Identity-Based Cryptography: Panacea or Pandemonium?

Kenny Paterson

Information Security Group

Royal Holloway, University of London

kenny.paterson@rhul.ac.uk

Royal Holloway University of London

Definitions

Panacea:

A remedy for all diseases, evils or difficulties; a cure-all.

Pandemonium:

"…*Pandemonium*, Citie and proud seate
Of *Lucifer*…"

Paradise Lost, John Milton, 1667.

(Colloquially, any noisy or unpleasant place.)

Overview

- Public key cryptography.
- Why is PKI so hard?
- Identity-based cryptography – a panacea?
  - Basic description and features.
  - Example applications.
- Advertising break.
- Identity-based cryptography – pandemonium?
  - A more detailed look at identity-based cryptography.
  - Patents.
- Conclusions

Public Key Cryptography (PKC)

- Also known as asymmetric cryptography.
- Each user has two keys: public and private.
- Alice's public key typically used for:
  - encryption to Alice by Bob
  - verification of Alice's signatures by Bob
- Alice's private key typically used for:
  - decryption by Alice
  - signing by Alice
- No need for Alice and Bob to share a common key before they begin secure communications!
  - Compare with symmetric key cryptography.

The Need for PKI

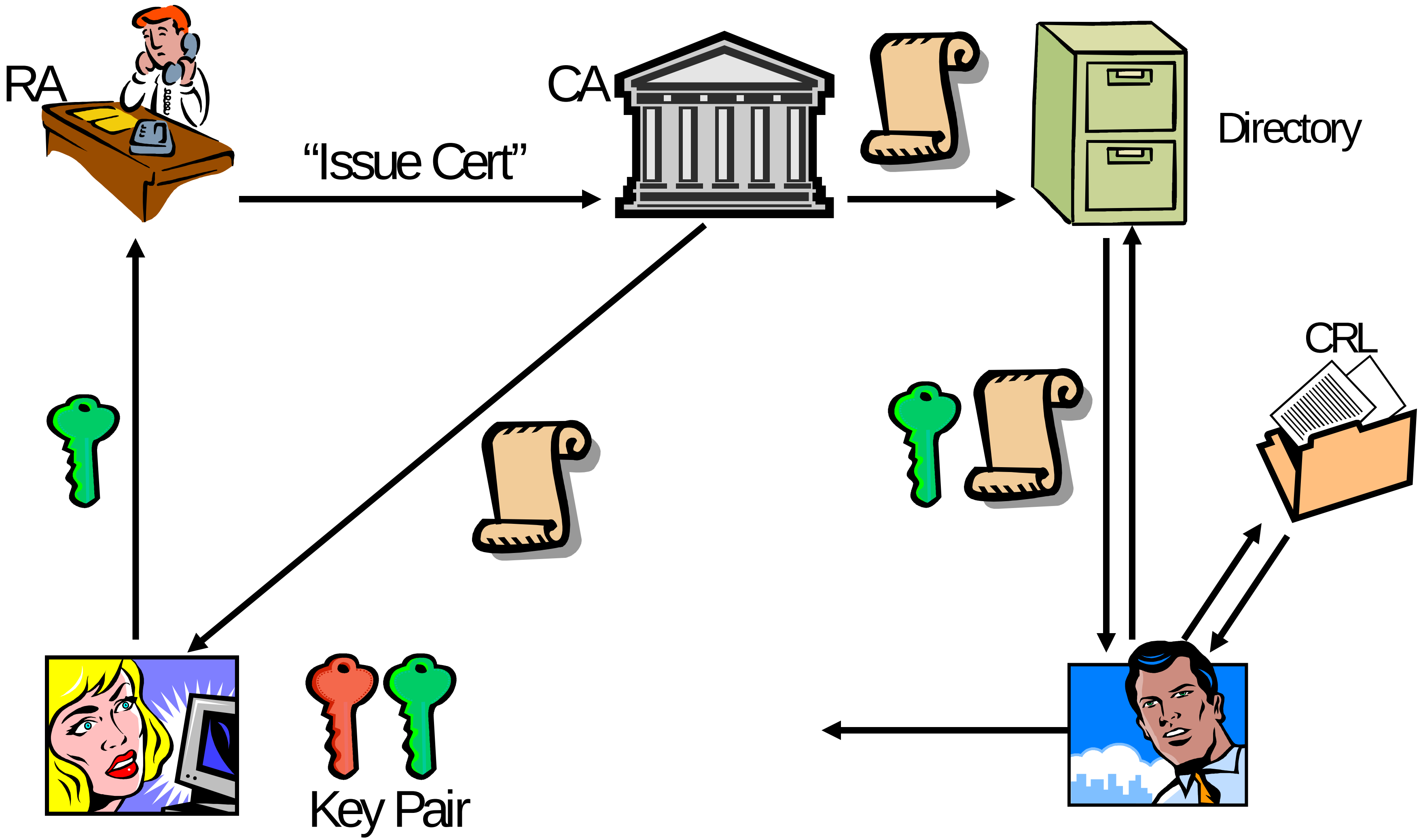
- We need some way of enabling Bob to actually find Alice’s key.
  - A directory service for encryption applications.
  - Or delivered as part of a protocol, or along with a signature.
- But how does Bob know that Alice's public key really is Alice's (and not Eve's)?
  - We need some way of binding public keys with identities.
  - Certificates in most circumstances.
- We will also need some way of signalling that a public key is no longer to be relied upon.
  - Alice’s private key might become exposed, or she might leave the company.
  - A revocation mechanism.

Royal Holloway University of London

PKI Components

- Registration Authority (RA)
  - Authenticates individuals/entities, optionally checks for possession of private key matching public key.
  - Passes off result to Certification Authority.
- Certification Authority (CA)
  - Issues certificates: CA issues signatures binding public keys and identities.
  - Relying parties need authentic copy of CA's public key…
- Directory Service
  - Directory of public keys/certificates.
- Revocation Service
  - May involve distribution of Certificate Revocation List (CRL) or on-line certificate status checking (OCSP).

Using the Infrastructure



Example PKIs

- SSL server certificates, authenticated via root certificate embedded in browser
  - Certificate hierarchy.
  - Provides server (not client!) authentication for e-commerce.
  - Rare example of open PKI.
- IPSec certificates
  - Gateway-gateway VPN and remote access solutions.
  - PKC enables authentication of endpoints via IKE protocol.
  - Generally closed PKI.
- Identrus PKI
  - Trust for b2b commerce, banks acting as CAs.
  - Complicated set of rules and contracts needed to define roles, responsibilities and liabilities.
  - Closed PKI.

Some PKI Problems

- Infrastructure should be largely invisible, but PKI often isn't.
- Legal and regulatory concerns.
- Interoperability and standards.
- Deployment and on-going management of costly and complex infrastructure.
- Commercial/business issues.
- The bottom line: in commercial circles, PKI has come to be seen by many as an over-hyped technology which has not lived up to its promise.

Complexity and PKI

- There is a massive complexity gap between the *concept* of public key cryptography and its *realisation* in the form of a traditional PKI.
- From an application perspective, the ability to provide non-repudiation seems to be the unique feature separating public key from symmetric key.
  - Once one appreciates the real-world complexities, symmetric key systems appear equally attractive in many circumstances!
- Certificates and their management are the source of some problems.
  - So somehow getting rid of certificates might help?

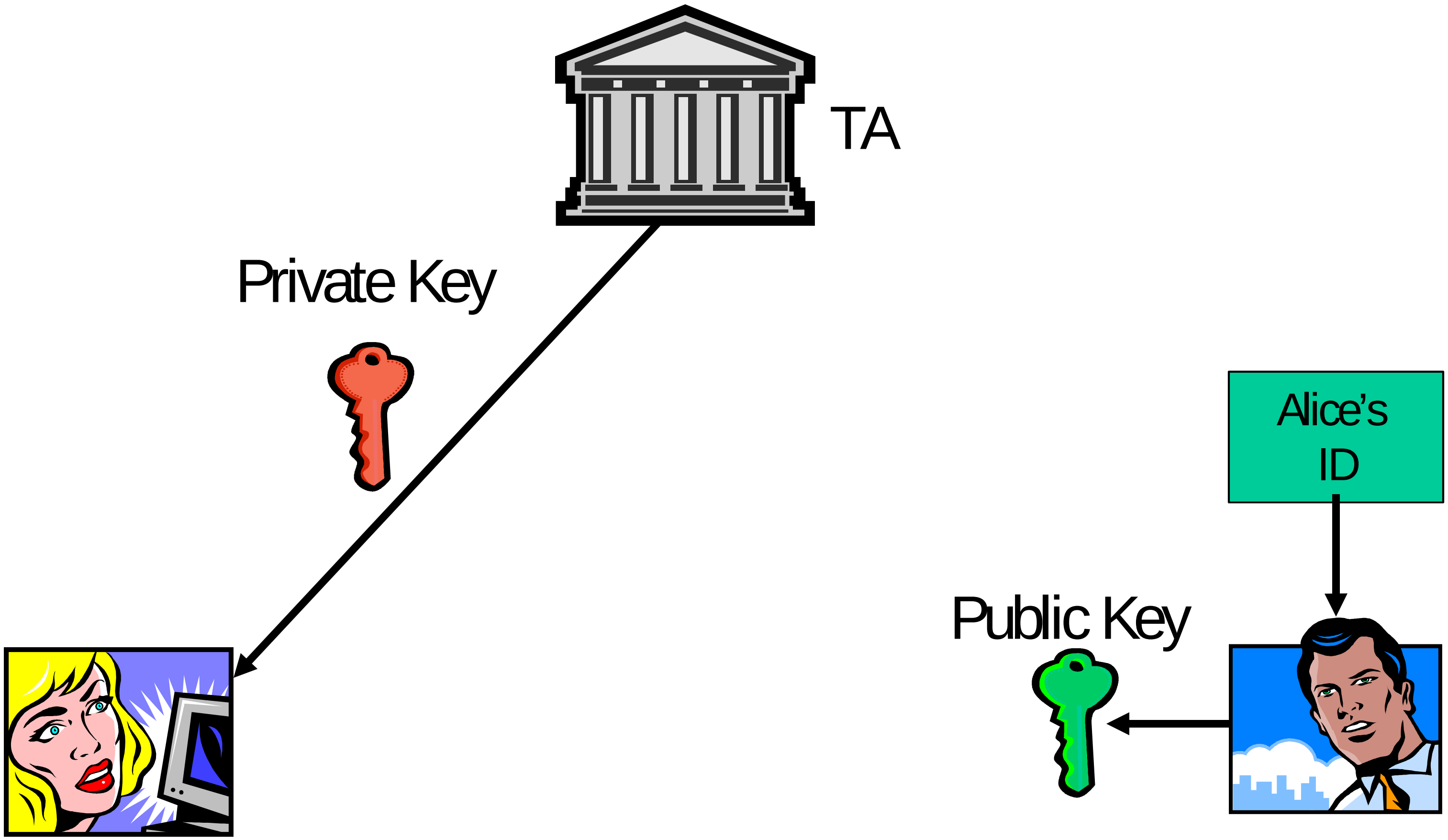
Identity-based Cryptography

Original idea due to Shamir (1984):

- Public keys derived directly from system identities (e.g. an e-mail address or IP address).
- Private keys generated and distributed to users in by a trusted authority (TA) who has a *master key.*
- As long as:
  - Bob is sure of Alice's identity and
  - The TA has given the private key to the right entity,

  then Bob can safely encrypt to Alice without consulting a directory and without checking a certificate.

Identity-based Cryptography



Identity-based Cryptography

- Apparently, elimination of certificates produces a far simpler infrastructure.
  - We'll examine this in more detail soon…
- *Identifier* often used in place of *identity.*
  - Reflecting idea that any string can be used to derive public keys.
- IBE = Identity/Identifier-based encryption.
- ID-PKE = ID-based public key encryption.
- ID-PKC = ID-based public key cryptography.

IBE – A Short History

- Shamir devised only an ID-based signature scheme.
- Construction of *truly practical* and *secure* ID-based encryption scheme an open problem until 2001.
  - Several insecure/inefficient proposals.
- Sakai, Ohgishi and Kasahara (SCIS, Jan. 2001)
  - Written in Japanese.
  - Pairing-based scheme, but no security model or proofs.
  - English version apparently rejected from Asiacrypt 2000.
- Boneh and Franklin (Crypto, August 2001)
  - Written in English.
  - Pairing-based scheme, practical and provably secure.
- Cocks' scheme (IMA Conference, Dec. 2001)
  - Scheme based on quadratic residuosity, not bandwidth efficient.
  - Research done in mid 1990's at UK government agency.
  - B&F paper prompted publication of Cocks' work.

Apparent Benefits of ID-PKC

- Certificate-free.
  – No processing, management or distribution of certificates.
- Directory-less.
  – Bob can encrypt for Alice without looking-up Alice's public key first.
  – Indeed, Alice need not have her private key when she receives Bob's encryption.

Apparent Benefits of ID-PKC

- Automatic revocation.
  - Simply extend identifier to include a validity period.
  - Alice's private key becomes useless at end of each period.
  - Alice needs to obtain private for current period in order to decrypt.
  - No need for CRLs or OCSP.
- Built-in support for key recovery.
  - When Alice leaves the organisation (or is run over by a bus).
  - Also enables applications like content scanning of e-mail at server.

Applications of ID-PKC

- ID-PKC and pairing-based crypto have undergone an extraordinarily rapid development since 2001.
  - Paulo Baretto's Pairing Based Crypto Lounge (no longer being updated?)
  - Apparent extensive use of Bellare's crypto topic generator.
    - http://www-cse.ucsd.edu/users/mihir/crypto-topic-generator.html
  - Growing commercial interest.
- Potential applications for ID-PKC
  - Secure e-mail.
  - Cryptographic workflow.
  - Domain-based security, GRID security architecture, securing router advertisements, ad hoc networks,…

ID-PKC and Secure e-mail

- ID-PKC seems well-suited to encryption for e-mail and other messaging technologies in corporate environments.
  - Natural candidate for TA.
  - Low interaction with infrastructure for sender.
  - Recipient of encrypted e-mail need not be pre-enrolled.
  - Key recovery feature allows message hygiene services to be conducted at mail server/organisational boundary.
  - Potential for lower costs through lightweight infrastructure requirements (compared to PKI-based solution).
  - Seems likely to be first mass-market application of ID-PKC:
    - Voltage Security: www.voltage.com

ID-PKC and Secure e-mail

- Is secure e-mail the killer application?
- Voltage Security certainly hope so:

"IBE easily solves some of the problems that have traditionally made implementing and supporting encryption technology difficult and expensive."

Luther Martin, Principal Engineer, Voltage Security in *"Identity-based encryption: a closer look", The ISSA Journal*, June 2005.

ID-PKC and Secure e-mail

But…

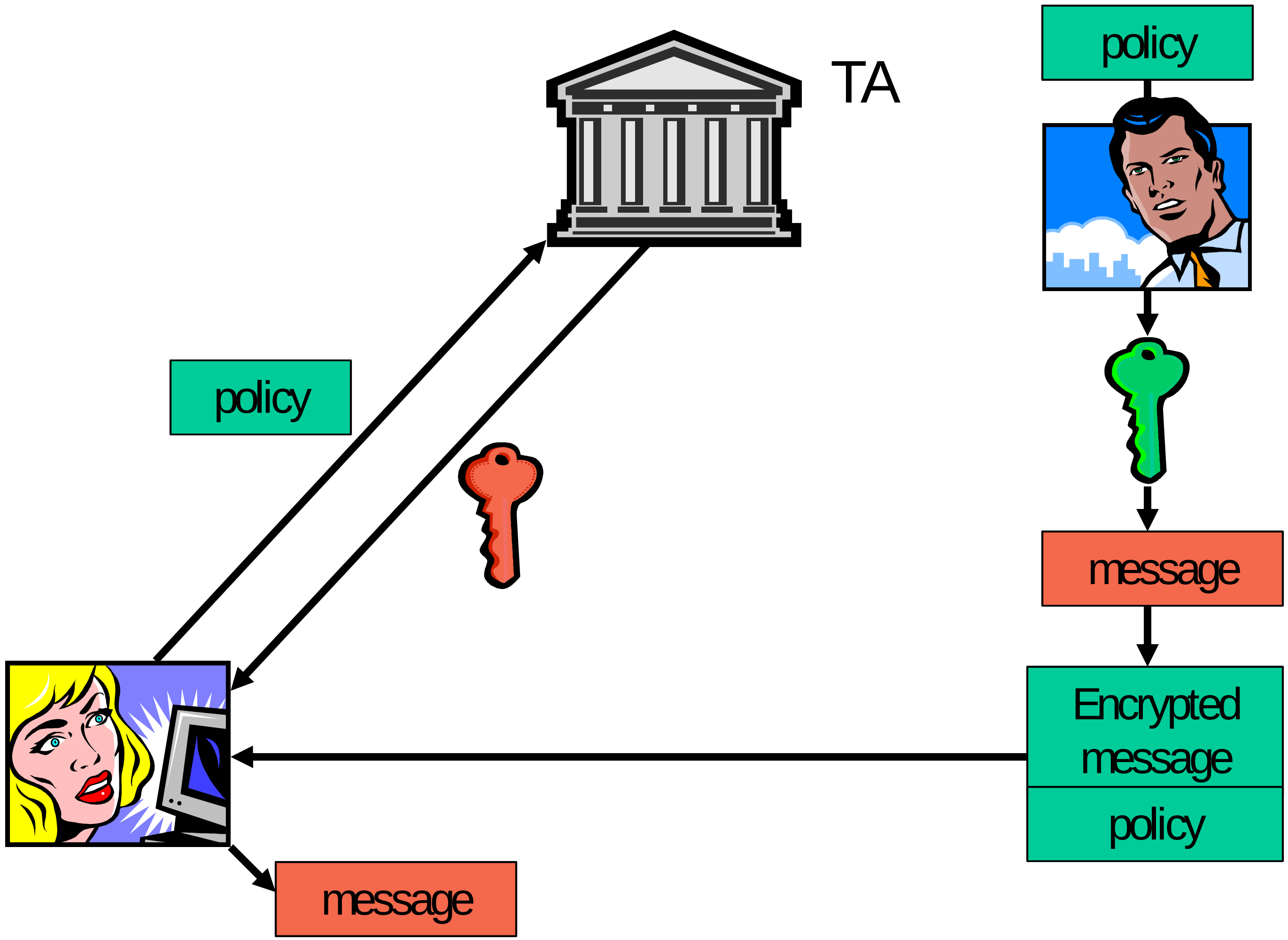
- Difficult to build non-repudiation services.
- May need to integrate with existing PKI-based authentication services.
- Voltage Security whitepaper, March 2005:
  - *“Combining IBE with PKI enables a secure messaging environment to benefit from the advantages of both systems.”*
- Do we really need secure e-mail anyway?
  - Lots of hype around SOX, HIPAA, GLBA,…

Cryptographic Workflow

- Identifier could be *any* string
- What if public key determined before private key?
  - Bob selects identifier string expressing a policy.
  - Bob encrypts message of value to Alice using public key matching the identifier.
  - Bob relies on TA to only release matching private key if conditions expressed in policy met by Alice.
- TA becomes a *decryption policy enforcer.*

Royal Holloway University of London

Cryptographic Workflow



Example of Workflow

- Bob selects identifier for Alice:

  Identifier = "Alice && over 18 && transaction value < $100".

- Bob sends Alice content encrypted under public key derived from this identifier.
- Alice convinces TA she satisfies conditions expressed in the identifier – age and limit on transaction value.
- TA then gives Alice private key matching identifier.
- Finally, Alice can decrypt to obtain content.

Royal Holloway University of London

Workflow Extensions

- Bob selects identifier for Alice:

  Identifier = “Alice && over 18 && transaction value < $100”.
- Now each component of policy corresponds to private key from different TA.
  - TA vouching for identity.
  - TA vouching for age.
  - TA handling payments.
- Alice convinces each TA in turn that she satisfies conditions expressed in the identifier.
- Alice gets a private key component from each TA and combines them to produce her final private key.
- Alice can decrypt to obtain content.
- Arbitrary Boolean expressions can be handled
  - Smart; Al-Riyami, Malone-Lee and Smart; Bagga and Molva,…

Workflow via PKI

- Cryptographic workflow is a nice idea, but it doesn't actually require ID-PKC …
  - TA has become policy enforcer, trusted to perform certain actions.
  - Now high degree of interaction between Alice and TA.
    - Each new policy is likely to be unique and require fresh private key.
- Alternative approach with same trust assumptions and message flows:
  - Bob encrypts content under TTP's (ordinary) public key and sends to Alice along with policy for decryption.
  - Alice takes encrypted content to TTP who decrypts it for Alice, provided Alice satisfies policy.

Further Applications of ID-PKC

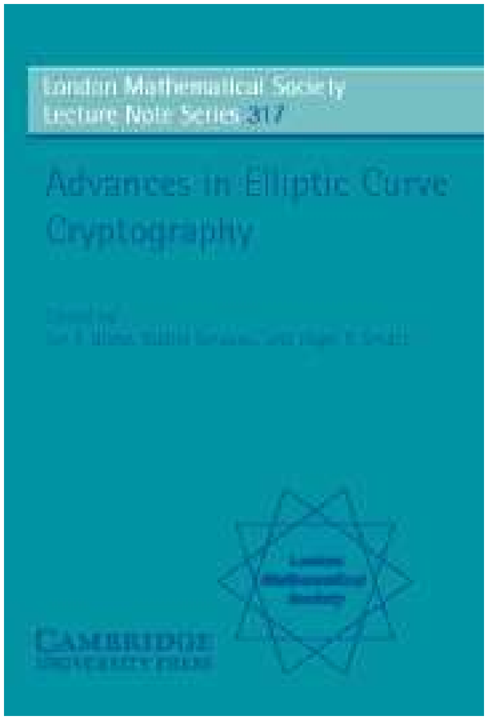
- Domain-based security (Smetters and Durfee, 12th USENIX Security Symposium, 2003).
  - Each DNS domain acts as TA for clients in the domain.
  - Use DNSSEC PKI to authenticate TA parameters.
    - Adapt DNS to transport TA public parameters between domains.
    - Support for inter- and intra-domain IP and e-mail security.
  - Various mechanisms for private key distribution including:
    - SSL (possibly with client certificates based on PKI!)
    - Distribution via e-mail to authenticate clients.
    - Or transmission over trusted network segment.
  - Proof of concept coded in Java on Linux.

Further Applications of ID-PKC

- GRID security (Lim and P., preprint).
  - Pure-ID-PKC architecture designed to meet security requirements for GRIDs:
    - Single Sign-On.
    - Delegation via proxying.
    - Secure channels.
  - Use of Gentry-Silverberg hierarchical ID-PKC to handle hierarchy of root TA, local TA, user, and user proxy.
  - Exploit identifiers to specify delegation policies, reduce round-trips and ease revocation.
  - ID-based version of SSL handshake protocol.
  - Select ID-PKC parameters to minimise bandwidth and computation.

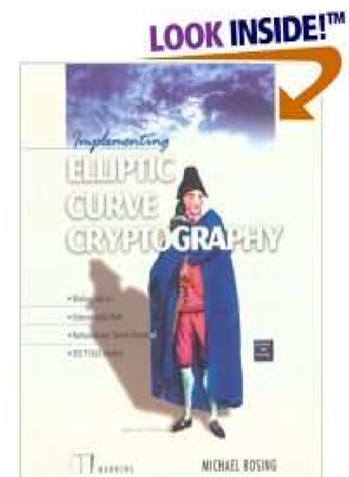
Royal Holloway University of London

Advertising



- Advances in Elliptic Curve Cryptography
- Cambridge University Press, LMS Lecture Note Series, Volume 317.
- ISBN 0 521 60415 X.
- Editors: I.F. Blake, G. Seroussi, N.P. Smart.
- Contributors: N.P. Smart, D. Brown, A.W. Dent, E. Oswald, M. Joye, F. Vercauteren, P. Gaudry, F. Hess, S.D. Galbraith, K.G. Paterson.

"Other elliptic curve cryptography books are available."



ID-PKC – Pandemonium?

- Focus so far on positive aspects of ID-PKC: certificate-free, directory-less, automatic revocation and key recovery.
- We've not really examined the operational issues associated with ID-PKC.
  - Only hinted at difficulties of private key distribution and the non-repudiation issue.
- Now we take a closer look…

…and discover that ID-PKC is not as straightforward as it at first appears.

Public Parameters

- Bob needs an *authentic copy* of the TA's public parameters before he can safely encrypt to Alice.
  - To prevent man-in-middle attacks.
- One solution is to hard-code TA parameters into client applications.
  - Could be OK for closed applications, but not very flexible.
  - Could use hierarchical approach to support multiple applications and parties.
- Another solution:
  - Certify TA parameters using a PKI.
  - A hybrid solution, as adopted in Smetters and Durfee.
  - Still need to distribute and check these certified parameters.

Registration

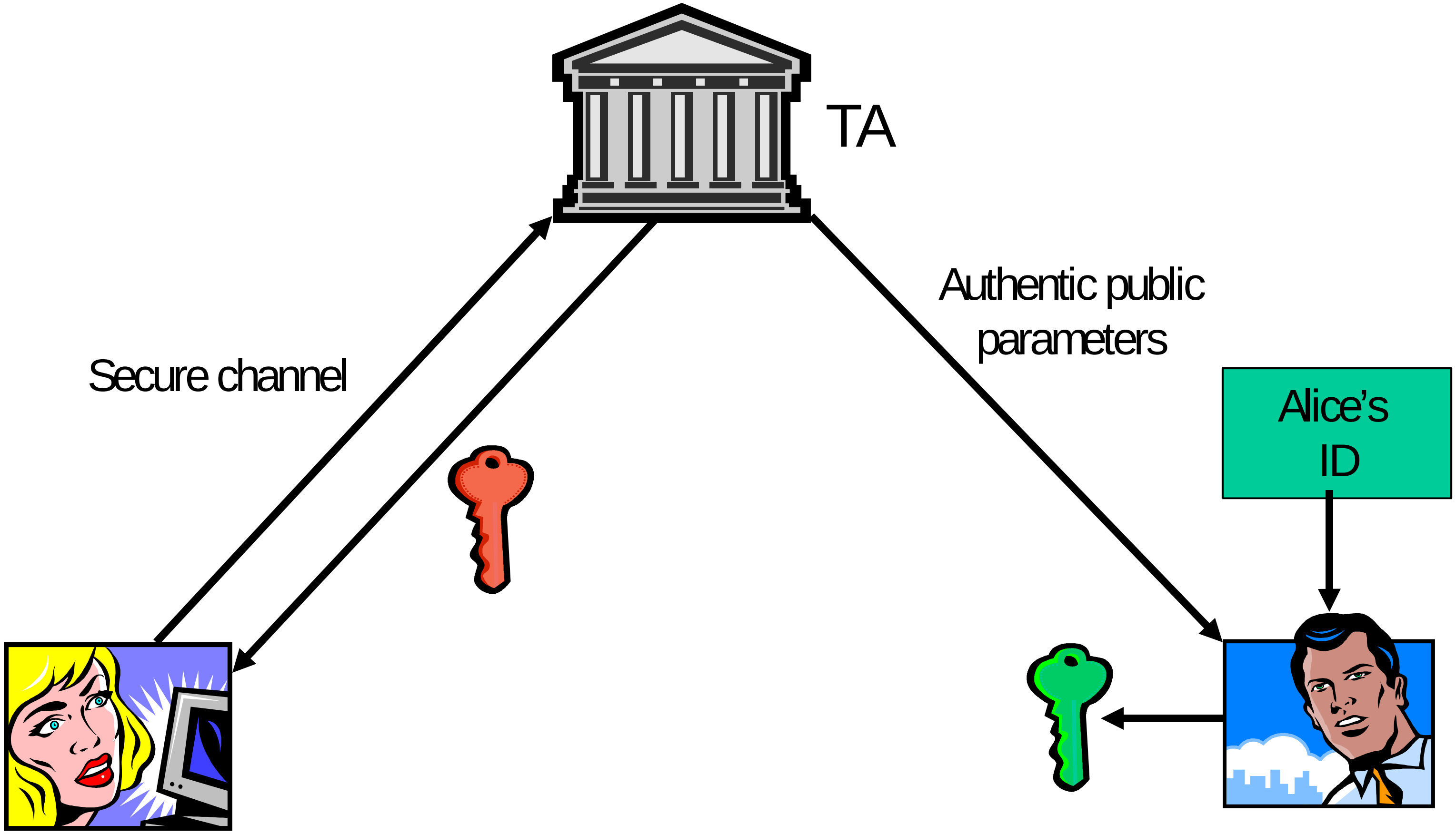
- A secure enrollment process is still needed.
  - *Pre*-enrollment *can* be avoided, but Alice does need to enroll at some point!
  - Secure process needed to ensure that Alice's private key is really being delivered to Alice.
    - PKI only needs an authentic channel.
    - ID-PKC needs a channel that is both authentic *and confidential*.

Royal Holloway University of London

Registration

- A secure channel is needed for registration and delivery of private keys.
  - How is this to be achieved in practice?
  - How often will the channel be used?
  - What security level does it need to provide?
    - For example, is delivery via e-mail appropriate?
  - If we have such a channel, what alternative uses might be found for it?
  - Where should we store private keys once we've distributed them?

Reality of ID-based cryptography



Effect of Catastrophic Compromise

What is the cost of compromise of the master secret?

- Potentially higher than cost of compromise of CA signing key in PKI:
  - CA in PKI could re-issue all certificates under new signing key.
  - No client private keys are compromised.
  - Only temporary exposure to threat of rogue certificates being used by encrypting/verifying party.
  - Meanwhile, in ID-PKC, all past encrypted messages are exposed and all old signatures become worthless.
- In reality, a CA/TA compromise is unacceptable in either architecture.
  - In both cases, appropriate steps to prevent occurrence are needed.

Key Escrow

The other side of key recovery:

- TA can calculate all the private keys in the system.
- PKI is more flexible in this respect.
- May limit applicability of ID-PKC to certain applications where some degree of trust in TA is inherent.
  - In fact, open PKIs are largely a myth and many PKIs operate under similar trust assumptions anyway.
- Split TA or certificateless PKC as possible solutions.

Inability to Provide Non-repudiation

- Another consequence of key escrow.
- TA *could* forge signatures if an ID-based signature were adopted.
  - So need to trust TA not to do that.
- However, EU electronic signature legislation requires private key to be under "sole control" of signer in order for signatures to be fully recognised.
  - So may be incompatible with some legislative regimes.
- Since certificate can always be sent along with signature, ID-PKC does not seem to have a big advantage here anyway.
- Then why do we have so many ID-based signature algorithms???

Non-repudiation (ctd.)

- In fact, use of ID-based signatures would be reasonable in some (many?) applications:
  - True non-repudiation is not always needed.
  - Non-repudiation rarely enforced using legislation, but rather by PKI scheme rules and contracts.
  - ID-PKC scheme rules could permit use of ID-based signatures, provided appropriate trust relationships in place.
  - (But we still don't need 27 different signature algorithms!)

Revocation in ID-PKC

- A revocation mechanism is needed in ID-PKC just as in traditional PKI.
  - In event of key compromise or change of status of entity related to identifier.
  - But how can you revoke an identifier?
- The simple "automatic revocation" solution:
  - Bob simply extend Alice's identifier to include a validity period.
  - Granularity of expiry times determines rate of private key issuance (yearly, weekly, daily,…).
  - Could conveniently specify expiry policy in TA's parameters.
- Hence "no need for CRLs or OCSP".

Reality of Revocation in ID-PKC

- Granularity also determines maximum length of exposure period between compromise of private key and update of public key.
- So higher security application would need shorter validity period and hence higher rate of private key issuance.
  - Extra workload on TA.
  - TA may need to be highly available.
  - Secure channel needs to be used at frequent intervals.
    - Should be invisible to users.
    - Could use previous identifier and private key if not compromised.

Reality of Revocation in ID-PKC

- In a PKI, a (delta) CRL can be pushed out at regular intervals limiting exposure period.
  - Or even every time a key is compromised,
- This is not true of the automated revocation mechanism.
- Ultimately, in high security applications, real-time information concerning status of identifiers/private keys will be needed.
- Then an OCSP-like solution will be required.
- Where is the cross-over point where OCSP becomes more cost-effective than automatic revocation?
  - Detailed comparison needed.
- Reality: an effective revocation mechanism requires the timely distribution of authentic status information, irrespective of which public key technology is used.
  - Automatic revocation may not always be appropriate for ID-PKC.

A Thought Experiment

- Imagine situation where fine-grained identifiers are in use.
  - E.g. workflow application or frequent automated revocation.
- Then TA is on-line and frequent use is made of secure channel between TA and clients.
- If the channel is sufficiently secure and convenient to support this, what else could it be used for?

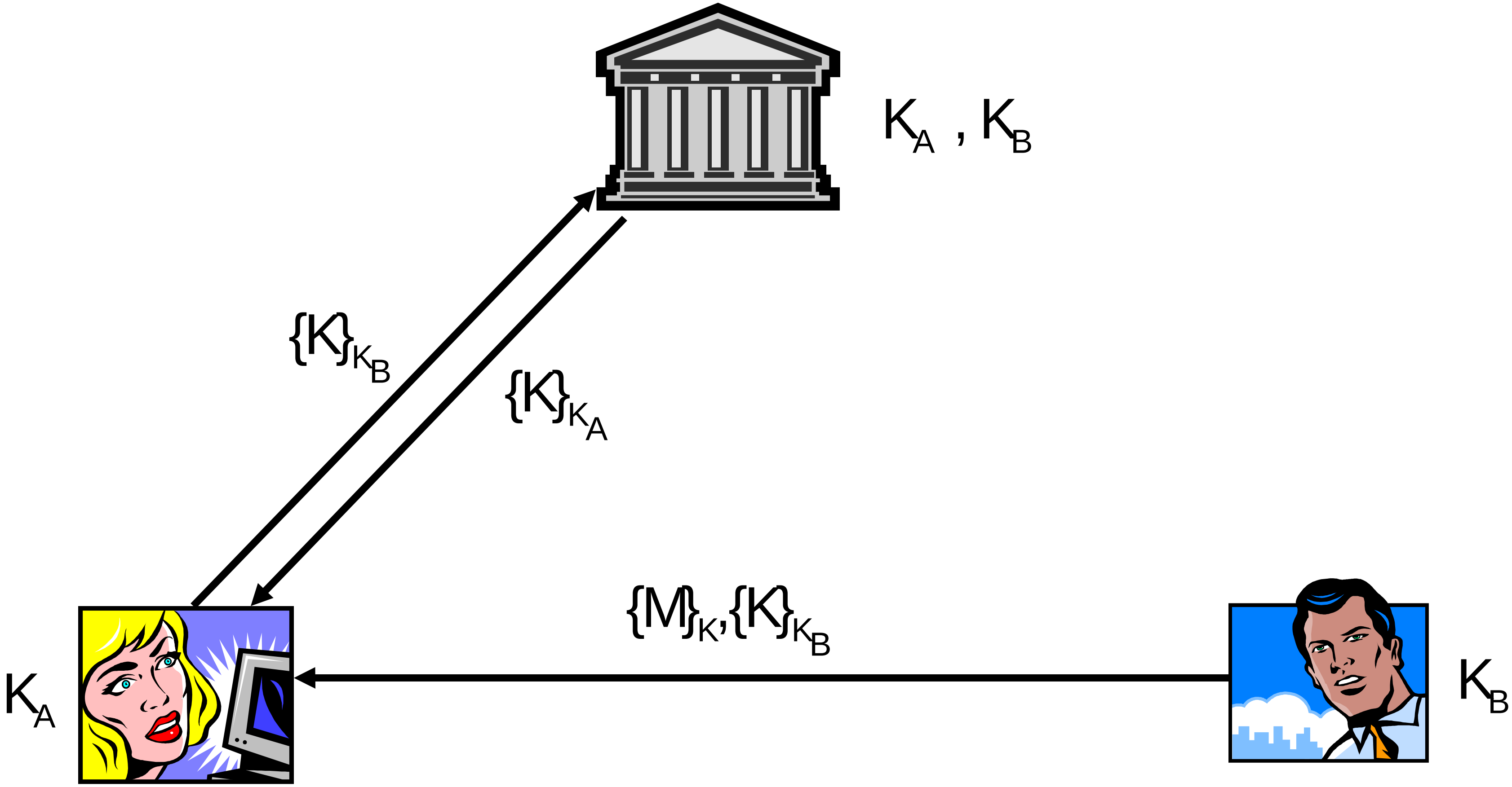
A Thought Experiment

A radical proposal: turn the TA into a KDC distributing *symmetric* keys to Alice and Bob.

- Assume Alice and Bob each have secure channel with TA/KDC.
- Use secure channels between KDC and users to distribute session keys.
- Session keys then used to protect application data between Alice and Bob.
- Canonical example: Needham-Schroeder protocol.
  - Similar message flow to ID-PKC approach.
  - Can be done without Bob ever contacting KDC.

Royal Holloway University of London

A Symmetric Approach



Analysis of Thought Experiment

- What have we lost with this symmetric approach?
- Apparently, only the ability to provide non-repudiation services!
  - Recall, we agreed earlier that this was the unique feature separating public key from symmetric key.
- But ID-PKC doesn't provide true non-repudiation!
  - In fact, KDC can provide arbitrated non-repudiation through use of symmetric key only.
  - Similar level of trust required in KDC as in TA.
- So we've lost nothing at all?
  - Maybe only a few extra protocol flows.
  - And no pairing calculations needed (sorry Mike and Paulo!)

Patents

Warning!

Warning!

Warning!

- I am not a patent attorney, just an interested bystander.
- Nothing I am about to say concerning patents should be interpreted as a legal opinion.
- Nothing here is intended to be against the interests of any particular party or parties.

Patents

- “Systems and methods for identity-based encryption and related cryptographic techniques”.
- Inventors: Dan Boneh and Matthew Franklin.
- US application 10/218,697.
- Provisional application filed August 13th 2001.
- Published May 1st 2003 (Pub No US 2003/0081785).
- Available for free from US patent office.
- Not yet granted.
- 82 claims in published version.
- Most claims concerned purely with IBE using bilinear maps.

Patents

- Claim 70:

  "A method of providing system parameters for a cryptographic system comprising: providing a system parameter representing an algebraic group G_1 and an algebraic group G_2 and providing a system parameter representing a bilinear map \hat{e} mapping pairs of elements of G_1 to elements of G_2."

- Appears to attempt to cover all pairing based cryptography using modified pairings!
- Yet there appears to be significant prior art using modified pairings in cryptographic settings.
  - At least Verheul's paper at EUROCRYPT 2001.
  - This paper is referenced in Boneh and Franklin's CRYPTO 2001 paper.

Patents

- Quite common for claims to cover more than has actually been demonstrated in a patent application.
- But:
  - Existence of Verheul's work may technically invalidate broadest claims covering pairing-based cryptography.
  - The work of Sakai *et al.* from SCIS 2001, if regarded as having been in the public domain prior to August 13th 2001, could potentially invalidate *all* the claims.
- Even so, the US patent may still be granted intact.
- Detailed analysis of US 6886096 (granted patent) may also be interesting.

Patents

- It is perfectly reasonable for inventors to seek intellectual property protection for their work.
- But legal uncertainty surrounding the technology may actually hinder its widespread adoption.
  - Haven't we all been here before with ECC?
  - Lack of standardisation also an issue here.
  - P1363 activity now proposed.
- Alternative approaches to ID-PKC which seek to avoid existing patents/patent applications are under development.

Complexity and ID-PKC

- There is a complexity gap between the *concept* of ID-PKC and its *realisation* in real-world applications.
  – Doesn’t this sound familiar?
- This makes certain initially attractive applications less compelling in practice.
- Getting rid of certificates helps.
  – But maybe not as much as we’d like to think…

Conclusions

- Traditional PKI has well documented problems and has not met (unrealistic) market expectations.
- Identity-based cryptography as an alternative
  - Solves some problems but introduces others.
  - Not the right choice for every application.
  - May be best suited to "corporate" or domain-restricted/closed applications where there is a natural choice for the TA.
- Lessons from history:
  - Avoid over-egging the pudding with unsupportable claims for the technology.
  - Don't misjudge the size of the gap between cryptographic theory and security practice.
  - Patents are legitimate tools, but can decelerate uptake of technology.
  - Don't forget about symmetric key cryptography.

Acknowledgements

- Talk based on joint research over last few years with: Sattam Al-Riyami, Hoon Wei Lim, Fred Piper, Geraint Price.
- PKI club: a research forum sponsored by Abbey, APACS, Barron McCann, beTRUSTed, BT exact Technologies, CESG, Hewlett-Packard Laboratories Bristol, Indicii Salus, Mondex and Prudential.
  - http://www.isg.rhul.ac.uk/research/projects/pkiclub
- Thanks also to the organisers of ECC9.