The Keys to The Kingdom: A Comprehensive Approach to Key Management

Tim Polk, Elaine Barker, Lily Chen, Quynh Dang and John Kelsey

May 31, 2012 for ISPAB

Goals, Assumptions & Cold Realities of Cryptography

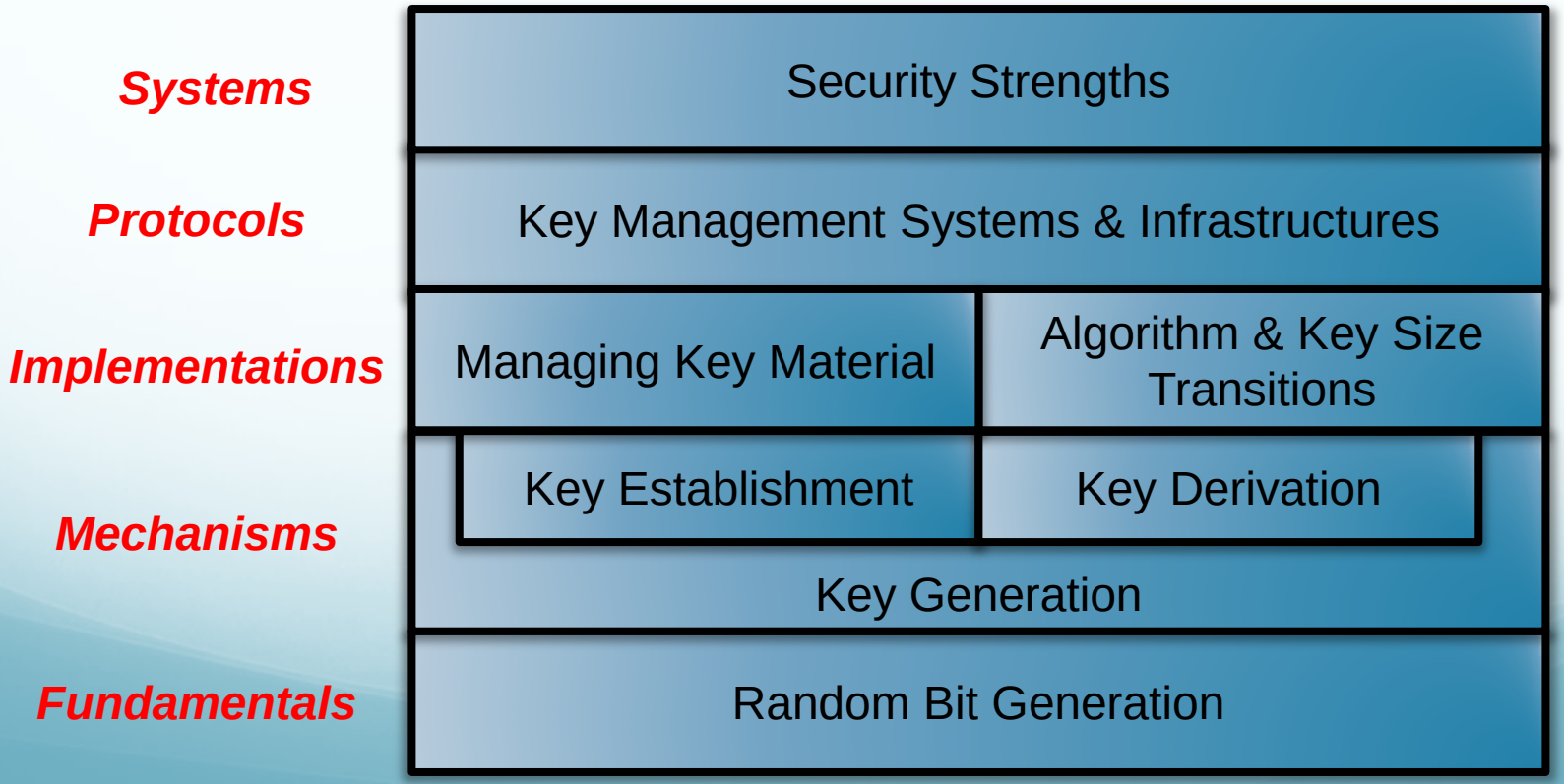
- Goals for Algorithm Designers
  - For symmetric encryption algorithms, an algorithm is secure if an attacker (given the algorithm and some ciphertext encrypted under an unknown key) cannot practically derive any information about the message (other than its length) or the key
  - For public key algorithms, 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!

Multiple Facets of the Key Management Problem

- Obtaining random values
- Generating strong keys from random values
- Establishing pairwise shared secrets to derive keys
- Distributing secret keys and 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



Obtaining Random Values

- Cryptographic key generation algorithms are deterministic
  - If you repeat the algorithm with the same set of inputs, the same “secret” value will be generated
- One of the inputs is intended to provide the entropy needed to generate random values
- When the entropy is insufficient, key generation algorithms produce weak keys

Random Bit Generation (RBG) Specifications

- Purpose: To provide approved RBGs with “tunable” security strengths (112, 128, 192, 256 bits).
  - Collaborative effort with NSA and industry.
  - Based on work conducted with ANSI X9 (X9.82)
- SP 800-90A documents deterministic algorithms for generating strings of random bits (January 2012)
- SP800-90B establishes requirements for entropy sources (draft summer 2012)
- SP 800-90C combines 90A and 90B, specifying constructions for DRBGs and NRBGs (draft summer 2012)

Key Establishment Using Public Key Cryptography

- Mechanisms that establish a shared secret value between two parties that support public key cryptography
  - Established symmetric keys will be used to protect data communications between the two parties
- Two basic flavors for key establishment: key agreement and key transport
- Based on two hard problems: Discrete log and integer factorization
- These schemes are widely used in internet security protocols, including IPsec, S/MIME and TLS

Key Derivation Functions

- Mechanisms that derive one or more secret keys from a secret value, such as a master key or a password
  - Generally based on hash functions or block ciphers
- Historically, every protocol has invented its own technique for key derivation
  - Results have been decidedly mixed!
- NIST has pursued a two-pronged approach
  - Identify robust mechanisms for use in new protocols
  - Evaluate KDFs in legacy protocols and identify acceptable options

NIST Key Management "Mechanism" Specifications

- Key Establishment Series
  - SP 800-56A – discrete log (DH, MQV key agreement)
  - SP 800-56B – factorization (RSA key transport & agreement)
  - SP 800-56C – alternative key derivation for key agreement using "extraction-then-expansion"
- Key Derivation
  - SP 800-108 – key derivation from a symmetric key
  - SP 800-132 – key derivation from a password (storage only)
  - SP 800-135 – application-specific key derivation
- SP 800-133 Cryptographic Key Generation
  - Covers the waterfront

Managing Key Material

- Even if generated appropriately, the secrecy of keys may be at risk if key lengths are too small, input values are disclosed, etc., etc.
- To ensure that cryptographer's assumptions are satisfied, the entire lifecycle for key material has to be considered, from multiple points of view:
  - General key management – security requirements for the lifecycle of a key
  - Organizational key management – policy and planning
  - Application & Infrastructures – selecting/negotiating appropriate algorithms, key lengths, & mechanisms

Managing Transitions

- Cryptographic security is often undermined in practice by a failure to manage cryptographic transitions
- Algorithms that were secure in 1980 were not necessarily secure in 2000
  - But migrating to a new algorithm was too much for some application owners to cope with!
- Some currently “popular” key lengths are now insecure (as stated in SP 800-57 Part 1)
  - But migration is still a challenge

Cryptographic Key Management Systems

- Cryptographic Key Management Systems are a collection of policies, procedures, components and devices used to protect manage and distribute cryptographic keys
- NIST is developing a Framework (draft SP 800-130) to assist CKMS designers
  - Topics to be considered with documentation requirements
- Sector−specific profiles are envisioned to provide further direction
  - Government profile currently in development
  - Smart Grid profile under consideration in SGIP

Security Strength for Systems

- NIST established the baseline for security strengths in 2005 (draft SP 800-57 Part 1), but it looked only at the algorithms and key lengths
  - Necessary, but insufficient: the security we achieve depends on so many additional parameters!
- Since 2005, we have been working to extend the security strength concept to other aspects of key management
  - E.g, the SP 800-90 RBGs have tunable strengths
- Ambitious new research project to help system designers select primitives and functions that ***in composition*** provide the desired security strength

Questions?