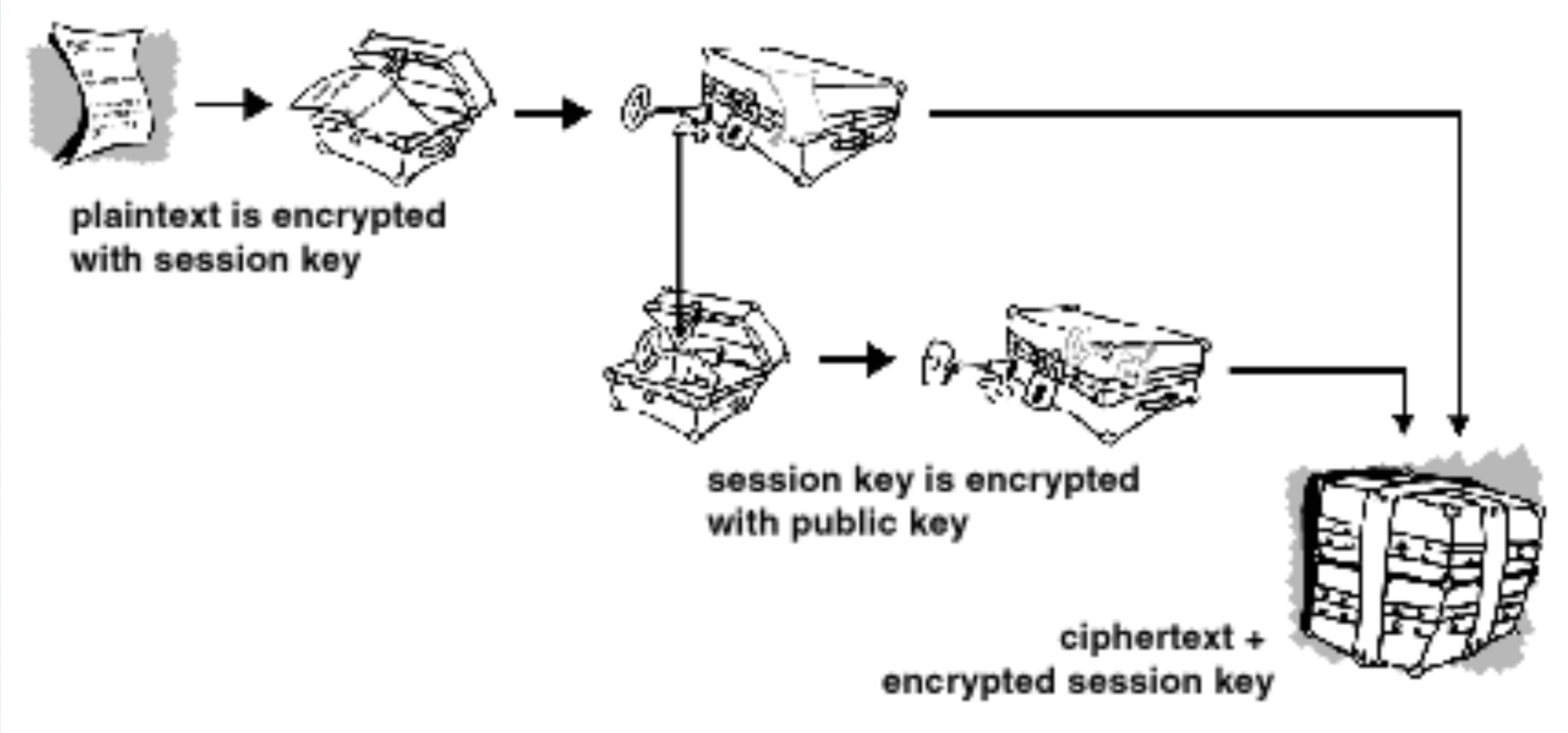
ENCRYPTING A FILE

- Encrypt symmetrically a file using a password **gpg -c filename**

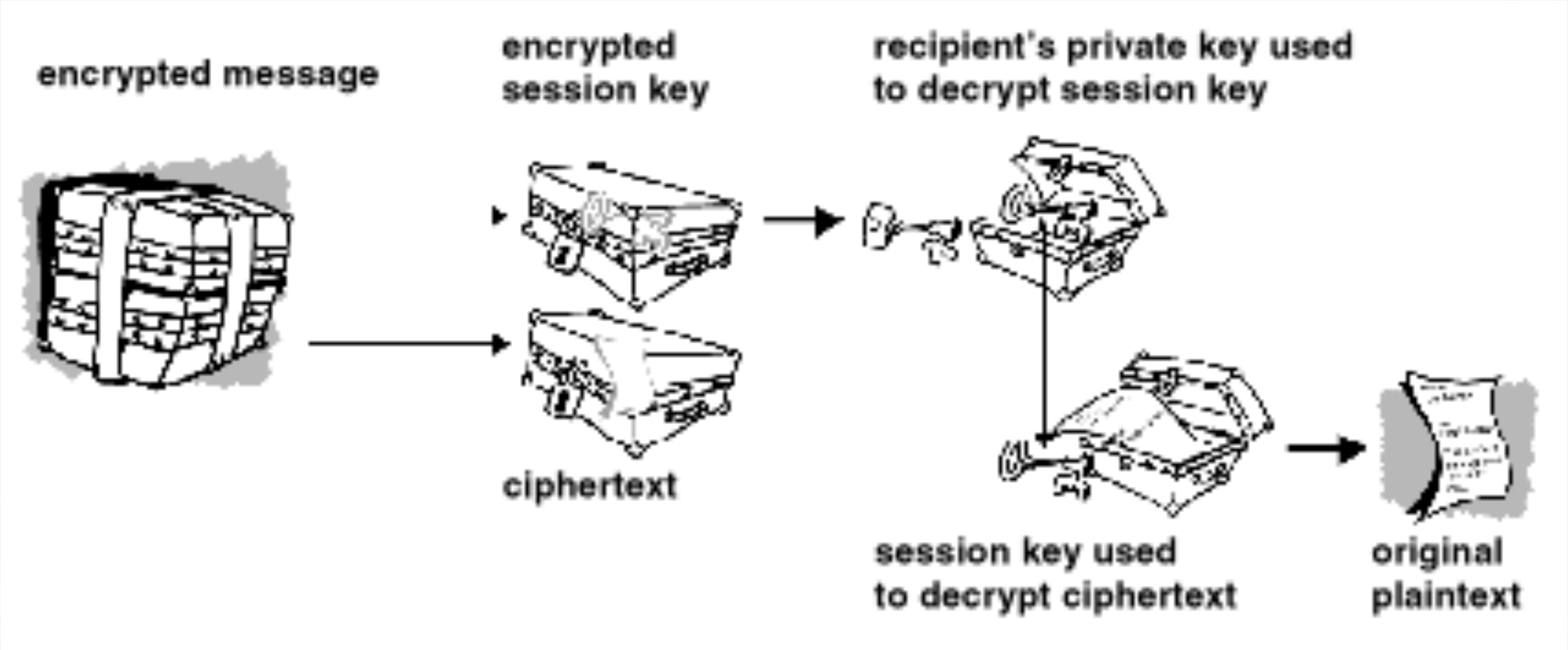
- Decrypt a file using a **gpg -d --output <new filename> filename**

- To encrypt a file with a specific public key **gpg --output document.gpg --encrypt --recipient “[email|name]” document.doc**

ENCRYPTING A FILE


plaintext is encrypted with session key
session key is encrypted with public key
ciphertext + encrypted session key

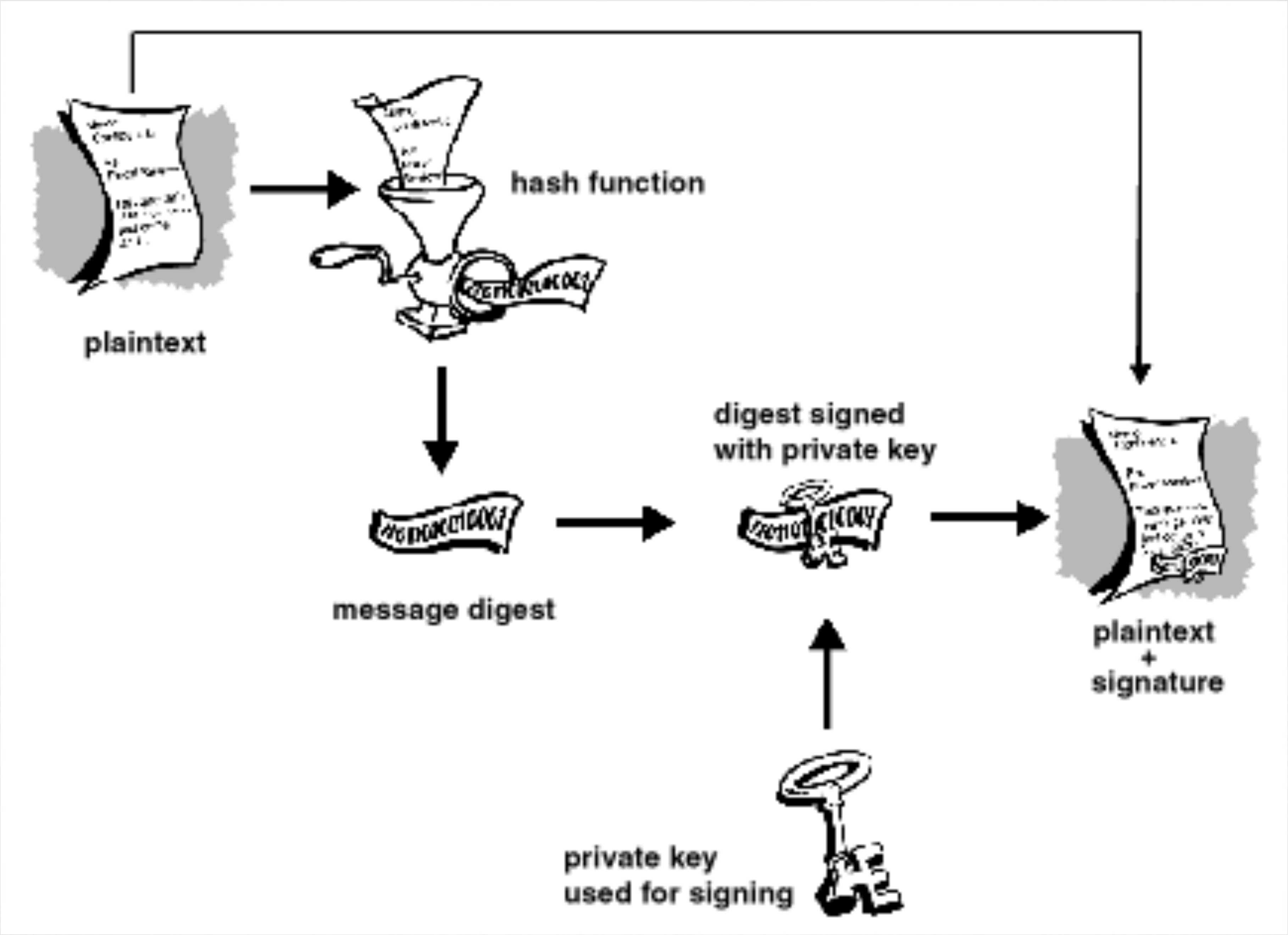
DECRYPTING A FILE


encrypted message
encrypted session key
recipient's private key used to decrypt session key
ciphertext
session key used to decrypt ciphertext
original plaintext

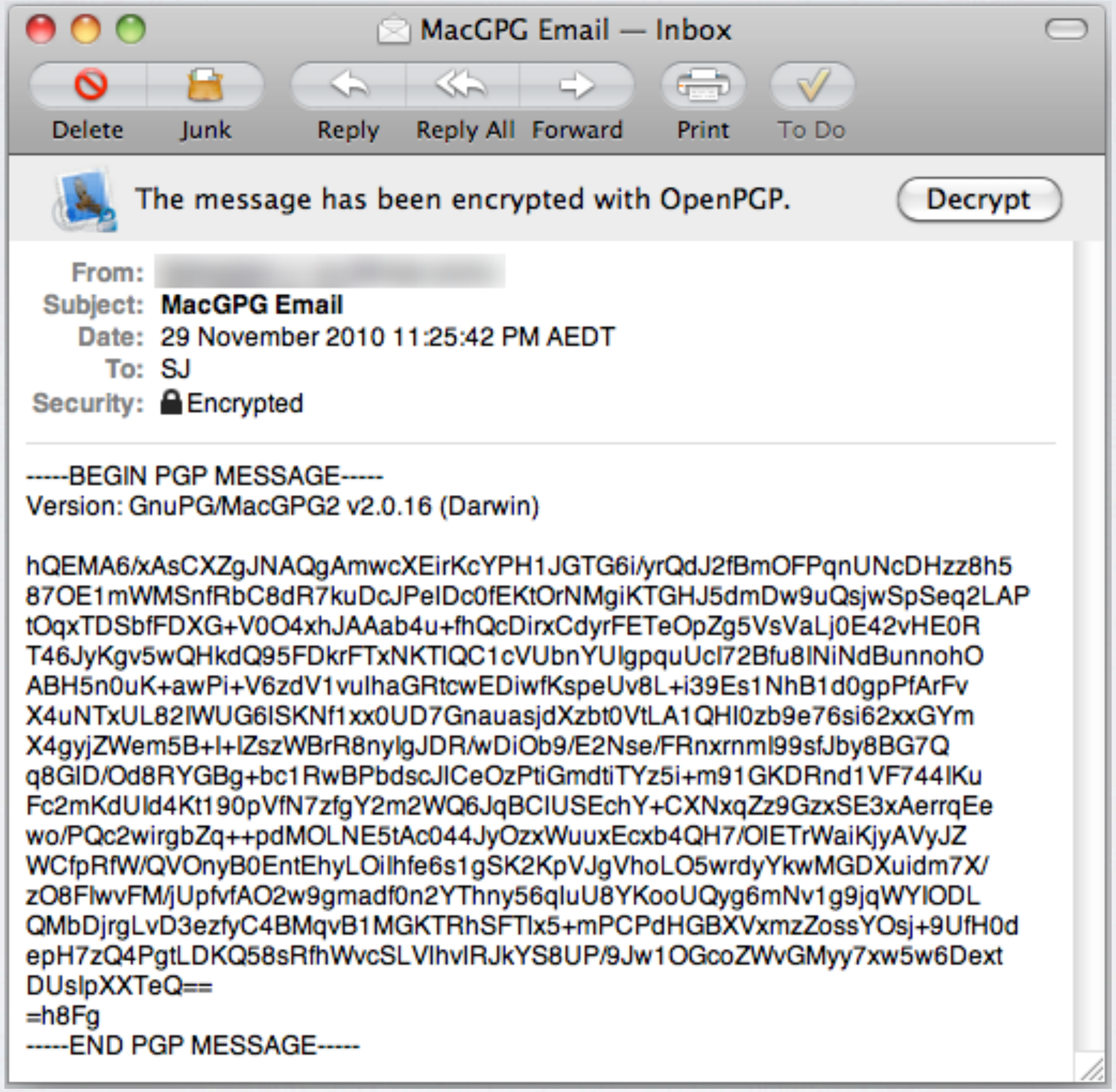
SIGNING AND VERIFYING A FILE

- To generate a signature for a file **gpg --output file.sig --sign file**
- To verify a signature both the sig file and the original file must be in the same folder **gpg --verify file.sig**

SIGNING AND VERIFYING A FILE



LEAKING TO MUCH INFORMATION



The message has been encrypted with OpenPGP.

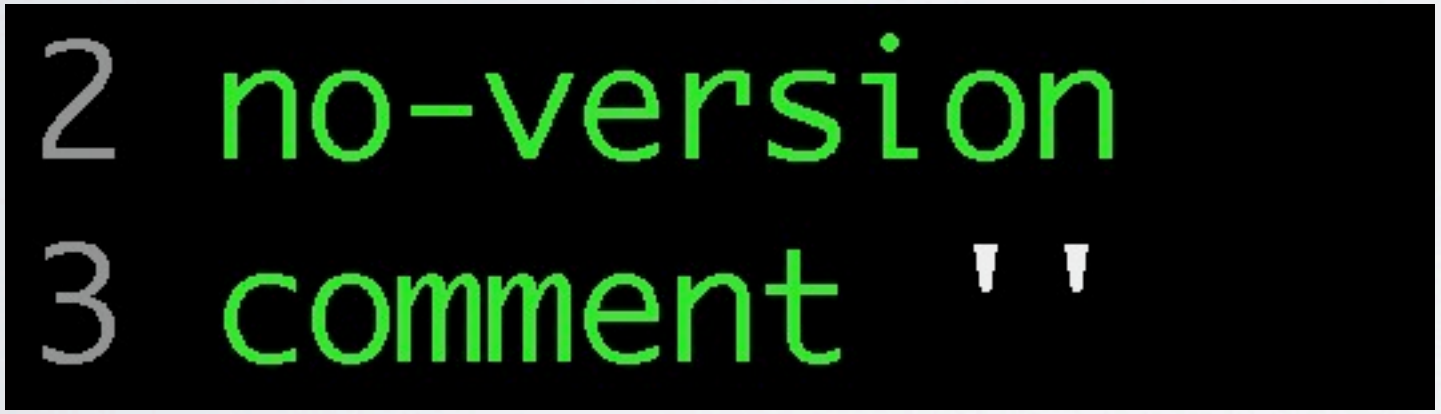
From: [redacted]
Subject: **MacGPG Email**
Date: 29 November 2010 11:25:42 PM AEDT
To: SJ
Security: 🔒Encrypted

```
-----BEGIN PGP MESSAGE-----
Version: GnuPG/MacGPG2 v2.0.16 (Darwin)

hQEMA6/xAsCXZgJNAQgAmwcXEirKcYPH1JGTG6i/yrQdJ2fBmOFPqnUNcDHzz8h5
87OE1mWMSnfRbC8dR7kuDcJPelDc0fEKtOrNMgiKTGHJ5dmDw9uQsjwSpSeq2LAP
tOqxTDSbfFDXG+V0O4xhJAAab4u+fhQcDirxCdyrFETeOpZg5VsVaLj0E42vHE0R
T46JyKgv5wQHkdQ95FDkrFTxNKTlQC1cVUbnYUlgpquUcl72Bfu8INiNdBunnohO
ABH5n0uK+awPi+V6zdV1vulhaGRtcwEDiwfKspeUv8L+i39Es1NhB1d0gpPfArFv
X4uNTxUL82lWUG6lSKNf1xx0UD7GnauasjdXzbt0VtLA1QHl0zb9e76si62xxGYm
X4gyjZWem5B+l+lZszWBrR8nylgJDR/wDiOb9/E2Nse/FRnxrnml99sfJby8BG7Q
q8GlD/Od8RYGBg+bc1RwBPbdscJlCeOzPtiGmdtiTYz5i+m91GKDRnd1VF744lKu
Fc2mKdUld4Kt190pVfN7zfgY2m2WQ6JqBClUSEchY+CXNxqZz9GzxSE3xAerrqEe
wo/PQc2wirgbZq++pdMOLNE5tAc044JyOzxWuuxEcxb4QH7/OlETrWaiKjyAVyJZ
WCfpRfW/QVOnyB0EntEhyLOilhfe6s1gSK2KpVJgVhoLO5wrdyYkwMGDXuidm7X/
zO8FlwvFM/jUpfvfAO2w9gmadf0n2YThny56qluU8YKooUQyg6mNv1g9jqWYlODL
QMbDjrgLvD3ezfyC4BMqvB1MGKTRhSFTlx5+mPCPdHGBXVxmzZossYOsj+9UfH0d
epH7zQ4PgtLDKQ58sRfhWvcSLVlhvlRJkYS8UP/9Jw1OGcoZWvGMyy7xw5w6Dext
DUslpXXTeQ==
=h8Fg
-----END PGP MESSAGE-----
```

DISABLE COMMENT AND VERSION INFO

- Add to your gpg.conf file the following lines:



```
no-version
comment ''
```

- Disables version information

- Sets the comment to an empty string

THANKS