The Keys to The Kingdom: Key Management

Tim Polk

Cryptographic Technology Group, NIST

September 11, 2012

Disclaimer

- I am not a motivational speaker, regardless of what the agenda implied!
  - That will teach me to submit my slides, or at least a title, on time.

- It would be fair to say I am the head cheerleader for key management, though.

Goals, Assumptions & Cold Realities of Cryptography

- Goals for Algorithm Designers
  - For symmetric encryption algorithms, an algorithm is secure if an attacker given the algorithm and some ciphertext encrypted under an unknown key cannot practically derive any information about the message (other than its length) or the key
  - For public key algorithms, an algorithm is secure if an attacker given the algorithm, public key, and a signature or ciphertext cannot practically forge a signature, decrypt a message, or obtain the key
- Implicit Assumptions – the attacker cannot get the secret/private keys through other means
- Cold realities - these assumptions often do not hold!

If We Could Accept the Implicit Assumptions

- Then we wouldn't have a problem
  - Cryptographic algorithm designers are really really good at their work
- The real world is complicated – it involves people, software, hardware…
  - Users, system and network admins, CKMS operational staff, etc.
  - Cryptographic algorithms, storage device

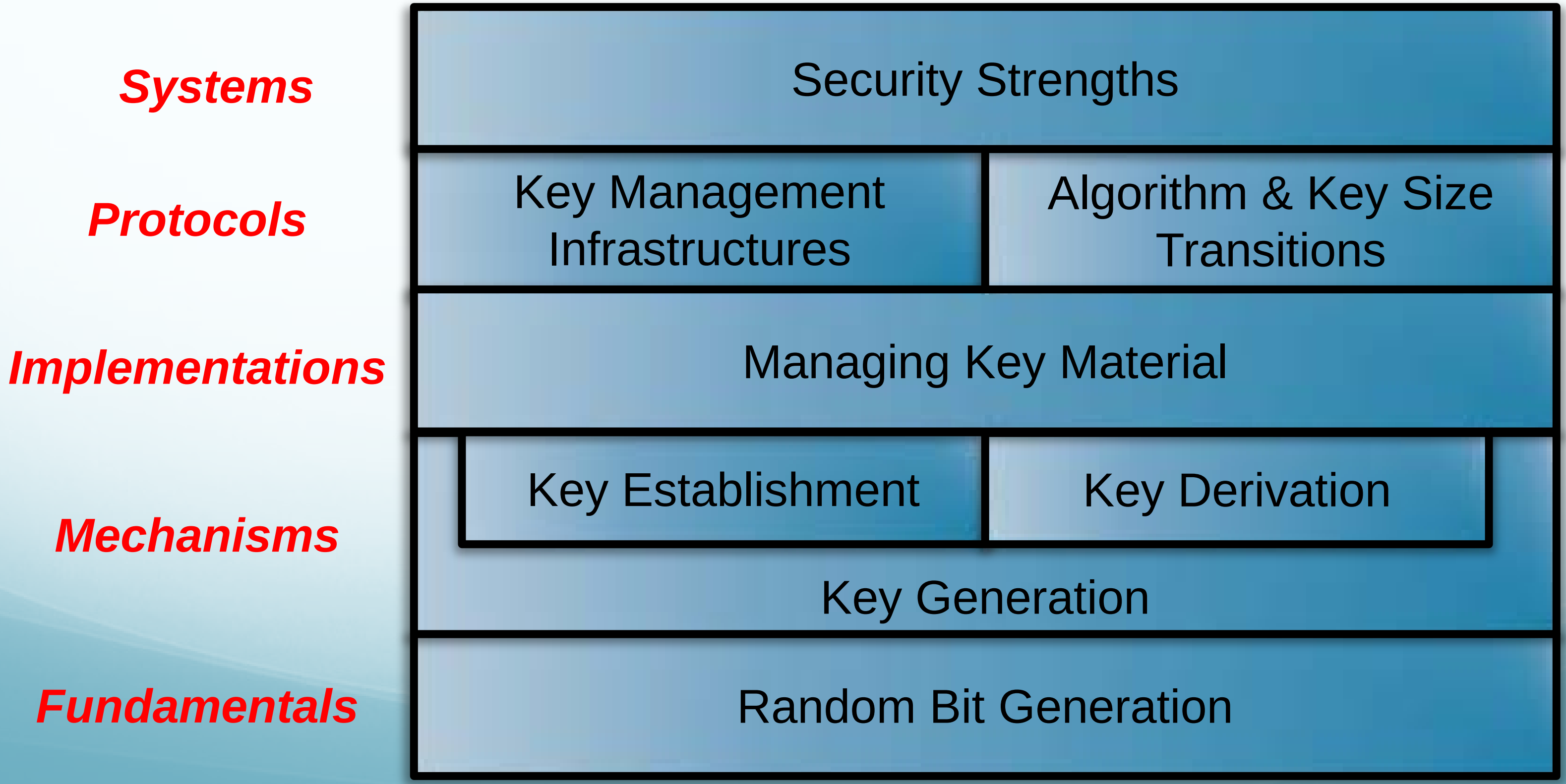
Key Management is how we make our very complex and chaotic world conform to the algorithm designers' expectations

Multiple Facets of the Key Management Problem

- Obtaining random values
- Generating strong keys
- Establishing pairwise shared secrets
- Distributing public values
- Maintaining acceptable levels of security over time
- Designing systems for acceptable strength

NIST's Key Management Standards and Guidelines…

- … are the foundational documents for solving the full range of key management problems
- NIST continues to expand this body of knowledge



Systems

Security Strengths

Protocols

Key Management Infrastructures

Algorithm & Key Size Transitions

Implementations

Managing Key Material

Mechanisms

Key Establishment

Key Derivation

Key Generation

Fundamentals

Random Bit Generation

Dual Roles for the Framework

- The Framework and our future profiles bring some order to the complex work of key management systems, allowing ckms developers and owners to more fully understand
  - What we need;
  - What we have; and
  - Want is missing.
- There is a second role that I believe is just as important – bringing the research issues into focus!

The Dirty Little Secret Is…

- There are attributes in the framework that are really important, but we can’t really field a CKMS that satisfies them today.
  - In some cases, there are no mature solutions
  - More commonly, today’s solutions don’t work as well as we thought
- The framework is an opportunity for the community to identify the game-changers
  - Problems where a (better) solution will let us leap ahead instead of just move forward

Cross-Domain Interoperability

- Key Management for a single domain is something we have a lot of experience with
- Interoperability across domains is problematic *even when the technology is explicitly designed for it*
- *Consider the Federal PKI experience: early products required horrific amounts of labor to cross-certify two CAs*
  - *The technology supported it in theory.*
  - *Products? Not so much.*

Algorithm Agility

- PKI is poster child for Algorithm Agility
  - Encodings that support any key sizes
  - Separate OIDs to identify algorithms and signatures
  - Subjects can have multiple keys to support algorithm change or algorithm negotiation
- Why was the transition from SHA-1 based digital signatures to SHA-256 based digital signatures in the Federal PKI such a problem?
  - CKMS was in reality a single algorithm system

Post Quantum Resiliency

- If we actually need to switch to McEliece or Multivariate public keys, would any current CKMS be a workable solution?
- With a signature limit (e.g., for lattices) , would any current CKMS be a workable solution?
- Could a hybrid SKI/PKI fill the gap?
- What if public key just isn’t viable? Could we build a purely symmetric key system that really met our needs?

Infrastructures for Constrained Devices

- You always assume that your adversary has no practical limitations in processing, memory, or power
- Most of our key management technologies assume the same about the legitimate players
  - Applying the tried and trusted techniques does not have the desired results
- Recent proposals for a PKI-based key management system for cars were a real eye-opener

Pseudonymity and Anonymity

- CKMSes are traditionally focused on getting the right keys to the right people, and identity is the primary tool we use to achieve that.
  - Often the goals are to preserve that information in transactions that use the CKMS

- What happens when the system goals include anonymity, pseudonymity, or unlinkability?

Scalability

- Once again, the solutions claim to provide scalability.
- X.509 was designed to support a global PKI with the help of the global X.500 directory
  - That turned out to be a lot harder than it sounded
- Kerberos is designed to for cross-domain interoperability
  - The worked examples are few and small

Questions?

- Me too!

- Let’s spend time today talking about the hard problems, and identifying opportunities…