Practical Key Management

by Benjamin Tomhave, MS, CISSP

CIScon 2009 Post-Session

A Few Requests

- Cell phones: silent/vibrate + calls outside
- Breaks (yes, please!)
- Please ask lots of questions!!

Instructor Background

- ~15 years security experience
- A mile wide, a mile deep
- MS InfoSec Mgmt (GWU in DC)
- Today: TechDir Security & Compliance
- Risk, Architecture, Compliance, Solutions

Course Objectives

- Baseline definitions
- Deep-discuss key management lifecycle
- Overview of emerging standards
- Overview of solutions/vendors

Topic Overview

- What is encryption?
- What are the types of encryption?
- Why is “key management” important?

Go-Forward Agenda

- The Key Management Lifecycle
- Emerging Standards
- Solution Landscape
- Use Cases & Architectures
- References

The Key Management Lifecycle

- We all know lifecycle…
- Why do we need one?
- What all could be involved anyway?

NIST SP 800-57

- Part 1 – General – 142 pages
  - Covers most aspects of key mgmt
  - Includes key configuration details
- Part 2 – Best Practices – 79 pages
  - Goes into security best practices
  - Planning, policies, documentation, etc.

NIST SP 800-57 (p2)

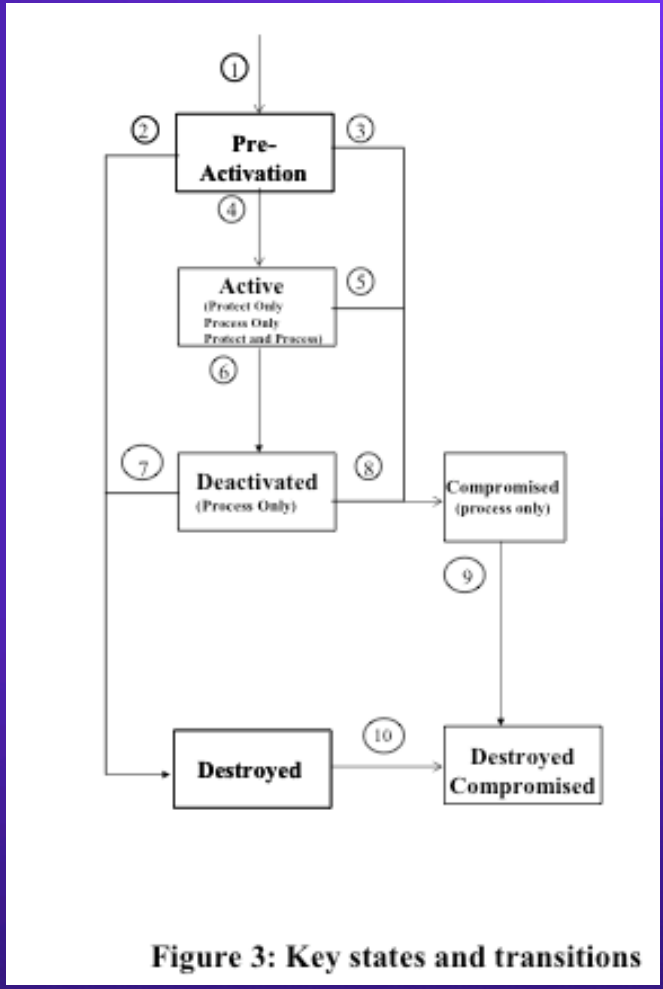


Figure 3: Key states and transitions

NIST SP 800-57 (p3)

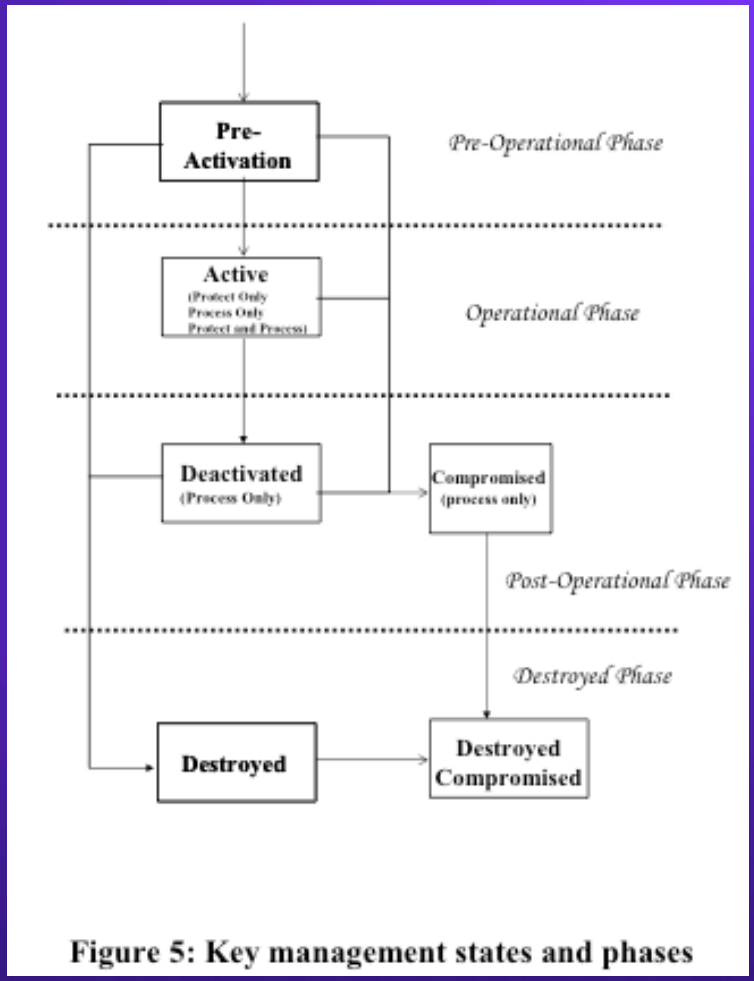
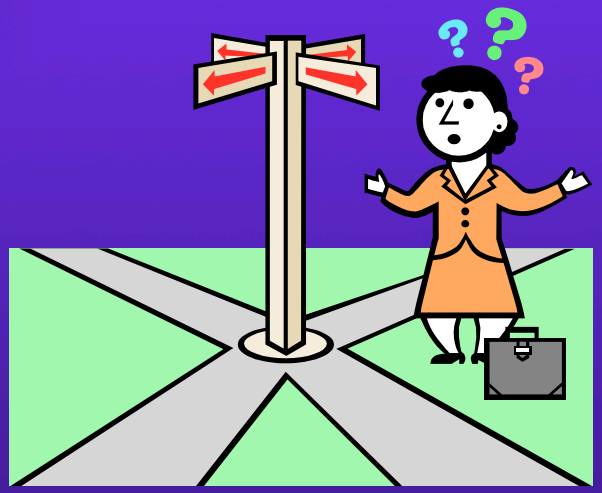


Figure 5: Key management states and phases

NIST SP 800-57 (p4)

- To recap…
  - 6 States
  - 10 Transitions
  - 4 Phases



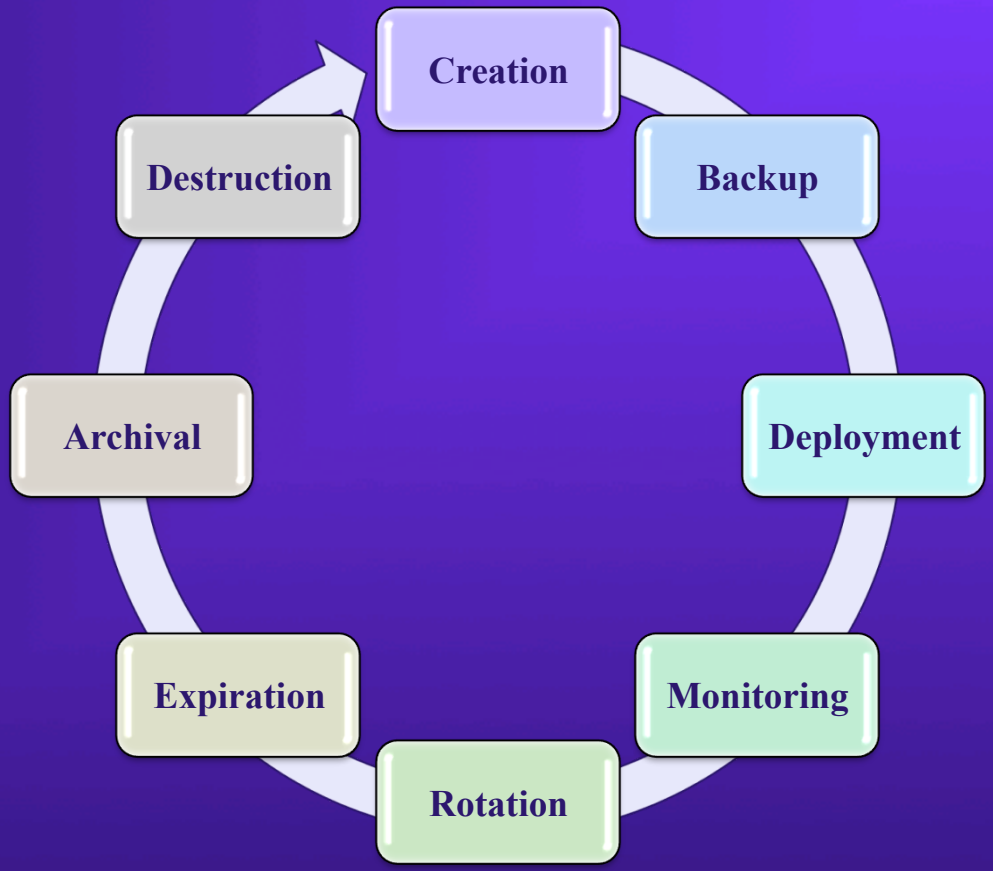
NIST SP 800-57 (p5)

- Conclusions
  - Excellent details for reference
  - Somewhat complicated
  - Leaning toward confusing
  - Too much information in one place

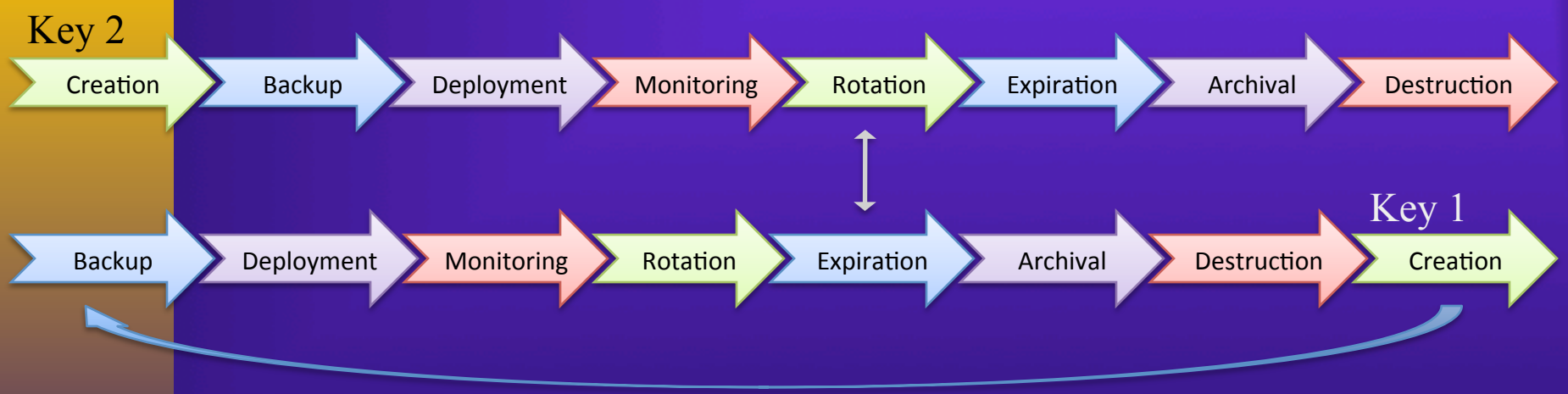
A Procedural View of KM

- 8 Procedural Steps (or Phases)
- Simple Transitions
- Maps (some) Requirements to 800-57

Procedural Lifecycle



Procedural Transitions



Procedures: Creation (p1)

- Generate the key in a secured environment
  - Use known good libraries, algorithms
  - Good random number generator
  - Good key strength (bits / size)

"If they need to keep the algorithm secret, then it probably can't withstand scrutiny and is thus a bad technique."

NIST SP 800-57 Cryptoperiod

Key Type	Cryptoperiod	
	Originator Usage Period (OUP)	Recipient Usage Period
12. Symmetric Key Agreement Key	1-2 years	
13. Private Static Key Agreement Key	1-2 years[15]	
14. Public Static Key Agreement Key	1-2 years	
15. Private Ephemeral Key Agreement Key	One key agreement transaction	
16. Public Ephemeral Key Agreement Key	One key agreement transaction	
17. Symmetric Authorization Key	≤ 2 years	
18. Private Authorization Key	≤ 2 years	
19. Public Authorization Key	≤ 2 years	

Procedures: Creation (p2)

- Protect the key
  - Split keys, procedures, or similar
  - Use asymmetric encryption
  - Enforce separation of duties
  - May split key gen box from key use box

Procedures: Backup

- Write to external media, store securely
  - Might use existing backup methods
- Encrypt the backup
- Apply consistent disaster recovery plans

"Brief-in the recovery team to help set and manage backup and recovery requirements and processes."

Procedures: Deployment

- Again, encryption recommended
  - Separation of duties may be required
  - Thorough documentation of workflows
- Objective: install the new key
  - Do not activate it
  - Do not remove/decommission old keys

Procedures: Monitoring

- Technically ongoing
- Three key aspects to monitor:
  - Unauthorized admin access
  - Crypto system performance
  - The key itself (integrity, availability)
- Goal: minimize or avoid downtime!

Procedures: Rotation

- Not the same as Deployment
- Turning new key “on” in production
- May include converting data to new key
- Overlaps with Expiration of old key

“Don't remove an old key from production until it can be proven that no data in production is still encrypted with it.”

Procedures: Expiration

- Depends in part on cryptoperiod
- Typically expire keys annually now
- Turn old key “off” in production
  - Key still available for decrypt operations
- Kind of like retirement…

Procedures: Archival (p1)

- Similar to Backup
- This is ***not*** deleting the key!
- Archival period based on life of data encrypted with the archived key.
- Document, document, document.

Procedures: Archival (p2)

- Three tips:
  - Document and index the key and its associated data.
  - Ensure that the archived copy of the key has itself been secured.
  - Include recovery of encrypted data using archived keys as part of your routing BCP/DR testing procedures.

Description of Procedures

- End of life for a key.
- Follow secure deletion practices.
- Be very, very, very, (…), very careful.

"Don't destroy a key and all its copies until you are absolutely, positively sure that you will never need it again."

Emerging Standards

- OASIS KMIP
- OASIS EKMI
- IEEE P1619.3
- IETF KEYPROV
- Others…

Solution Landscape

- StrongKey / StrongAuth Inc.
  - http://www.strongkey.org/
  - Open-source reference platform for OASIS EKMI
  - Active deployments, supported by Strong Auth, Inc.

Solution Landscape (cont'd)

- The Thales Group (acquired nCipher)
  - http://iss.thalesgroup.com/
  - wide span of solutions, grew out of HSMs

Solution Landscape (cont'd)

- Venafi
  - http://www.venafi.com/
  - Primarily focused on asymmetric encryption
  - Essentially an abstraction layer beyond direct key management

Solution Landscape (cont'd)

- RSA Security (Acquired by EMC)
  - http://www.rsa.com/node.aspx?id=1203
  - Full spectrum of solutions, including coding libraries
  - Generally software-oriented solutions
  - Parts of BSAFE library now free through RSA Share Project

Solution Landscape (cont'd)

- SafeNet (Acquired Ingrian Networks)
  - http://www.safenet-inc.com/
  - Broad spectrum of encryption and key management solutions
  - Can work at the application or database level
  - Blend of hardware and software solutions

Solution Landscape (cont'd)

- Valicore Technologies, Inc.
  - http://www.valicore.com/
  - Custom encryption and key management solutions
  - Most work-for-hire has supported cable boxes

Solution Landscape (cont'd)

- nuBridges
  - http://www.nubridges.com/
  - Key management and tokenization solutions
  - Current focus appears to be more on tokenization (too bad)

Solution Landscape (cont'd)

- KeyMG
  - http://xircles.codehaus.org/projects/keymg
  - To be an open-source Java implementation of OASIS EKMI draft standard

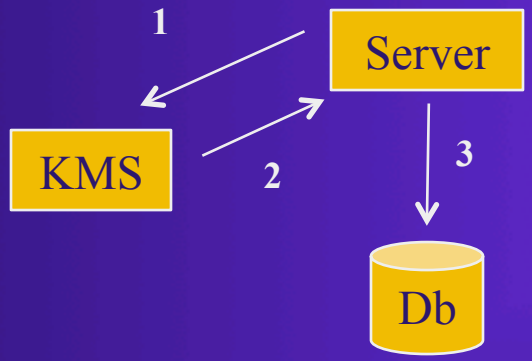
Use Cases & Architectures

- Use Cases:
  - Inline/Transparent Encryption
  - Transactional-Appliance Encryption
  - Transaction-Application Encryption
  - Disk & Volume Encryption

Use Cases & Architectures

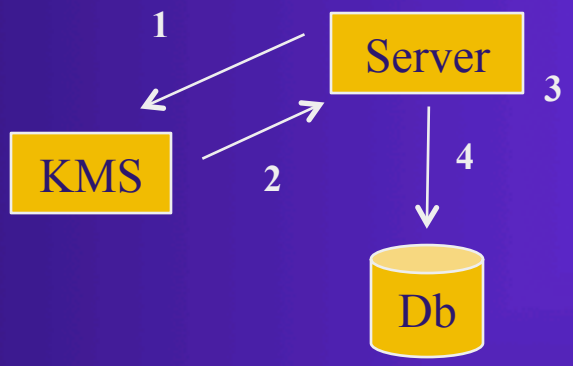
- Different Architectures
  - Transactional-Appliance
  - Transactional-Application
  - Inline-Storage
  - Transparent-Database
  - Disk Encryption

Transactional-Appliance



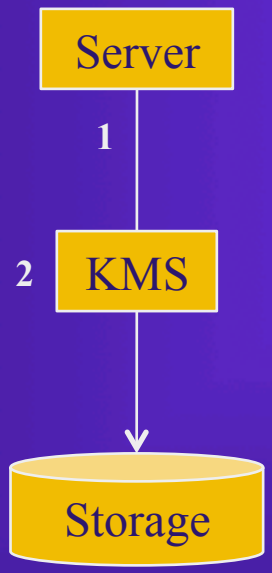
1. Server sends cleartext blob to KMS over a secure connection.
2. KMS responds with ciphertext blob.
3. Server writes ciphertext blob to Db.

Transactional-Application



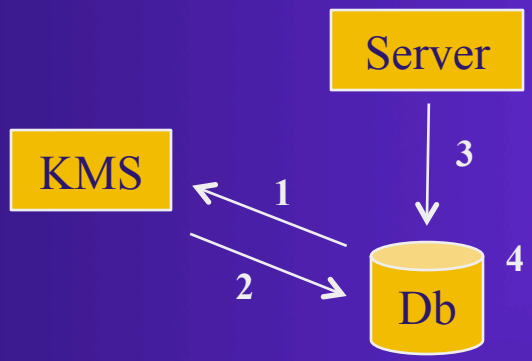
1. Server requests key from KMS over a secure connection.
2. KMS provides the key to the Server over a secure connection.
3. Server encrypts blob with key (and likely caches the key).
4. Server writes ciphertext blob to Db.

Inline-Storage



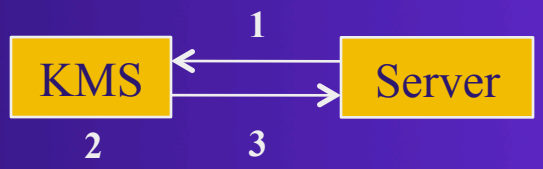
1. Server writes data blob to Storage.
2. KMS transparently encrypts the data inline without the knowledge of the Server or Storage

Transparent-Database



1. Db requests a key from the KMS over a secure connection.
2. KMS provides the key to the Db over a secure connection.
3. Server writes cleartext blob to Db.
4. Db transparently encrypts the blob before committing the write.

Disk Encryption



1. Server requests a key from KMS over a secure connection, possibly during disk encryption configuration.
2. KMS generates key, possibly holding a copy in escrow.
3. KMS provides the key to the Server over a secure connection.

References

- "The Key Management Lifecycle" http://www.secureconsulting.net/2008/03/the_key_management_lifecycle.html

  or: http://bit.ly/1RSFLE

References

- NIST SP 800-57 "Recommendation for Key Management"
  - Parts 1 & 2 + Part 3 (DRAFT)
  - http://csrc.nist.gov/publications/PubsSPs.html
  - or: http://bit.ly/2V436R

Benjamin Tomhave, MS, CISSP
http://www.secureconsulting.net/
tomhave@secureconsulting.net
http://twitter.com/falconsview

THANK YOU!