why PKI and digital signatures suck

some theories on real-world crypto usage

chaos communication camp 2003

christian mock

<cm@quintessenz.org>

http://www.tahina.priv.at/~cm/talks/pkisucks.{sxi,pdf}

version 0.2

contents

- this talk is not about
  - algorithms and code
  - cryptanalysis
  - PGP vs S/MIME
- it is about
  - PKI, commercial CAs, signature law
  - the painful experience of implementing real world systems with PKI
  - the politics of digital signatures, e-government etc.

alice and bob in crypto-wonderland?

Photo Credit:US National Oceanic and Atmospheric Administration

- in theory, assymmetric crypto with PKI is simple and elegant:
    - get a certificate from a CA
    - sign & verify
    - encrypt & decrypt
    - authenticate
- solves a lot of problems, easily

if it only were that simple...

- problems:
  - certificate authorities
  - standards: so many of them
  - handling of certificates
  - software bugs
  - digital signature laws

commercial certificate authorities

- *trusted* third party?
- sure: you're **forced to trust**
- ...and forced to pay
- ...and handle all the paperwork, yearly

are CAs trustworthy?

- 2001: verisign issues two "microsoft corp" SW signing certs to unknown persons
  - and windows apparently didn't use CRLs
- 2003: .at "bürgerkarte" shows wrong first name for holder
  - this is a legally binding signature certificate
  - identity checked with an official ID document

who owns the CAs?

- verisign belongs partly to SAIC
- SAIC's board members come from the NSA and the military...
- verisign bought up thawte, it's biggest competitor

- austrian CA a-trust is owned by a consortium of banks, lawyer's and notary public's association, chamber of commerce and incumbent telco

inhouse certificate authorities

- the "big" commercial solutions are complex and expensive
- CAs integrated in other systems (windows, FW-1, ...) are often not interoperable
- open source? no "ready to deploy" solutions
  - both OpenCA and pyCA need integration effort
  - both are too complex for smaller needs
- want to have your root cert signed by a commercial CA? spend even more money!

a certificate is a certificate is a certificate - not!

- there's (at least) four ways to put email addresses in X.509 certs
  - and in any real life project, you can be sure to meet at least three
- what to encode in a cert depends on the software using it
  - and that may mean your cert will *either* work with your VPN client *or* your email client
- and don't forget the many file formats and MIME types...

juggling certificates

- now which of those root certs to I need to download and trust in order to be able to accept Alice's certificate?
  - in the end, you "trust" them all until it works
- oh fsck, my cert expired, so I need to renew and send it to everyone I communicate with
  - and there is software out there which can only keep one "own" cert at a time
  - there's no single certificate server like with PGP keyservers

all client software sucks!

- 2002: MS software and Konqueror treat "end user" certs as CAs
  - easy MITM attack against those
  - complete breakdown of the PKI trust-model
- but ... nobody seemed to care much.

- lots of software are incomplete implementations (CRL support, features, interoperability, ...)

digital signatures

- these conclusions are drawn from the austrian situation, but should be similiar EU-wide (EU directive on electronic signatures)
- security requirements are very high:
  - "advanced signature" replaces manual signature
  - non-advanced signature not useable for eGovernment
  - needs a high level of security
    - keys & certs on smart card
    - smart card reader with display and keypad
    - "secure viewer" application for signing and verification

but: is it really secure?

- solutions run on windows -> weakest link
- CA doesn't even forbid usage on win9x/ME - no OS security features - what about malware?
- TOS say end user has to make sure no unauthorized code runs on his machine
- only a few "secure viewers" are approved -> easily targetted by malicious software

legal questions

- assume someone wants to dispute a digitally signed contract
  - could have been his secure viewer showing something other than the signed doc
  - could have been some malware sneaked the document thru his signing process
  - what if expert witnesses couldn't exclude those possibilities?
  - what if malware was actually found on the PC?
- what would that mean for the whole legal model?

technical/handling complexity

- .at signature law requires "certificate blocking" and "certificate revocation"
  - ever seen software that knows about "certificate blocking"?
- to verify a signature you need the components prescribed by the signer's CA
- in 2005, you'll need to re-sign all signed docs
  - signature law defines "lifetime" for algorithms and key lengths

solution in search of a problem

- why should I spend money on a cert + hardware if nobody accepts it?
  - and I make it harder for me to weasel out of stuff
- market for qualified signatures is very small
  - two austrian CAs merged last year (resulting in chaos for cert owners)
  - only 1 of 5 .at CAs issues qualified certs
- CAs try to push their services where people can't escape, e.g. universities

Bürgerkarte (Citizen Card)

"Further, with the Bürgerkarte a security infrastructure will be created which in the future will be accessible to citizens and therefore to the economy's customers. Companies can create secure online services for their customers and use the Bürgerkarte infrastructure. The Bürgerkarte therefore helps to dispel a fundamental restraint of electronic business - the presently lacking trust in the transaction's security."

www.buergerkarte.at (translation: cm)

Bürgerkarte...

- even those who made the signature laws have to relax the requirements
  - "Bürgerkarte light": keys stored on service provider's server, signatures confirmed via mobile phone + SMS
  - additional key pairs required because you can't authenticate with a qualified certificate
  - public sector allowed to sign without qualified cert
- need Bürgerkarte + payment service + delivery service accounts
- one single application exists ATM

conclusions

- PKI/CA security model is flawed
  - mix of high-security components and low-security components
  - human error not taken into account
  - possible scaling problems (CRLs)
  - gets watered down even by the makers of sig laws
  - nobody cared when it was completely broken
- handling and interoperability problems
- who needs that stuff, anyways?

alice and bob in reality-land
(or: enough of that bitching!)

- PKI and public-key crypto has its place
  - try to avoid falling under signature laws
  - build your own inhouse CA
  - make your own risk assessment and select appropriate (and realistic) security measures

real-world example

- customer wanted secure, easy VPN authentication for road-warriors and in-house WLAN
- risk analysis: cert-based auth is more secure and easier to handle than username/password
- built CA software on openssl and perl
- works fine, customer thinks about using it for email encryption with business partners

discuss!

- thank you for listening
- now it's time for fundamental discussions on my theories

cm@quintessenz.org

http://www.tahina.priv.at/~cm/stuff/pkisucks.{sxi,pdf}