Cryptographic Transition Strategies

Tim Polk, NIST

October 27, 2009

Alternative Title:

Diamonds are Forever, but Cryptographic Security Is *NOT*

Overview

- Historical Perspective:
  - Why Cryptographic Transition Strategies are necessary
- Roles and Responsibilities
- Worked Example
- NIST's transition timelines
- SHA-3 Competition

Historical Perspective

- The Data Encryption Standard (DES) was published as FIPS PUB 46 in January 1976
  - Reaffirmed in 1983, 1988, 1993
- Reaffirmed in 1999, but
  - Triple DES required for all new systems
  - (single key) DES was restricted to legacy systems
- AES was published as FIPS PUB 197 in 2001
- DES withdrawn in 2005…
  - In spite of all the warnings, system owners weren't ready to transition

What Happened?

- DES got old
  - It was overcome by Moore's Law
- System engineering was short-sighted
  - Cryptographic modules implemented DES but not Triple DES or AES
  - Applications and networks were designed to support DES, rather than cryptography in general
- Unlike old age, bad system engineering is theoretically avoidable

What Happened?
And
Whose Fault Was It Anyway?

- System owners discovered far too late they had no transition path to AES or Triple DES
- It was everyone's fault
  - Cryptographic module vendors
  - Protocol designers
  - Protocol/Application developers
  - System owners
- But it was the system owners' *responsibility* to protect the data

It's Happening Again…

- 1024 bit RSA, DSA, and Diffie-Hellman are running out of steam
- SHA-1 is showing its age for digital signature applications
  - Still strong in HMACs, though
- And we still aren't ready!
  - Product developers aren't supporting bigger key sizes and new padding schemes
  - Protocols are not well-defined for emerging algorithms

Requirements for Orderly Transition

- Cryptomodule must support current algorithm(s) and future algorithm(s)
  - Possibly legacy suite as well…
- Protocols must be well-defined for all required algorithms
  - Specifications complete, Code points assigned
- Implementations must recognize and support complete set of algorithms
- System Owners must have a plan!

Example: Plan for Deploying a PKI and Smart Card based TLS Application

- Current Algorithms
  - RSA 2048 bit cryptography, PKCS#1 padding, SHA-256 hash, AES 128
- Legacy Algorithms (accept don't generate)
  - RSA 1024, PKCS#1 padding, SHA-1 hash, Triple DES
- Future Algorithms (accept by 2012, generate after 2012)
  - RSA 2048, PSS padding, SHA-256, AES 128
  - ECC curve P-256, SHA-256, ECDH, AES 128

TLS/PKI Example: Implications for cryptomodules

- Server modules must support complete algorithm suite
- Smart cards must support RSA 2048 bit crypto
  - After 2012, may support RSA 2048 or ECC P-256
- Desktop modules must support RSA 1024 and 2048, PKCS#1 padding, SHA-1 and SHA-256, Triple DES and AES 128
  - ECDH by 2012, RSA with PSS padding by 2012

TLS/PKI Example: Implications for Protocol Developers

- PKI standards must be well-defined for RSA and ECC keys and signatures (inc padding)
- TLS specifications must be well-defined and code points assigned for RSA and ECC suites with Triple DES and AES

TLS/PKI Example: Implications for Product Developers

- PKI clients must recognize and validate RSA 1024 and 2048 keys and signatures, PKCS#1 padding, with both SHA-1 and SHA-256
  - By 2012, must support ECC and RSA PSS
- TLS server and client software must support RSA/AES cipher suites with both SHA-1 and SHA-256
  - By 2012, must support ECC/AES cipher suites

TLS/PKI Example: Summary

- Cryptographic transitions are complicated
- Getting all the pieces in place for new algorithms takes about a decade

- Develop a strategy early, ask your vendor lots of questions, and stay with it at every step!

NIST's Transition Timelines

- Goal: Adequate Cryptographic Protection for the Lifetime of the Data
- Phase out 1024 bit RSA/DSA/DH, SHA-1 in digital signatures, and 2 key Triple DES by the the end of 2010
  - If you don't have a plan in place, you may be too late already!

SHA-3 Competition

- Prompted by collision attacks on a number of hash functions including NIST SHA-1 standard
  - SHA-2s not yet affected, but are in same family as broken algorithms
    - SHA-2 design rationale never fully explained
- NIST competition for new “SHA-3” hash family
  - Plug replacement for SHA-2:
    - 224, 256, 384 and 512-bit hash algorithms
    - “on-line:” process messages in small pieces
  - 51 submissions received Oct. 2008
    - Heavy international participation
  - Now down to 14 second round candidates
  - Next SHA-3 Candidate Conference Aug. 2010 after Crypto 2010
  - Expect to pick winner by 1012

Cryptographic Algorithms are Vulnerable…

- To Moore's Law
  - All cryptographic algorithms can be solved by brute force (trying every key); it is just a question of resources
- To cryptanalysis
  - Some cryptographic algorithms can be solved more quickly using analytic attacks

- Have a plan and be ready to transition to stronger algorithms!

Resources

- http://csrc.nist.gov
  - NIST Special Publication 800-57 Part 3: Application-Specific Key Management Guidance
    - Final publication by November 1
  - The Transitioning of Cryptographic Algorithms and Key Sizes (Draft White Paper)

Questions?