GNU Privacy Guard

Protect your mail from eavesdroppers

and worse

Why email?

- Email is interoperable (Gmail users can talk to Yahoo users, etc.)
- Email has wide adoption
- Email does not depend on one company or government
- Email is extensible

A brief history of email

- **1960s:** methods exist for message passing between users on the same system (e.g., MIT's CTSS)
- **1971:** Ray Tomlinson creates the first mail transfer agent and sends the first email message to a user on another system with the user@host notation
- **1979:** Eric Allman creates delivermail, allowing mail to be routed between different networks such as ARPANET and BerkNet
- **1980s:** email begins to be adopted by the consumer market through walled-garden (i.e., non-Internet) networks such as Prodigy
- **1990s:** most email is now transferred over the Internet

Email vulnerabilities: sender authentication

From: service@paypal.com
Subject: **Security verification - Please update your records**
Date: November 29, 2005 7:41:57 AM PST
To: Vaughn Aubuchon

PayPal *The way to send and receive money online*

Email vulnerabilities: interception/seizure

Any non-end-to-end encryption is subject to subversion at the provider level:

180

Email stored on server for 180 days subject to warrantless (subpoena-based) US government requests per 1986 Stored Communications Act

Lavabit

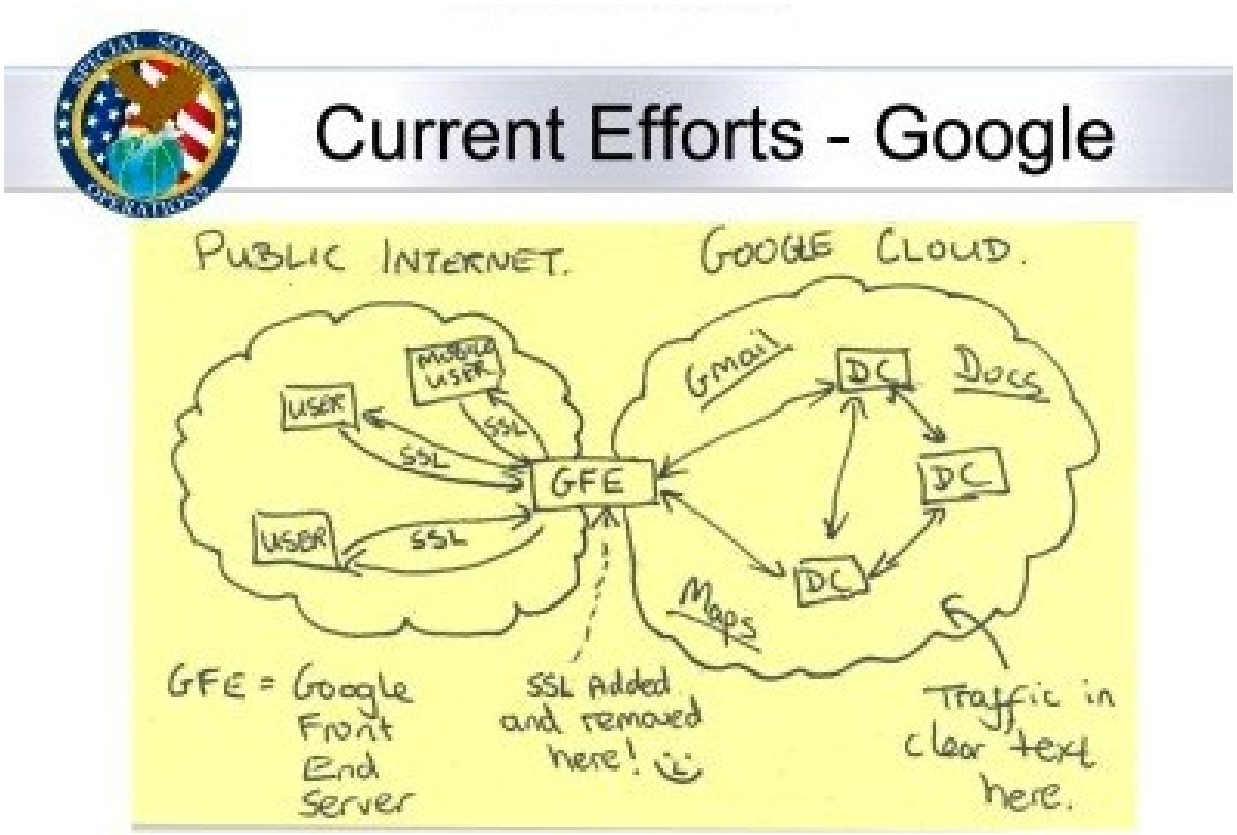
2013: FBI receives warrant requiring Levison to turn over SSL keys protecting Lavabit's 300,000 clients

Google reports 32,000 government requests for confidential data in the first half of 2014; complies with 65%

Provider-based security

- Most providers use STARTTLS and HTTPS-enabled webmail to provide *transport encryption*. This protects traffic to and from the mail server.
- Your provider has full access to your mail
- This exposes you to privacy risks and your provider to legal liability

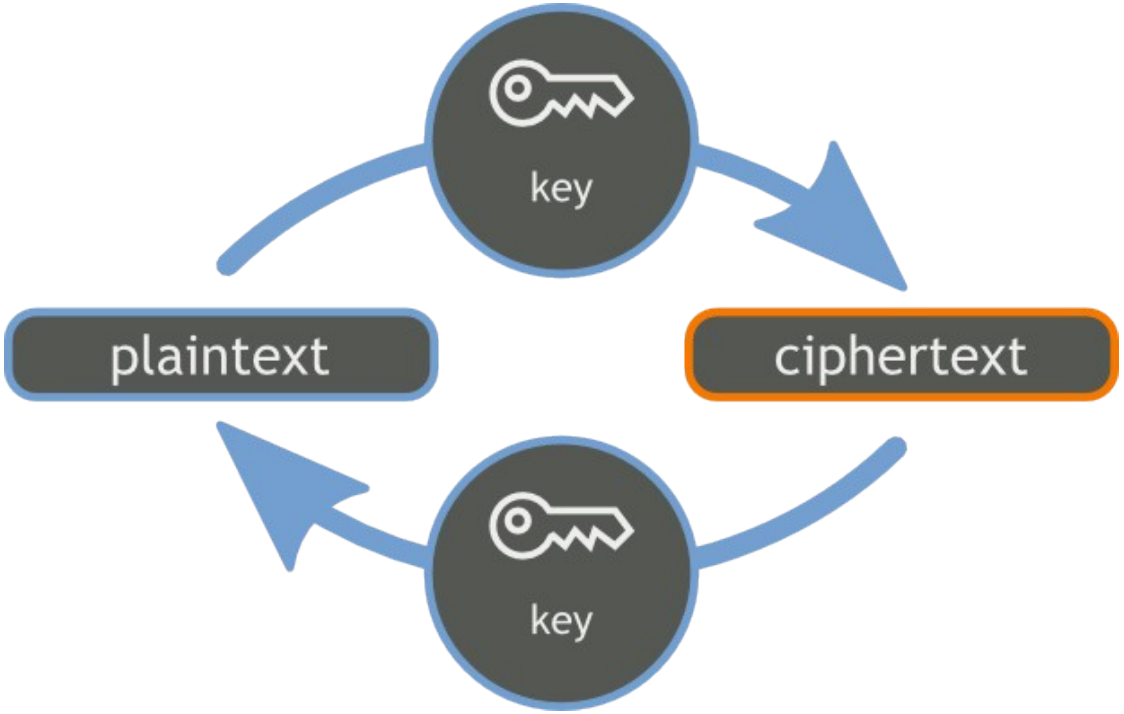
Why TLS is insufficient



Transport Layer Security defeated.
Use End-to-End encryption!

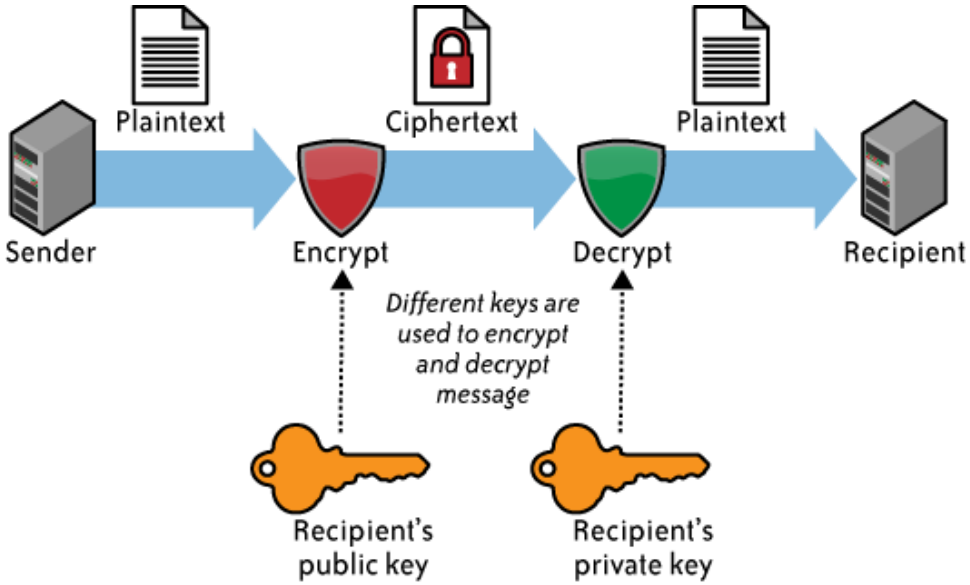
Symmetric cryptography

- A shared secret (e.g., password) allows "symmetric" ciphers (AES, IDEA, DES)
- Only way to provide "perfect" security
- Sharing a secret requires a secure channel



Asymmetric cryptography

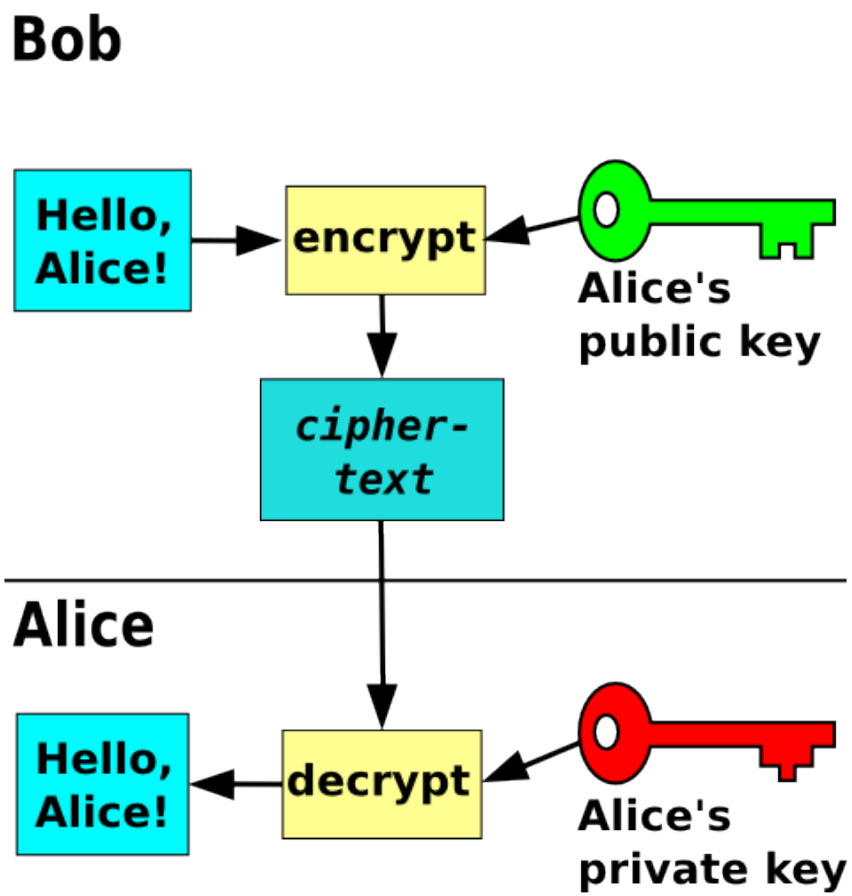
- Asymmetric cryptography uses a keypair consisting of "private" and "public" keys
- Each key decrypts messages encrypted by the other
- Computationally expensive: requires very large keys
- Messages are signed by "encrypting" with the private key



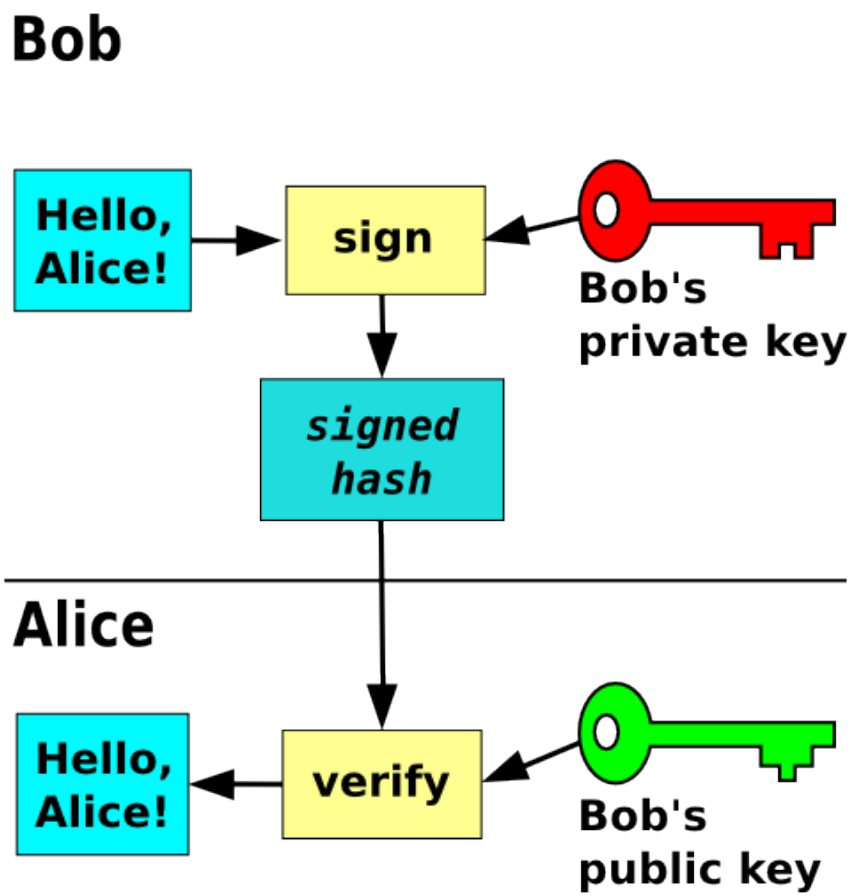
Hybrid cryptography: PGP

- Public-key encryption establishes shared secret
- Message is symmetrically encrypted with that secret
- Message hashes are signed rather than the messages themselves
- Minimizes computational cost while retaining convenience of public-key cryptography

Encrypting a message



Signing a message



Mathematics of RSA

- Security based on the difficulty of factorization
- Public key is two numbers: an exponent and a modulus (e, n)
- Private key is one number (d)
- A plaintext chunk *p* is a number to be converted into a corresponding ciphertext *c*

$$c = p^e \mod n$$
$$p = c^d \mod n$$

Mathematics of RSA: simplified example

$$c = p^e \mod n$$
$$p = c^d \mod n$$

$$1394 = 89^3 \mod 3127$$
$$89 = 1394^{2011} \mod 3127$$

RSA: risks and pitfalls

- Failure to verify key ownership
- Key compromise
- Endpoint compromise
- Metadata exposure to keyservers

Installing GnuPG

- Linux users, check your repo for **gpg2**
- Mac users should use **GPGTools**
  - https://gpgtools.org
- Windows users should use **Gpg4win**
  - http://www.gpg4win.org

Don't forget those checksums!

- ae694b45a91b1091625beefbd230dad953b31376 gpg4win-2.2.2.exe (SHA1)
- ac7a636bfee1027d8f43a12a82eea54e7566dcb8 GPG Suite - 2013.10.22.dmg (SHA1)

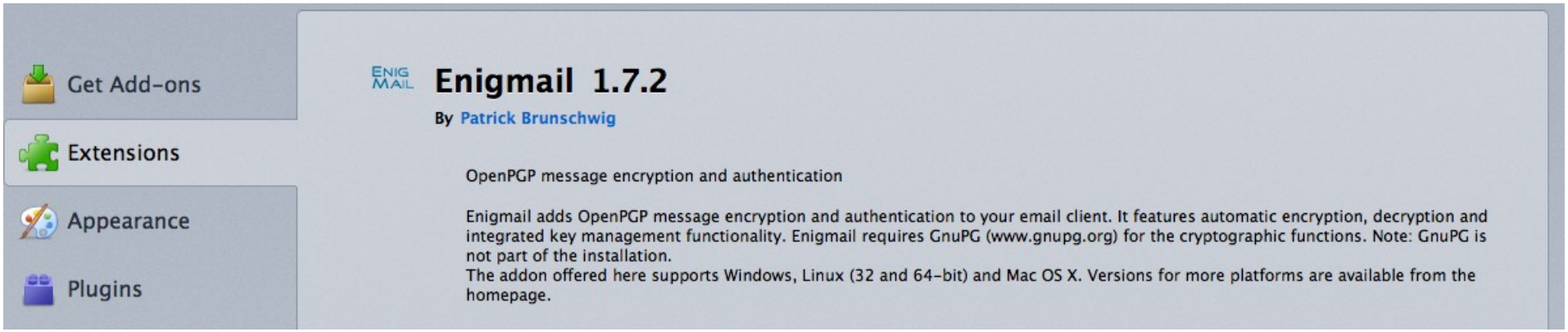
Key generation

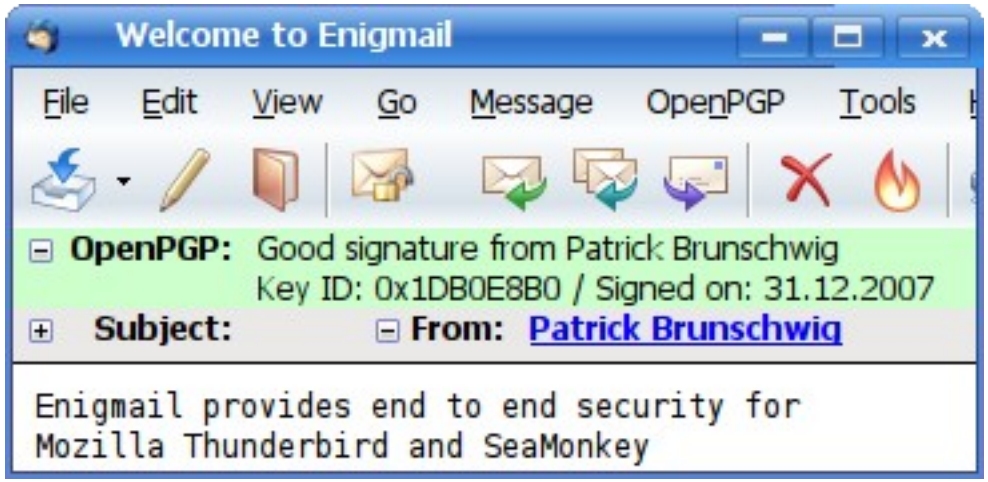
`gpg --gen-key`

- 2048 or 4096 bits
- Expiration date
- Name
- Email
- Comment
- Passphrase

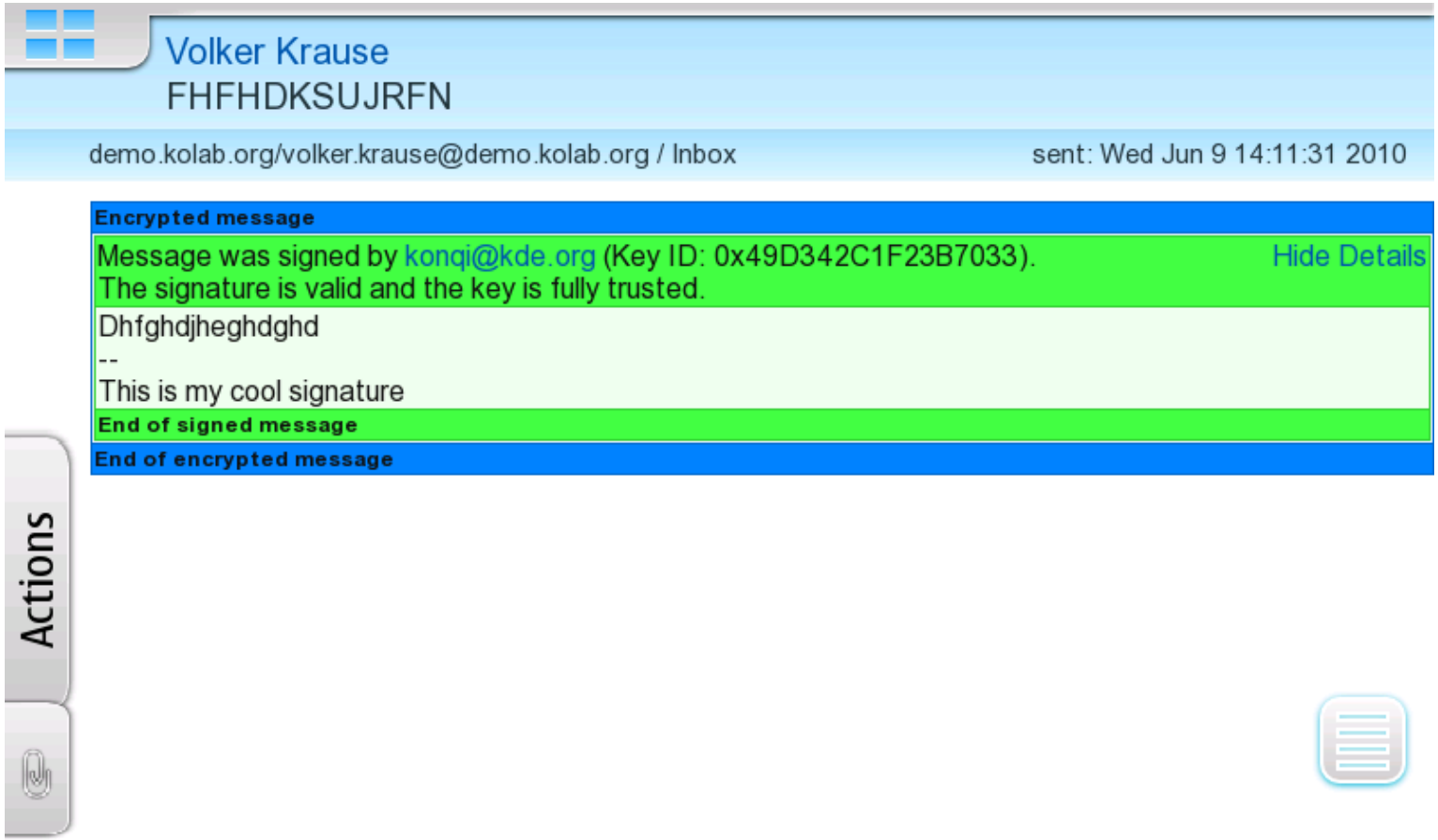
Thunderbird supports GPG

(with a little help)





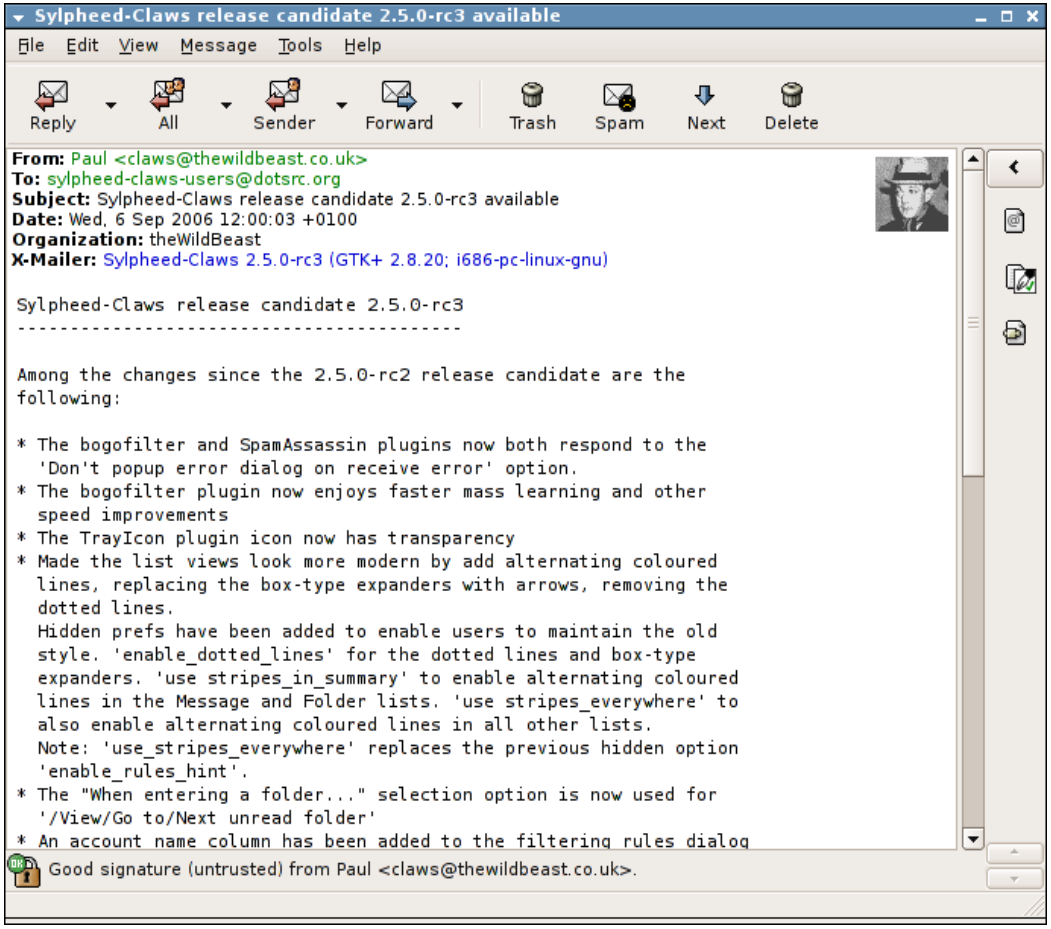
KMail supports GPG



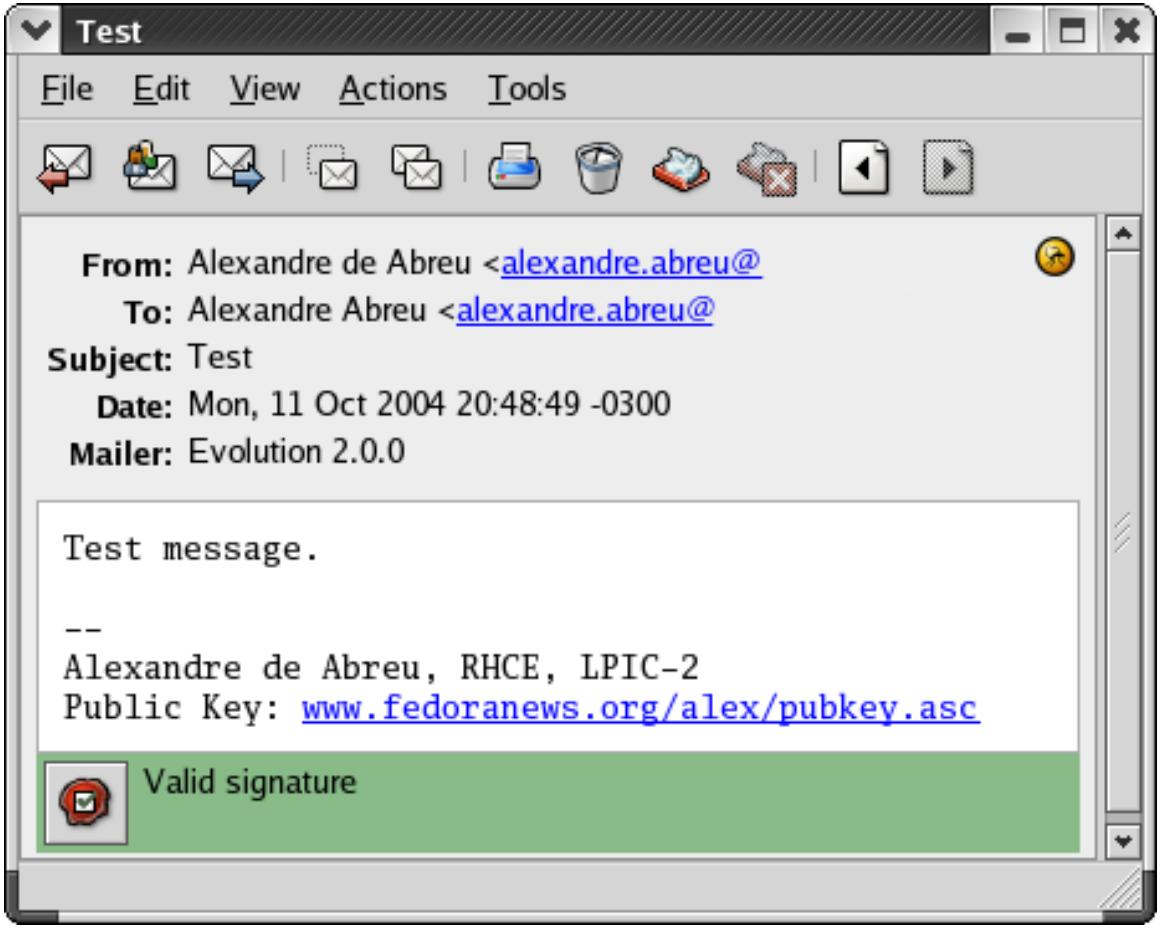
Mutt supports GPG

```
i:Exit  ^B:PrevPg  ^F:NextPg v:View Attachm.  d:Del  r:Reply  j:Next ?:Help
Date: Tue, 23 Jul 2013 19:42:07 +1200
From: Tom Ryder <tom@sanctum.geek.nz>
To: Joe Somebody <joe@example.com>
Subject: PGP Test Message
User-Agent: Mutt/1.5.21 (2010-09-15)

[-- Begin signature information --]
Good signature from: Thomas Ryder (tyrmored, tejr) <tom@sanctum.geek.nz>
            created: Tue 23 Jul 2013 19:42:07 NZST
[-- End signature information --]

[-- The following data is PGP/MIME signed and encrypted --]

Test message only.

--
Tom Ryder
<http://www.sanctum.geek.nz/>


[-- End of PGP/MIME signed and encrypted data --]
   1  PF Jul 23 at 07:42 PM  Tom Ryder  PGP Test Message                  -- (all)
Invoking PGP...
```

Claws supports GPG



Evolution supports GPG



GPG in webmail

we're not quite there yet!

GPG in webmail

Google End-to-End

- End-to-End can only generate P-256, P-384, and P-521 elliptic curves believed by Bruce Schneier to be insecure.
- If you use End-to-End, you should import your own key.

Off-the-Record Messaging

- OTR protocol allows:
  - Encryption
  - Authentication
  - Deniability
  - Perfect forward secrecy
- Recommended clients include
  - Adium (OS X)
  - pidgin-otr (Linux/BSD/Windows)