IBE: Is It Needed?

Quick Notes

• It’s easy to conflate IBE with Bilinear Map PKC
• I will try to stay to IBE in general
• Comments are not about implementations
  – Except when noted
  – Most comments apply to my IBE as well as others
• Everything has plusses and minuses
• Everything has appropriate uses
Inherent Brittleness

- All IBE (and all BLM) crypto has
  - A basic "key"
  - Subordinate, derived keys
  - This is the PKG for IBE
- This means that rollover, revocation, expiration, etc. are hard
  - And they're hard in unique ways
  - There are also interesting solutions
    - Identum, for example, has one PKG for all users
    - This is the Mark Twain solution
      - ”Put all your eggs in one basket and then watch the basket”

Naming is Hard!

- Nearly all rough edges of PKI reduce to naming problems
  - Ellison, Schneier, others point this out
- Some PKI systems are key-centric (SPKI)
  - Key-centrism exists because naming is hard
- Reducing a key system to naming removes the easy periphery
- We’re still left with the hard, thorny central issue
  - The thorny issue of naming is arguably harder with IBE
  - Since every name is a key, managing keys is managing names
Networks Help Solve Naming Issues

• The core IBE advantage:
  – Key = F(Name)
• Can be satisfied with a database / directory

• Is IBE needed when you can easily look up keys from the net?

Online vs Offline IBE

• Offline IBE can compute K=F(N) with no network
• Online IBE uses the net to compute K=F(N) via the network
  – I presented this in 2006
  – Trades online-ness for ability to use traditional keys/certs
    • RSA, DSA, Elgamal, EC variants, etc.
    • Even works for Lattice, hash-chain, etc. PKC
Is true IBE possible?

- Names alone are not enough
  - Even original Boneh-Franklin paper has a name of:
    - “bob@company.com □ current-year”
    - “bob@company.com □ current-date”
- Ironically, this is Certificate-Based Encryption
- Metadata is important!
  - Current trends create more metadata
- Names alone have no metadata
  - Lambda naming alone is good math, and bad information science

How Do I Own a Name?

- It is trivial for me to prove I own the string “jon” to my server
- It is difficult for me to prove I own “jon” to your server
- It is easy for a server to assign a name
- It is hard to correctly assign a name
- Many entities have many names
  - These turn in to many keys
Key Management is Still Hard

- IBE creates many keys per name
  - bob@company.com via bank.com
  - bob.lastname@company.com via bank.com
  - bob@home.com via merchant.com
  - bob.lastname@company.com via …
- Result is a sparse matrix of:
  - All your names * All the PKGs
- Thus the key management problem
  - Is very easy for each PKG
  - Grows in $n^2$ complexity for all users
- End-users are notoriously bad at complexity

Thank You