MAINFRAME CRYPTO

Unscrambling the Complexity of Crypto!

Key Rotation – Why? Which? When? How? Who?

Greg Boyd
(gregboyd@mainframecrypto.com)

October 2020

Copyrights and Trademarks

- Copyright © 2020 Greg Boyd, Mainframe Crypto, LLC. All rights reserved.
- Presentation based on material copyrighted by IBM, and developed by myself, as well as many others that I worked with over the past 30+ years

- All trademarks, trade names, service marks and logos referenced herein belong to their respective companies. IBM, System z, zEnterprise and z/OS are trademarks of International Business Machines Corporation in the United States, other countries, or both. All trademarks, trade names, service marks and logos referenced herein belong to their respective companies.
- **THIS PRESENTATION IS FOR YOUR INFORMATIONAL PURPOSES ONLY**. Greg Boyd and Mainframe Crypto, LLC assumes no responsibility for the accuracy or completeness of the information. TO THE EXTENT PERMITTED BY APPLICABLE LAW, THIS DOCUMENT IS PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND, INCLUDING, WITHOUT LIMITATION, ANY IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, OR NONINFRINGEMENT. In no event will Greg Boyd or Mainframe Crypto, LLC be liable for any loss or damage, direct or indirect, in connection with this presentation, including, without limitation, lost profits, lost investment, business interruption, goodwill, or lost data, even if expressly advised in advance of the possibility of such damages.

Agenda – Key Rotation

- Why?
- When?
- Which?
- Who?
- How?

Why rotate keys?

- Because the standards say so

- Limit the amount of data (encrypted by a key)
- Key leakage

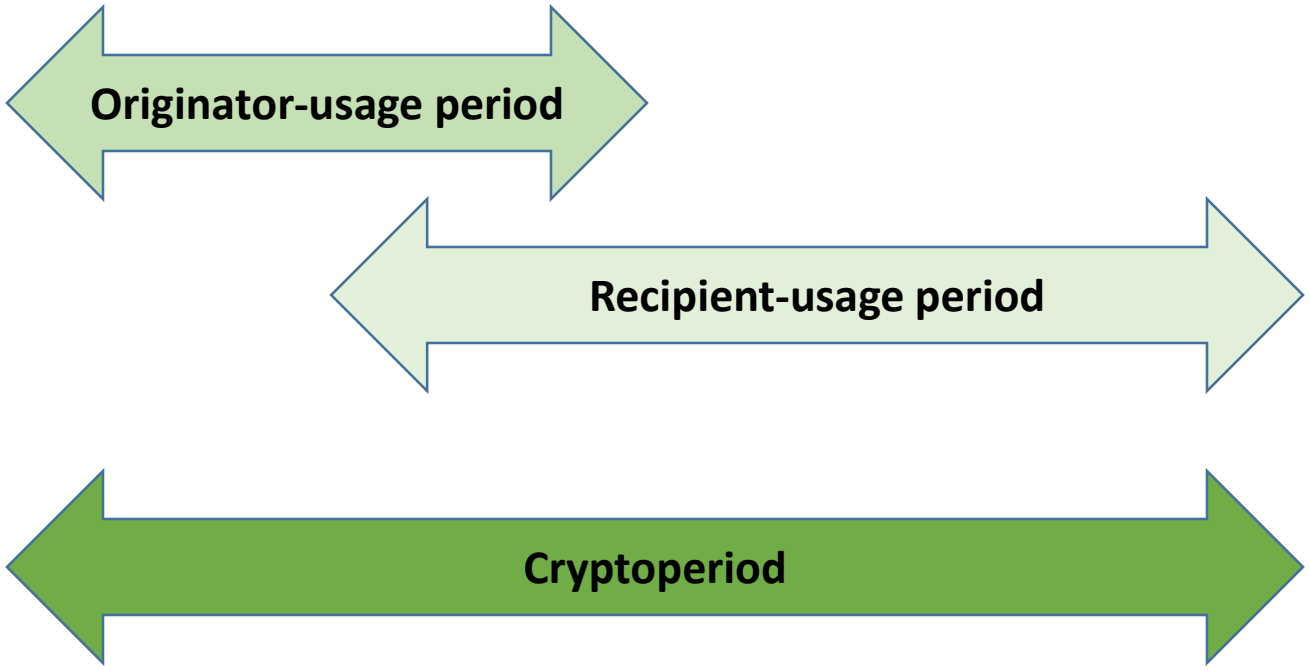
- Changing standards – DES being deprecated

PCI DSS 3.2.1

- **Requirement**
  - **3.6.4 Cryptographic key changes for keys that have reached the end of their cryptoperiod** (for example, after a defined period of time has passed and/or after a certain amount of cipher-text has been produced by a given key), as defined by the associated application vendor or key owner, and based on industry best practices and guidelines (for example, NIST Special Publication 800-57).
- Guidance
  - **A cryptoperiod is the time span during which a particular cryptographic key can be used for its defined purpose.** Considerations for defining the cryptoperiod include, but are not limited to, the strength of the underlying algorithm, size or length of the key, risk of key compromise, and the sensitivity of the data being encrypted.
  - Periodic changing of encryption keys when the keys have reached the end of their cryptoperiod is imperative to minimize the risk of someone's obtaining the encryption keys, and using them to decrypt data.

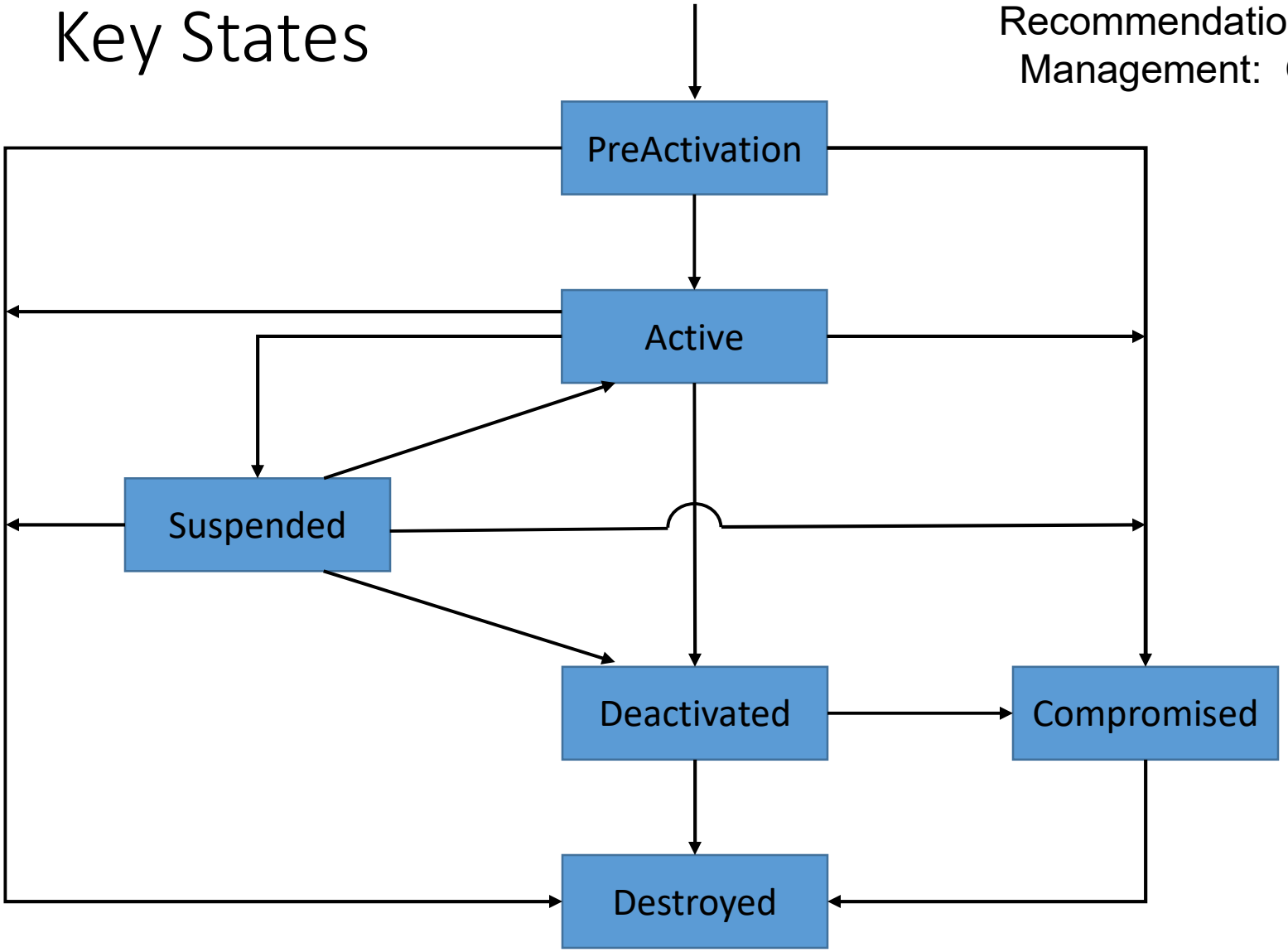
When to rotate keys?

- Whenever your security policy says …



Key States

From SP800-57 Part 1 Recommendation for Key Management: General



Cryptoperiod - Symmetric

Key Type	Originator-Usage Period (OUP)	Recipient-Usage Period
Symmetric Authentication	<=2 years	<=OUP + 3 years
Symmetric Data Encryption	<=2 years	<=OUP + 3 years
Symmetric Key Wrapping	<=2 years	<=OUP + 3 years
Symmetric RBG	See SP800-90	--
Symmetric Master	About 1 year	--
Symmetric Key Agreement	1 to 2 years	

Table 1, Suggested cryptoperiods for key types

Recommendation for Key Management Part 1: General

NIST SP800-57 Part 1 Release 4

Cryptoperiod - Asymmetric

Key Type	Originator-Usage Period (OUP)	Recipient-Usage Period
Private Signature	1 to 3 years	--
Public Signature-Verification	Several years (depends on key size)	
Private Authentication	1 to 2 years	--
Public Authentication	1 to 2 years	--
Private Key Transport	<=2 years	
Public Key Transport	1 to 2 years	
Symmetric Key Agreement	1 to 2 years	
Private Static Key Agreement	1 to 2 years	
Public Static Key Agreement	1 to 2 years	
Private Ephemeral Key Agreement	One key-agreement transaction	
Public Ephemeral key Agreement	One key-agreement transaction	

Risks that affect Cryptoperiod

- Strength of the crypto mechanism (algorithm, key length, block length, mode)
- Security of the crypto module (FIPS 140 Level 4) vs software
- Operating environment (secure facility vs open office environment vs publicly accessible terminal)
- Volume of information (number of bytes or transactions)
- Lifecycle of the data
- Security function (data encryption, digital signature, key protection)
- Rekeying method (human intervention, vs PKI vs key management system)
- Key update or key-derivation process
- Number of nodes that share the key
- Number of copies of the key and the distribution process
- Personnel turnover
- Value of the data to attackers
- Threat to the data from new, disruptive technologies

from SP800-57 Part 1 Revision 4

Other Factors (related to cryptoperiod)

- Operational Impact
  - Cost of an outage
    - Pervasive Encryption
      - DB2 - Supports dynamic key rotation
      - Non-DB2 – At least a brief outage is required
    - Guardium Tool, Pervasive Encryption (non-DB2)
      - An outage is required
- DoS Risk
  - How complicated is the key change process? (How long does it take?)
  - What is the risk if there is a problem?

Which keys?

- All keys … but the cryptoperiod will be different
  - Symmetric Keys
  - Signing Keys
  - Key Management Keys

- Only master keys?
  - No, a master key is just a data key, where the encrypted data is … other keys

Who? And How?

- Depends on the implementation
  - Pervasive Encryption
    - Owner?
    - Storage Admin?
  - Guardium Infosphere
    - DBA
  - Application Encryption
    - Application coder
  - Digital Certificates

Data Set Encryption - Reencipher (1 of 5)

PROD.CUSTDATA.**
ACCESS LIST
DFP Segment
KEYLABEL(PRODCUST.KEYX)

Ciphertext file, protected by PRODCUST.KEY1

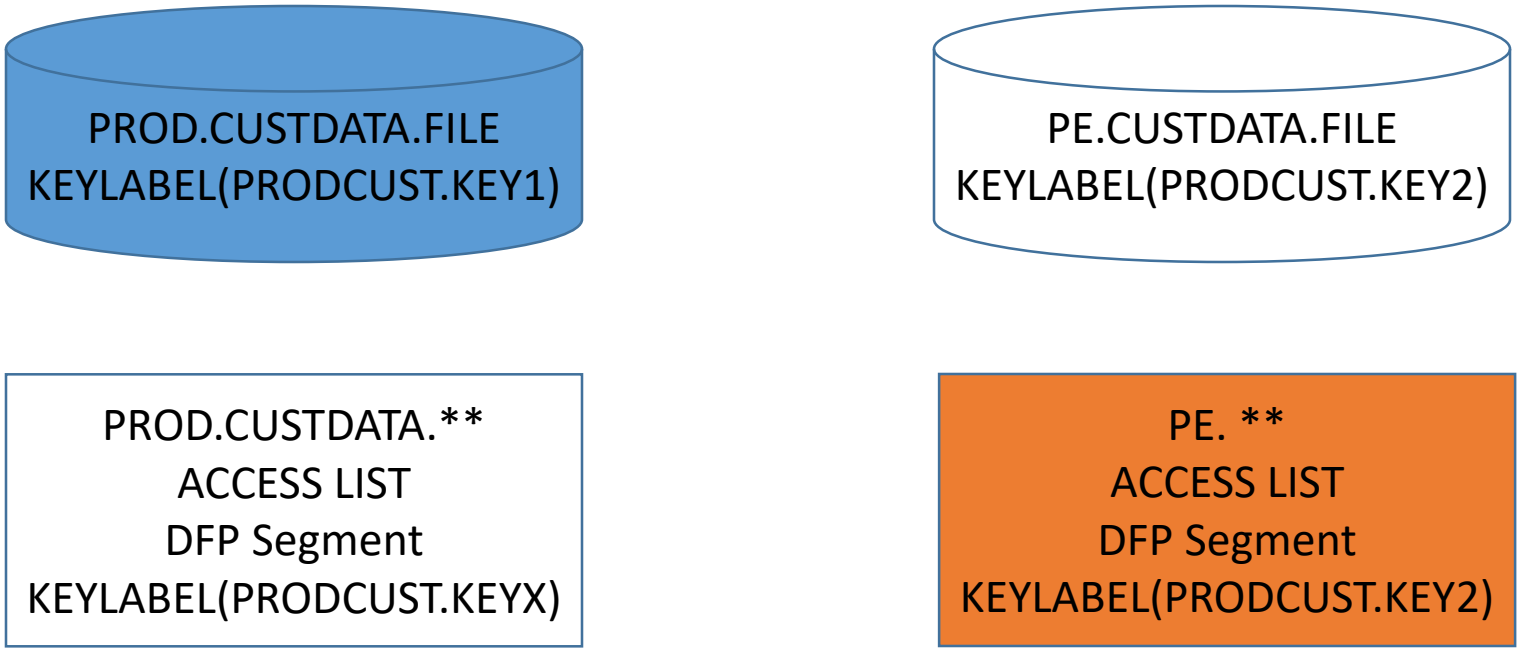
Data Set Encryption - Reencipher (2 of 5)

PROD.CUSTDATA.**
ACCESS LIST
DFP Segment
KEYLABEL(PRODCUST.KEYX)

PE. **
ACCESS LIST
DFP Segment
KEYLABEL(PRODCUST.KEY2)

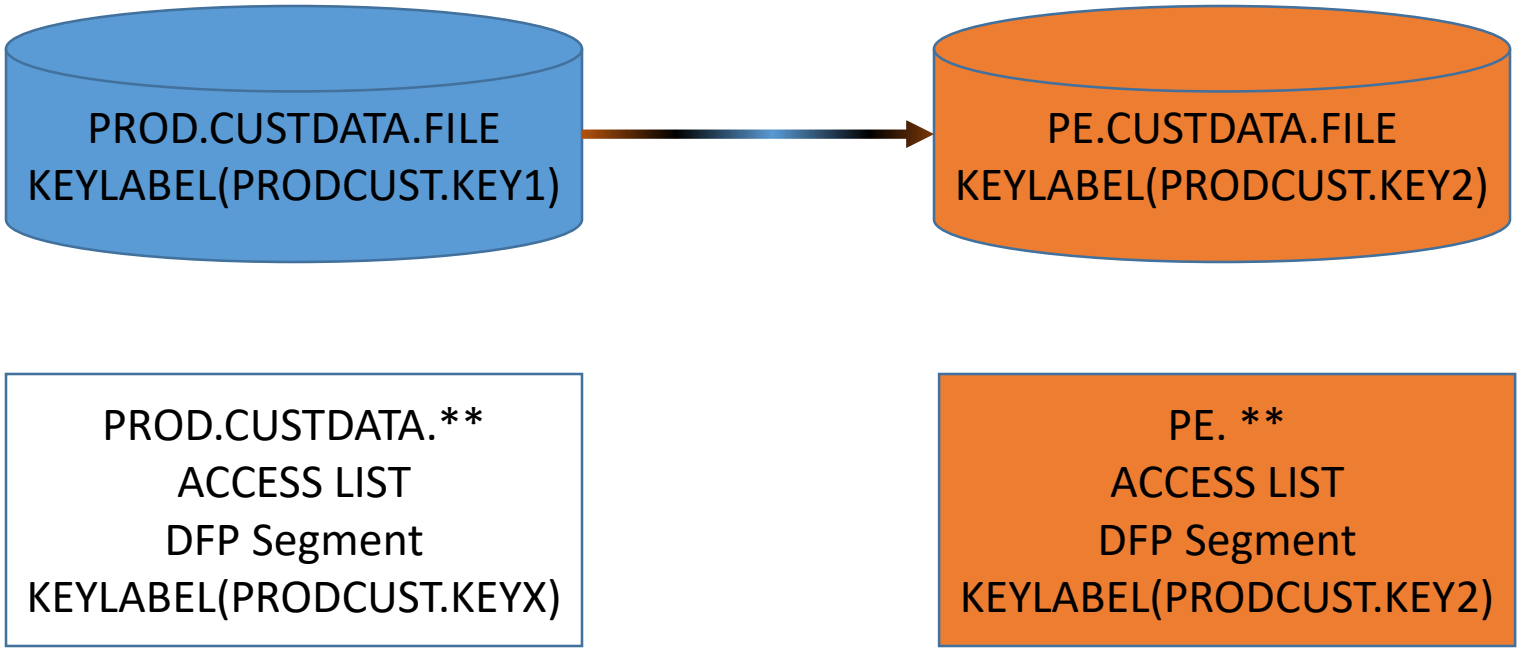
Create a new data set profile, referencing the new key label, PRODCUST.KEY2

Data Set Encryption - Reencipher (3 of 5)



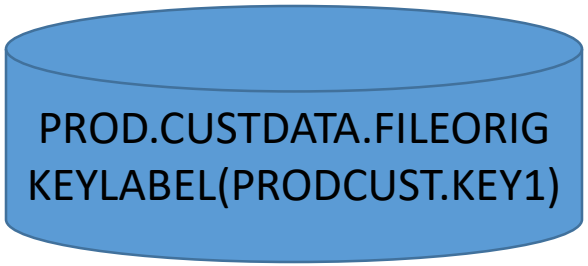
Allocate

Data Set Encryption - Reencipher (4 of 5)



Copy the file

Data Set Encryption - Reencipher (5 of 5)

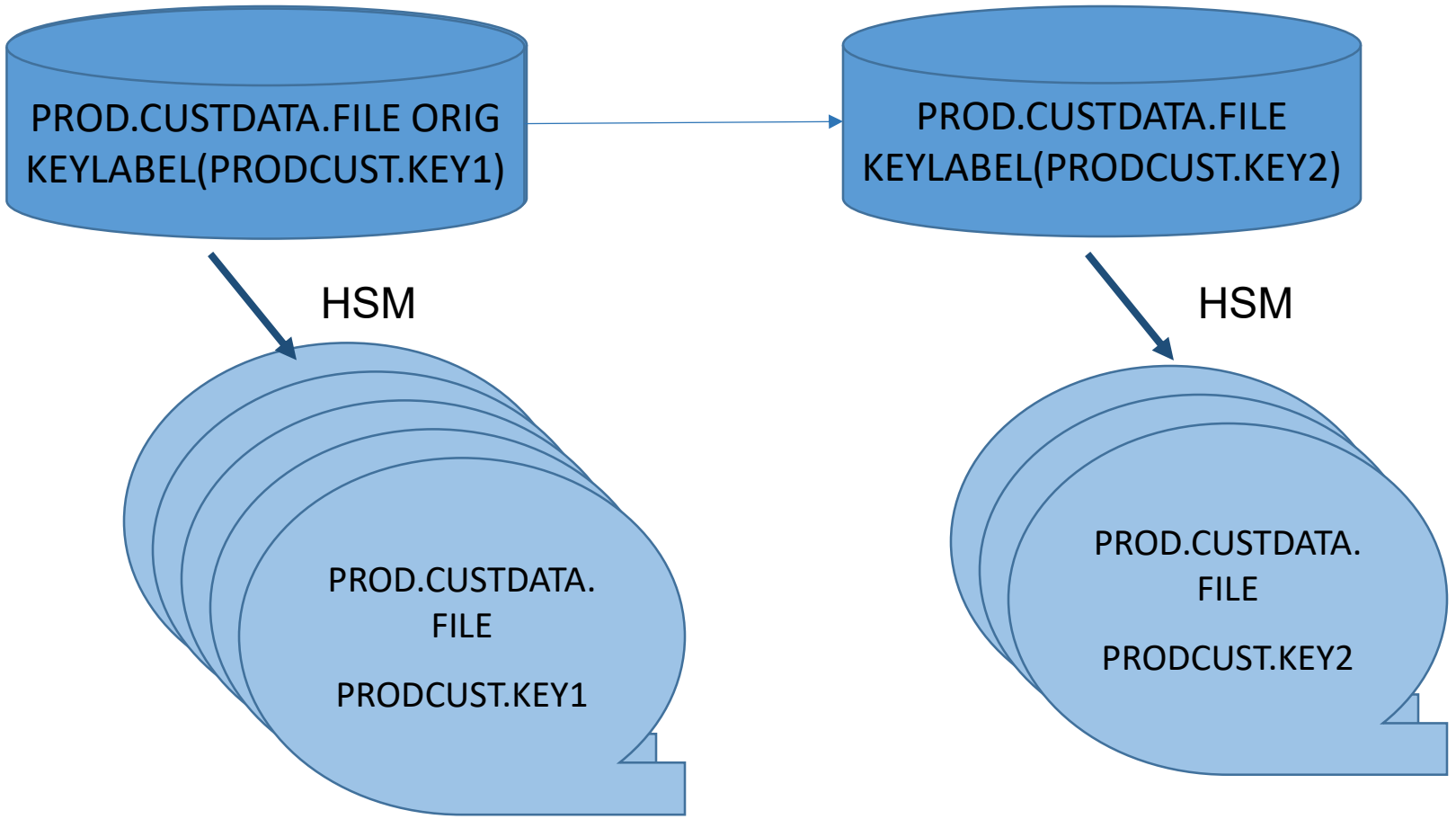


PROD.CUSTDATA.**
ACCESS LIST
DFP Segment
KEYLABEL(PRODCUST.KEYX)

PE. **
ACCESS LIST
DFP Segment
KEYLABEL(PRODCUST.KEY2)

Rename the two files

Data Set Encryption - Archives



zDMF

- IBM z/OS Dataset Mobility Facility V3.4
  - Announcement letter 218-533, Oct. 23, 2018
  - Support for encryption of extended format data sets during migration
- December 2018
  - zDMF encrypts EF data sets while in use
- 2Q2019
  - zDMF converts basic/large format data sets to EF data sets and encrypts at the same time
  - zDMF will reencipher the data set
- Future
  - Support additional data set types
  - Dynamic data set compression option

Changing Keys - Database Encryption Tool

- Option #1 – New Key (New label in the CKDS)
  - Unload
  - Change EDITPROC/FIELDPROC to reference new key
  - Drop/Recreate the table
  - Reload
- Option #2 – Change the key value (Not the label)
  - Unload
  - Change current key
  - DB2 restart
  - Reload

*Can only change the key material, not the type of key

When to rotate keys

- Routine
  - Whenever your standards call for key rotation
- Non-routine
  - In case of emergency

Key Management Policies/Procedures

- Master Keys
  - How Often is the DES-MK changed? The AES-MK? The RSA-MK? The ECC-MK? The P11-MK?
  - How Many Master Key Officers/Master Key Parts?
  - Where are the Key Parts Stored for Emergencies?
- Operational Keys
  - Who decides key types? Algorithm; Secure/Protected vs Clear
  - How Often are Application Keys changed?
  - Key retention policies
- Master Key Officers = Operational Key Officers?

Standards References

- NIST SP 800-131A Revision 2 Transitioning the Use of Cryptographic Algorithms and Key Lengths
  - https://doi.org/10.6028/NIST.SP.800-131Ar2
- NIST SP 800-90A Rev. 1 Recommendations for Random Number Generation Using Deterministic Random Bit Generators
  - https://csrc.nist.gov/publications/detail/sp/800-90a/rev-1/final
- PCI DSS 3.2.1
  - https://www.pcisecuritystandards.org/document_library

zDMF References

- zDMF Announcements
  - IBM z/OS Data Set Mobility Facility V3.4.1 data migration software enables heterogeneous storage migrations with continuous transparent access (219-473)
    - https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?infotype=an&subtype=ca&appname=gpateam&supplier=897&letternum=ENUS219-473
  - Preview: IBM z/OS Dataset Mobility Facility (zDMF) V3.4 will position zDMF to deliver new functions in the service stream (218-533)
    - https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=897/ENUS218-533&infotype=AN&subtype=CA
- zDMF Session at Share Fort Worth
  - 'Data Set Level Encryption in 4 Easy Steps with zDMF! And That's Not All' Session #26236, Rebecca Levesque

Questions?