Privacy & Security

Johan Vromans
<jvromans@squirrel.nl>

The problem

You get an email:

From: Larry Wall <lwall@hotmail.com>
To: jv@squirrel.nl
Subject: Refugee fund

We are raising funds for refugees from the USA. Please send money to ...

Does that sound like a message from Larry?

Often it is not that easy

From: "Klantenservice" klantenservice@postbank.nl
To: willekeurig.slachtoffer@example.com
Subject: emailadres controle

Lieve Postbank Klant,

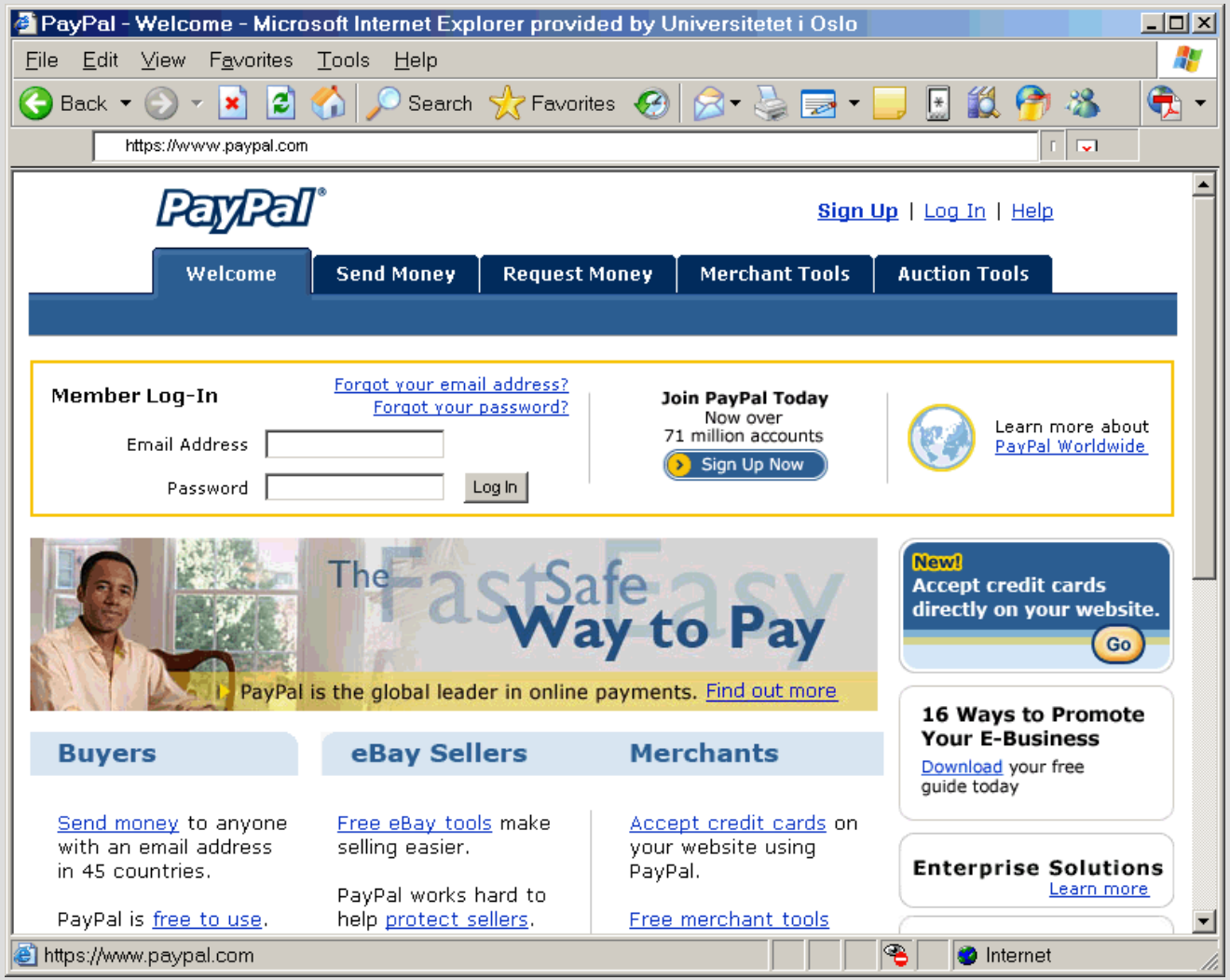
Deze email werd verzonden door de Postbank server om uw identiteit vast te
stellen. Je moet dit proces afmaken door de link beneden aan te klikken om door te
gaan in het volgende menu en daar je gebruikersnaam en wachtwoord invullen.
Dit is gedaan voor je eigen veiligheid - omdat sommige van onze leden niet langer
toegang hebben tot hun emailadres en wij het moeten verifieren.

https://www.p3.postbank.nl/sesam/SesamLoginServlet

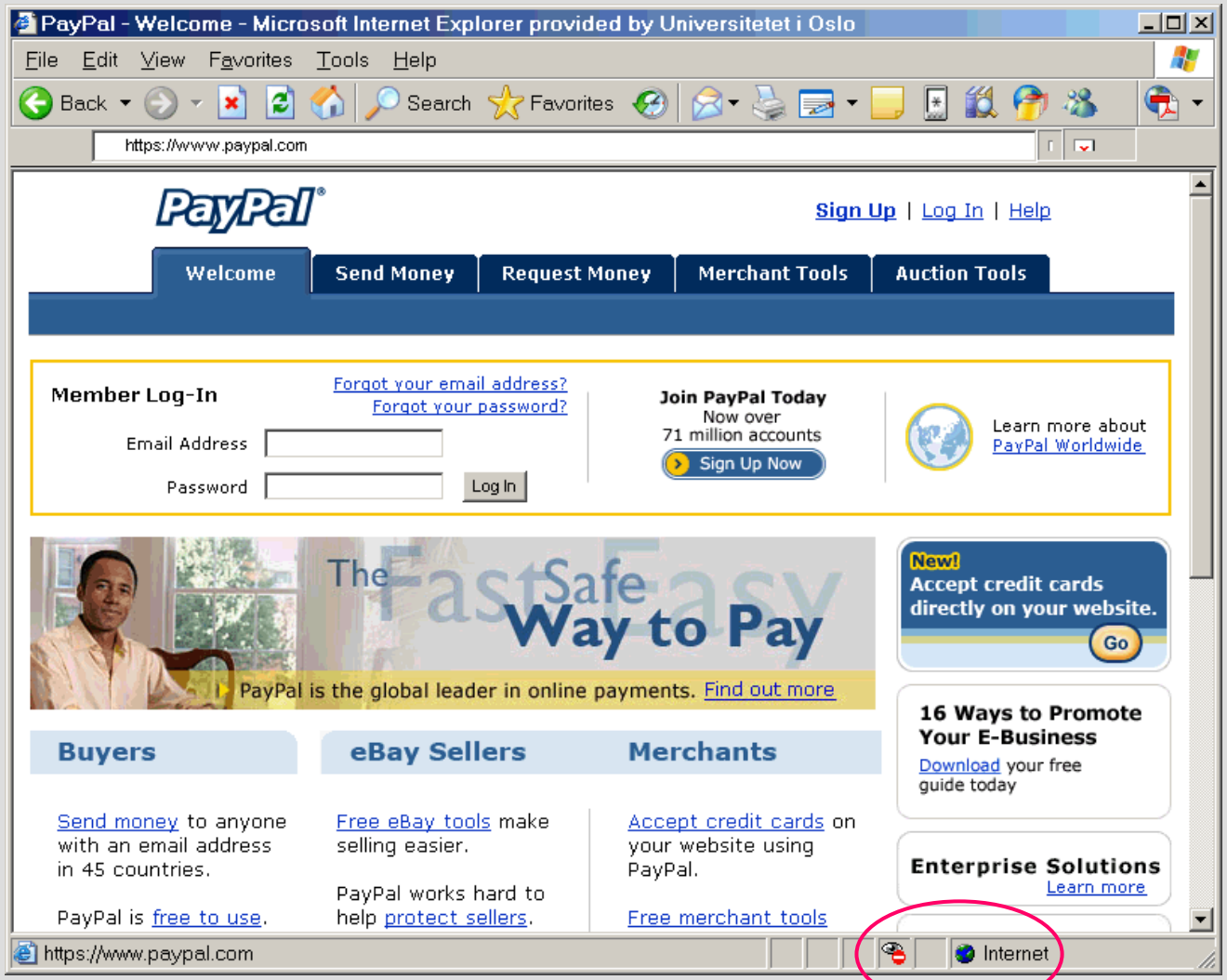
Wij hopen u hiermee voldoende te hebben geïnformeerd.
Met vriendelijke groet,

Klantenservice Postbank

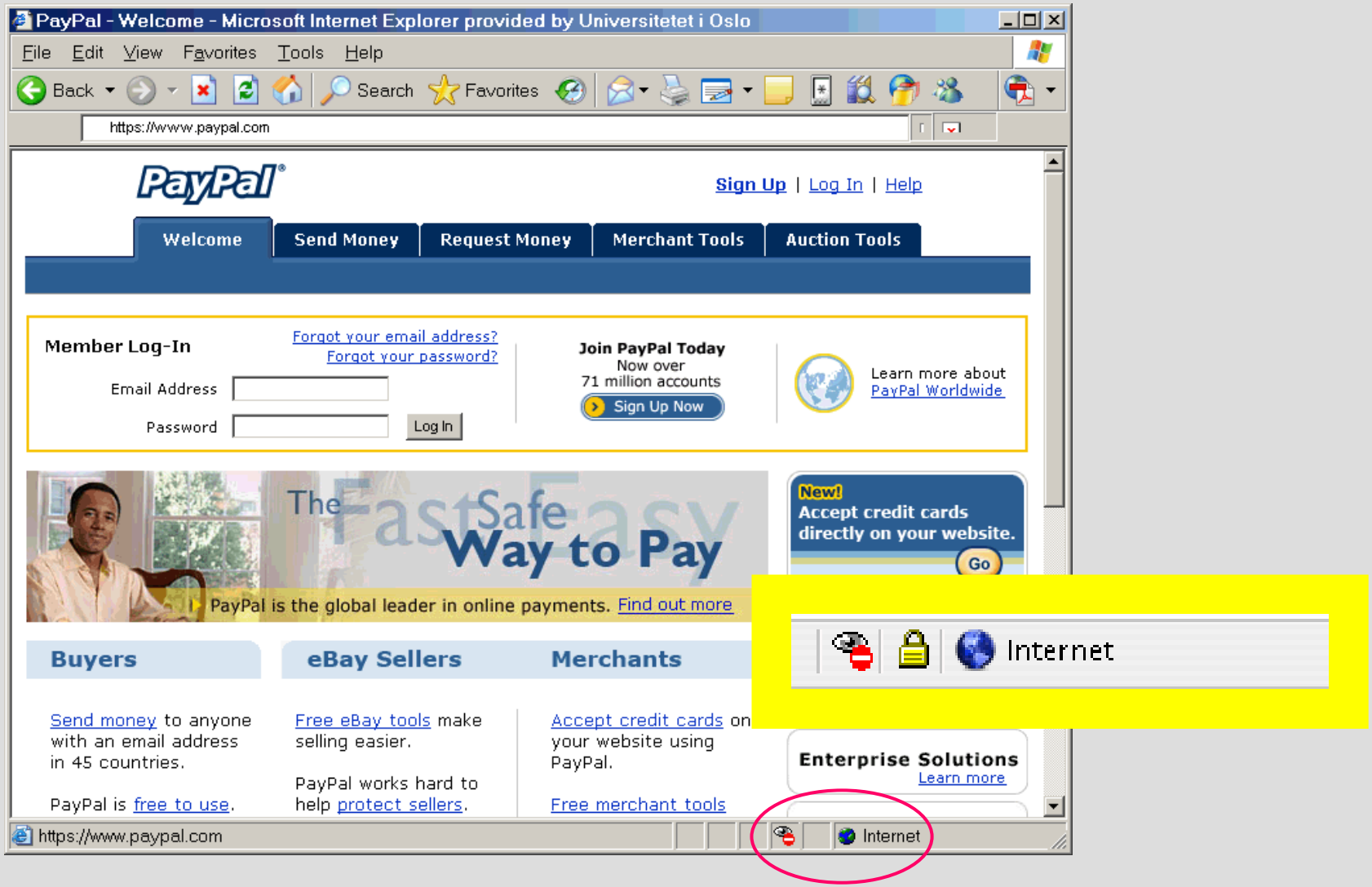
Often it is not that easy



Often it is not that easy



Often it is not that easy



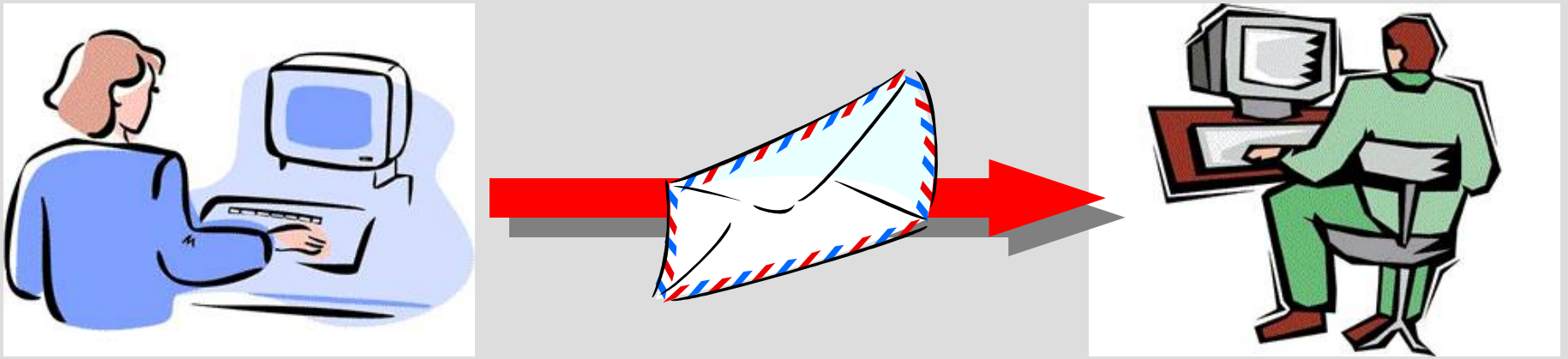
On the internet, everybody is a dog

"On the Internet, nobody knows you're a dog."

Meet Alice and Bob

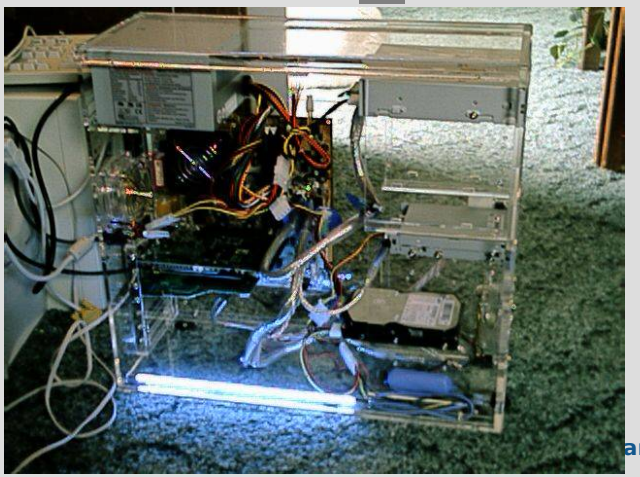
Meet Alice and Bob

Alice sends a message to Bob



Alice sends a message to Bob

Alice sends a message to Bob



Problems

- The message flows through several computer systems.
- It is backed up several times.
- The message is clear text.
- Anyone with access tot he computers or the network can see it.

Worse

- Identical copies can be made without the possibility to detect this.
- The message can be changed without the possibility to detect this.
- In fact, there’s no guarantee that the message even originates from the sender.

Problems to be solved

- **Privacy**: Noone except the recipient may be able to see the contents.
- **Integrity**: The message must arrive unchanged. If is has been modified this must be detected.
- **Authentication**: It must be guaranteed that the message originates from the sender.

Encryption

- Encryption is the process of turning a clear message into something uncomprehensible.
- It’s a reversible process, of course.
- The sender and recipient have to agree on how the encryption is done, and how it should be reversed.
- This agreement is called the encryption ‘key’.
- The process is symmetrical: one key to encrypt and decrypt.

Example: Rot-13

Gbc gra ernfbaf gb ebg-13 lbhe HFRARG zrffntr:

Znxr crbcyr guvax lbh fcrnx Lvqqvfu
Sbvy gur Cebqvtl prafbef
Znxr yvsr qvssvphyg sbe gur Angvbany Frphevgl Ntrapl
Xrrc Qnivq Yrggrezna sebz fgrnyvat lbhe zngrevny
Znxr nyg.pbafcvenpl areibhf
Orpnhfr nowhere orpbzrf abjurer, pyrex orpbzrf clerk, naq green orpbzrf terra
Onssyr arjovrf
Vg'f purncre guna gur Pyvccre Puvc
Nzrevpn Bayvar hfref pna'g qrpbqr lbhe zrffntrf naq svaq bhg jung lbh'er fnlvat
Zbz'f ba gur Vagrearg gbb

Example: Rot-13

Top ten reasons to rot-13 your USENET message:

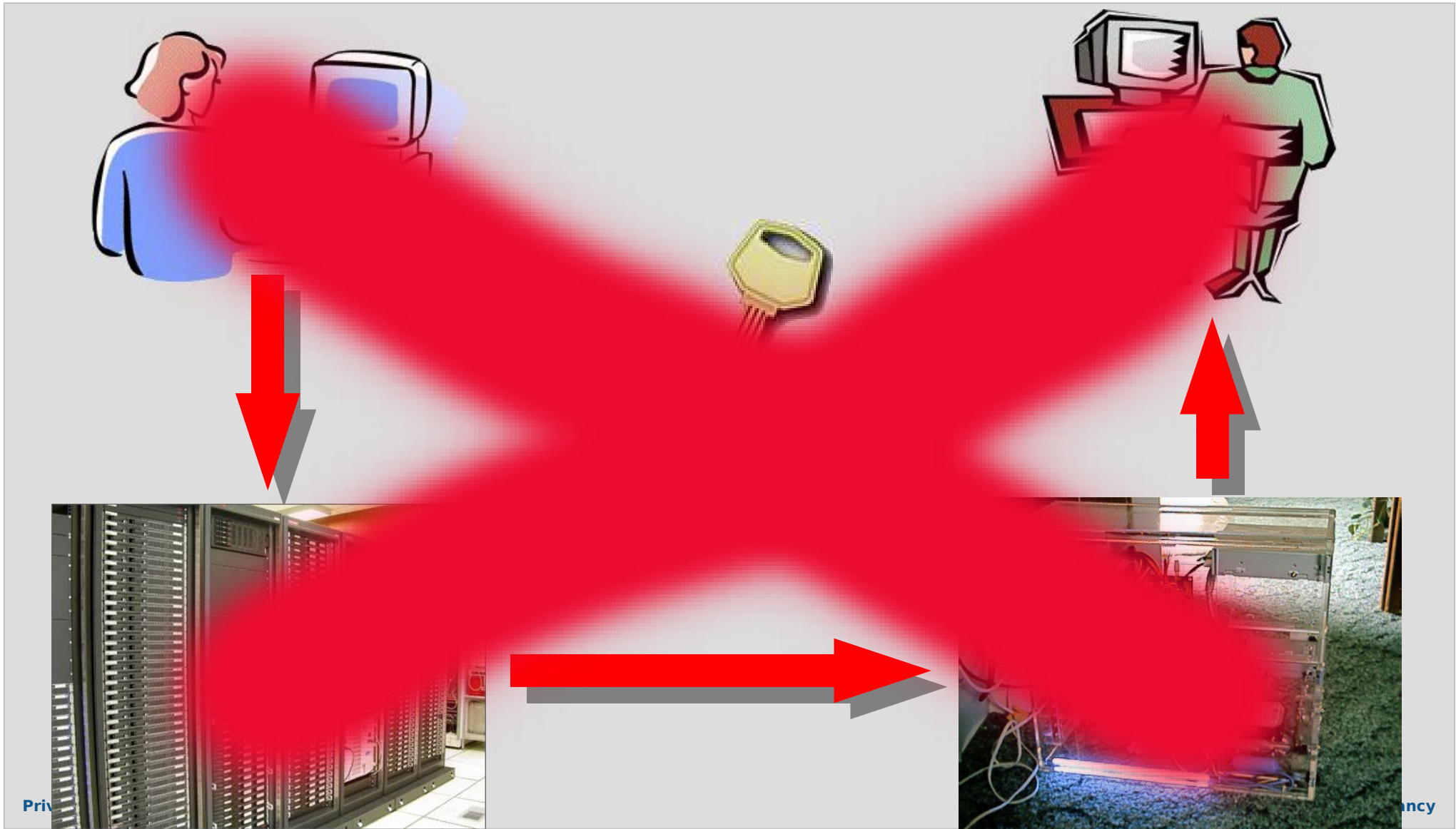
Make people think you speak Yiddish
Foil the Prodigy censors
Make life difficult for the National Security Agency
Keep David Letterman from stealing your material
Make alt.conspiracy nervous
Because abjurer becomes nowhere, clerk becomes pyrex, and terra becomes green
Baffle newbies
It's cheaper than the Clipper Chip
America Online users can't decode your messages and find out what you're saying
Mom's on the Internet too

Problems

- Sender and recipient have to exchange the encryption key.
- Oh, but that’s easy ...

Alice sends her key to Bob

Alice sends her key to Bob

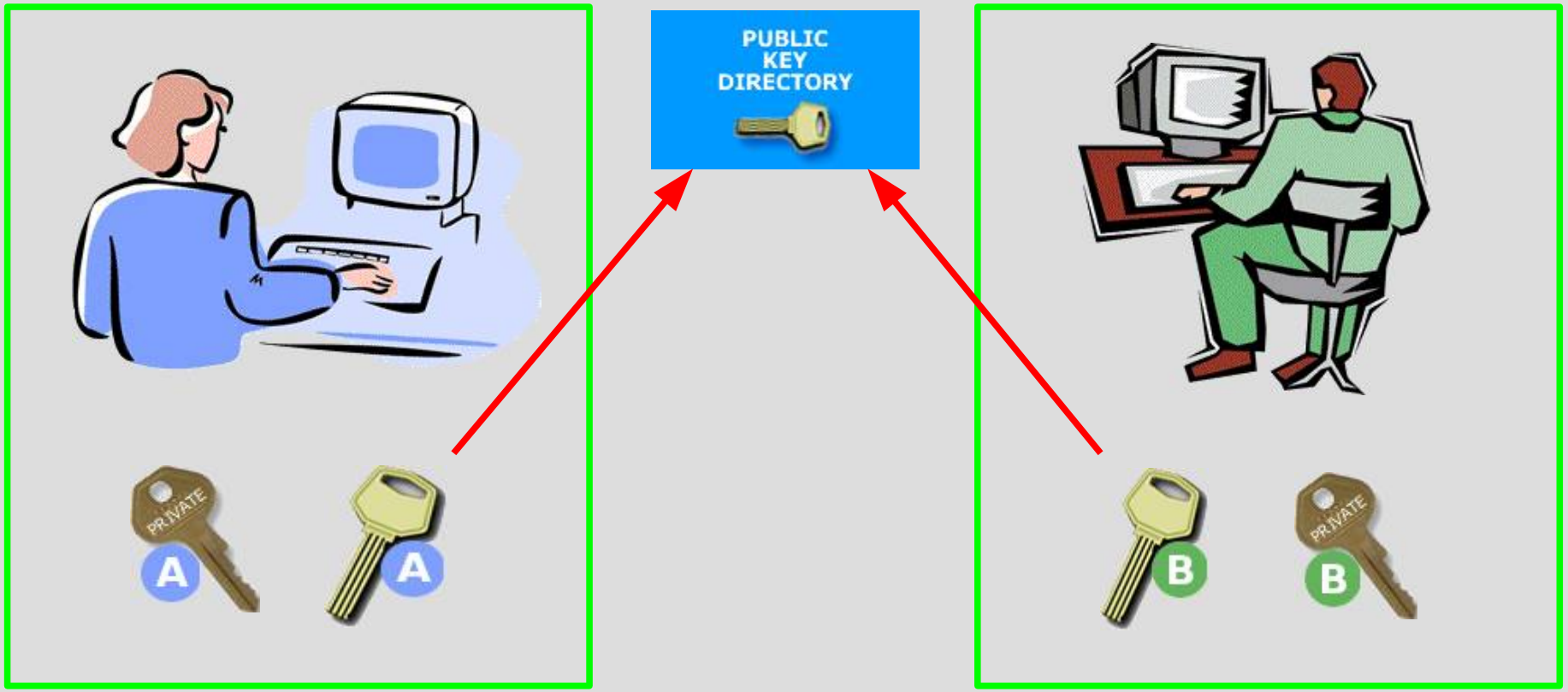


Public Key Encryption

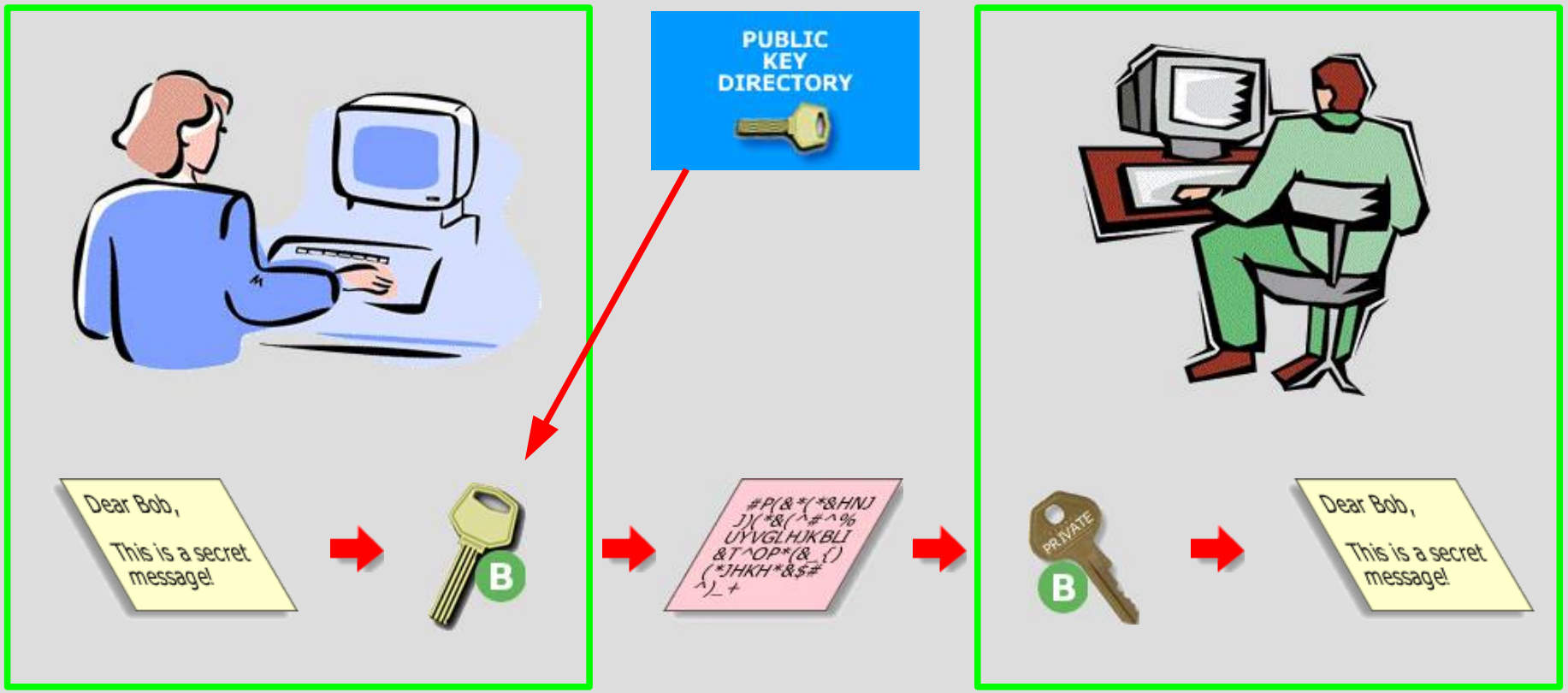
- PKE works with *two* keys.
- One key is *private*, and will never be revealed.
- The other key is *public*, anyone may have it.
- Both keys are mathematically related.
- The public key can decrypt a message that has been encrypted with the private key.
- The private key can decrypt a message that has been encrypted with the public key.

TWO KEYS
1 Public
1 Private

Public Key Encryption



Public Key Encryption



Alice knows that noone but Bob can see the message.

Problems to be solved

Privacy: Noone except the recipient may be able to see the contents.

- **Integrity**: The message must arrive unchanged. If is has been modified this must be detected.

  Nope, not al all ... worse:

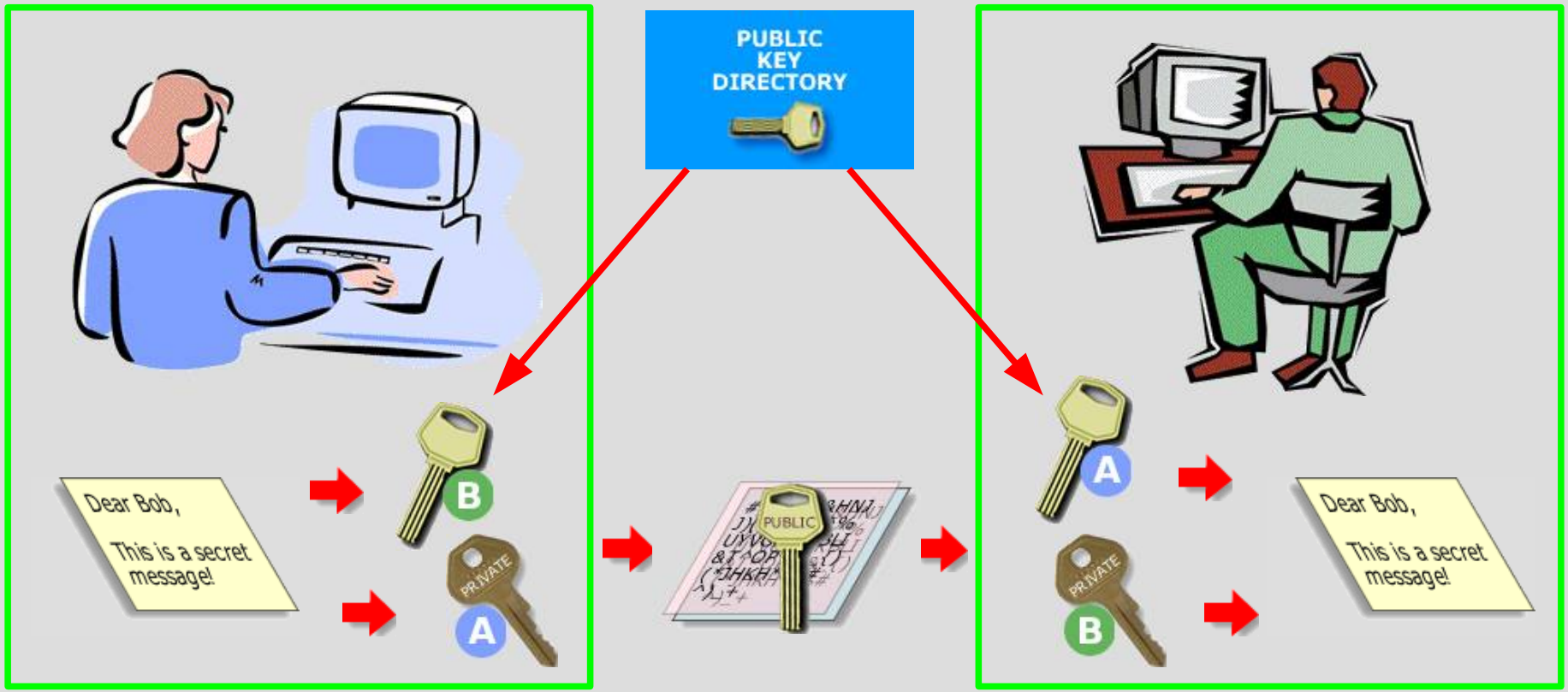
- **Authentication**: It must be guaranteed that the message originates from the sender.

  Anyone can fetch Bob’s public key and send a message, pretending he is Alice.

Digital signatures

- A *digest* or *hash* is a sophisticated checksum derived from the original message.
- The digest is encrypted with the sender's private key and attached to the message. This forms the *digital signature*.
- The recipient can use the sender's public key to decrypt the digest, and verify that it still applies to the message.
- Digital signing is usually combined with encryption (but doesn't need to be).

Digital signing + encryption



Alice knows that noone but Bob can see the message.
Bob knows it's really from Alice, and has not been modified.

Problems to be solved

- **Privacy**: Noone except the recipient may be able to see the contents.
- **Integrity**: The message must arrive unchanged. If is has been modified this must be detected.
- **Authentication**: It must be guaranteed that the message originates from the sender.

Implementations

- PGP / GPG
  - Pretty Good Privacy
  - Implemented by Phil Zimmerman.
  - Reimplemented as Gnu Privacy Guard.
  - Works with a ‘web of trust’.
- X.509
  - ISO standard.
  - Hierarchical model.
  - Works with Certificate Authorities.

PGP / GPG

- Pretty Good Privacy
- Implemented by Phil Zimmerman.
- Reimplemented as Gnu Privacy Guard.
- Identified by email address.
- Works with a ‘web of trust’.
- Command line tool to be used with mail tools.
- Supported by many modern email programs.
- Easiest to use with the Thunderbird plugin Enigmail.
- Can be used inline and with S-MIME.

Example PGP encrypted message

```
To: Johan Vromans <jvromans@squirrel.nl>
Subject: PGP demo 1
X-Mailer: VM 7.19 under Emacs 21.4.1

-----BEGIN PGP MESSAGE-----
Version: GnuPG v1.4.5 (GNU/Linux)
Comment: Processed by Mailcrypt 3.5.8 <http://mailcrypt.sourceforge.net/>

hQIOAz87uyr9kAavEAgAgWtn7HhhPnLzvVlgLPASy2K7XrSXmUO8ld3M9lElz8S/
90ldlFtRs0UUAzKVrxFDiZJO9NptTLpOoagp29eVWr5GnaBHM7IMwTunVYmoVny8
qqiFYPhXl9UyoLueukLM3jBPetMFeMN+pJLfwtRMWFBvQbXE5Qp8Uhb2kCDNmZRT
k8ngerHOYp2Pu39QJ7GBbsO0J6LOpfTjNoxXNP5PR4n2zfD8g2Otdzjr7gZ0WOFj
kkhyYirmslS9VQc/gWZPUfsLhwyScTlRmHZW33Waldo9rFXFClXJmf5Zve+S20F5
q8SlebenAlhLfTQNs4o6HrGkCPzH+CR8uJpGXvM2lggAkplXZBRbwwEVv7u3IDtz
NS+aA3SNtSmpq3KVb78hrUcmP27/aE1nK05D4zproFOouDNplLGhDV+lRyidBxkk
qmRzres5W3RnNDDiMvU+q7ycC8V5ML8W4VaVv7AcMwqaZPHSmMhsRMyVcQcw57KH
fs4+b0YVIEsaOjPPYZCifK2p8utSy2dQS/BDWORRBFZAzKtPIQ+HfTiY2FyH92LV
wdSKSqRQbcPGAlB7bYkuDNX8wkcpfYn5tslJa3UPkNP0y/oOGYIcz9CSJDCer4wB
WOXyBNFmgS1GnOFRV5kQjBnW3tRXaeamvfYTeYXj6fkztD7rWJC7wzdysyx5Mwi/
YNJVARmMZxW3D3qQob2QgTfZJfg4P5FOsnOo0F8i7WBmdkUlCkcF1Lyb7nMbLBFS
XS1trygabyiYIhU+gTyRC9jlbMDJQSrydnp05cbjBxWaGH1MIyXeuA==
=Ntx0
-----END PGP MESSAGE-----
```

Example PGP signed message

```
To: Johan Vromans <jvromans@squirrel.nl>
Subject: PGP demo 1
X-Mailer: VM 7.19 under Emacs 21.4.1

-----BEGIN PGP SIGNED MESSAGE-----
Hash: SHA1

Dit is een demo van PGP.
-----BEGIN PGP SIGNATURE-----
Version: GnuPG v1.4.5 (GNU/Linux)
Comment: Processed by Mailcrypt 3.5.8 <http://mailcrypt.sourceforge.net/>

iD8DBQFFXGlHDmJrrRKYwrQRAugUAJ0fZCySXr3vozrHqxWn44f6Oik3eACdElzF
e2bigRkrMdBlcwqVFwJpgzs=
=PjGY
-----END PGP SIGNATURE-----
```

Who’s key is this

- How do you know that the public key really belongs to who you think it does?
- You don’t.
- All you have is the email address.
- This is where the Web of Trust (WoT) comes in.

Web of Trust

- By meeting people in person you can validate their identities and public keys.
- When you are sure that a person really is who she says she is, you can add your digital signature to her key.
- By adding your signature, you state publicly that you have verified the owners identity.
- The more signatures are attached to a public key, the more chance that that key really belongs to who it claims it does.

Example PGP keyslip

```
---------------------------------------------------------------------------

pub 1024D/1298C2B4       2005-06-26 Johan Vromans <jvromans@squirrel.nl>
    Key fingerprint = 60B5 6330 872C FD47 B337 0B34 0E62 6BAD 1298 C2B4
sub 2048g/FD9006AF       2005-06-26

---------------------------------------------------------------------------

pub 1024D/1298C2B4       2005-06-26 Johan Vromans <jvromans@squirrel.nl>
    Key fingerprint = 60B5 6330 872C FD47 B337 0B34 0E62 6BAD 1298 C2B4
sub 2048g/FD9006AF       2005-06-26

---------------------------------------------------------------------------

pub 1024D/1298C2B4       2005-06-26 Johan Vromans <jvromans@squirrel.nl>
    Key fingerprint = 60B5 6330 872C FD47 B337 0B34 0E62 6BAD 1298 C2B4
sub 2048g/FD9006AF       2005-06-26

---------------------------------------------------------------------------
```

Example PGP keyslip

Johan Vromans
ir J.J.M. Vromans RI

Squirrel Consultancy

Open Source Software Consultancy

Cederlaan 6, 7875 EB Exloo, tel. 0591-548468, http://www.squirrel.nl

OpenPGP key: 1024D/1298C2B4 2005-06-26
Key fingerprint: 60B5 6330 872C FD47 B337 0B34 0E62 6BAD 1298 C2B4
Key uid: Johan Vromans <jvromans@squirrel.nl>

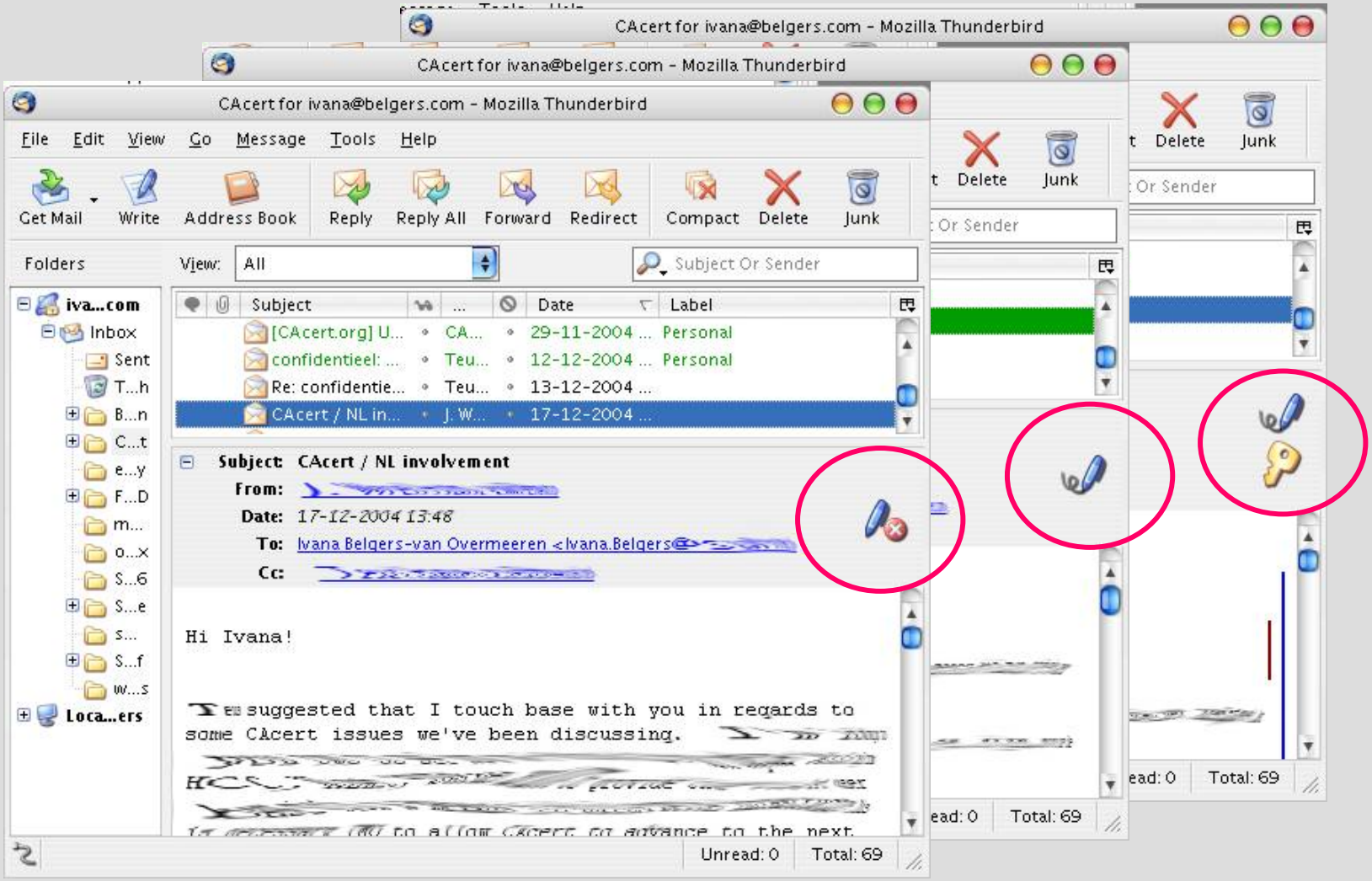
X.509 Digital Certificates

- ISO standard.
- Hierarchical model.
- Works with Certificate Authorities.
- Supported by most modern email programs.
- You can use it to attach digital signatures to PDF and OpenOffice documents as well.
- You can use it to set up secure web servers.
- You can use it for code signing.

X.509 Digital Certificates

- Certificates are identified by email address.
- To add your name, you have to identify yourself at the appropriate authority.
- Or gather trust points via a Web of Trust like approach.

Example X.509 use



Certification Authorities

- All commercial, except one.
- You need to pay, often a substantial amount, for your certificate.
- And again, and again, as certificates expire with time.
- Some provide free, but limited, services.

Thawte.com

- At Thawte.com you can get a free email certificate. By gathering trust points it can be augmented with your name.
- You can get trust point when you identify yourself to a Notary – someone who already collected a sufficient number of trust points.
- Supported by all major browsers and email programs.
- Register at http://www.thawte.com .

CAcert

- CAcert is a free CA.
- Free, since privacy and security should be available to anyone.
- Anyone can get fully functional certificates.
- For encryption, for digital signing, for web services.

Cacert is community work

- More than 10.000 assurers.
- World Wide.
- Translations into 30 languages.
- More than 100.000 certs in use.
- More than 100 people provide 7 $\times$ 24 email support.

Cacert is supported

- CAcert services run on Oophaga Foundation highly secured servers in the Netherlands.
- Sponsored by
  - HCC, NLUUG, NLnet
  - SUN/AMD, Tunix, Cisco, Net Apps
  - and hopefully by you too!

CAcert

- Certificates are identified by your email address.
- To add your name to it, you can collect trust points by meeting Cacert assurers.
- The assurers will verify your identity and give you trust points.
- At 50 points, you can have your name added to your certificate.
- At 100 points, you are an assurer yourself and can start giving points to other people.

CAcert

- There’s a small catch...
- The CAcert public key (root key) is not yet part of all major browsers and mail programs. You need to add it yourself.
- CAcert is currently being audited for inclusion in mainstream software.
- Register at https://www.cacert.org .