MAINFRAME CRYPTO

Unscrambling the Complexity of Crypto!

Database Encryption

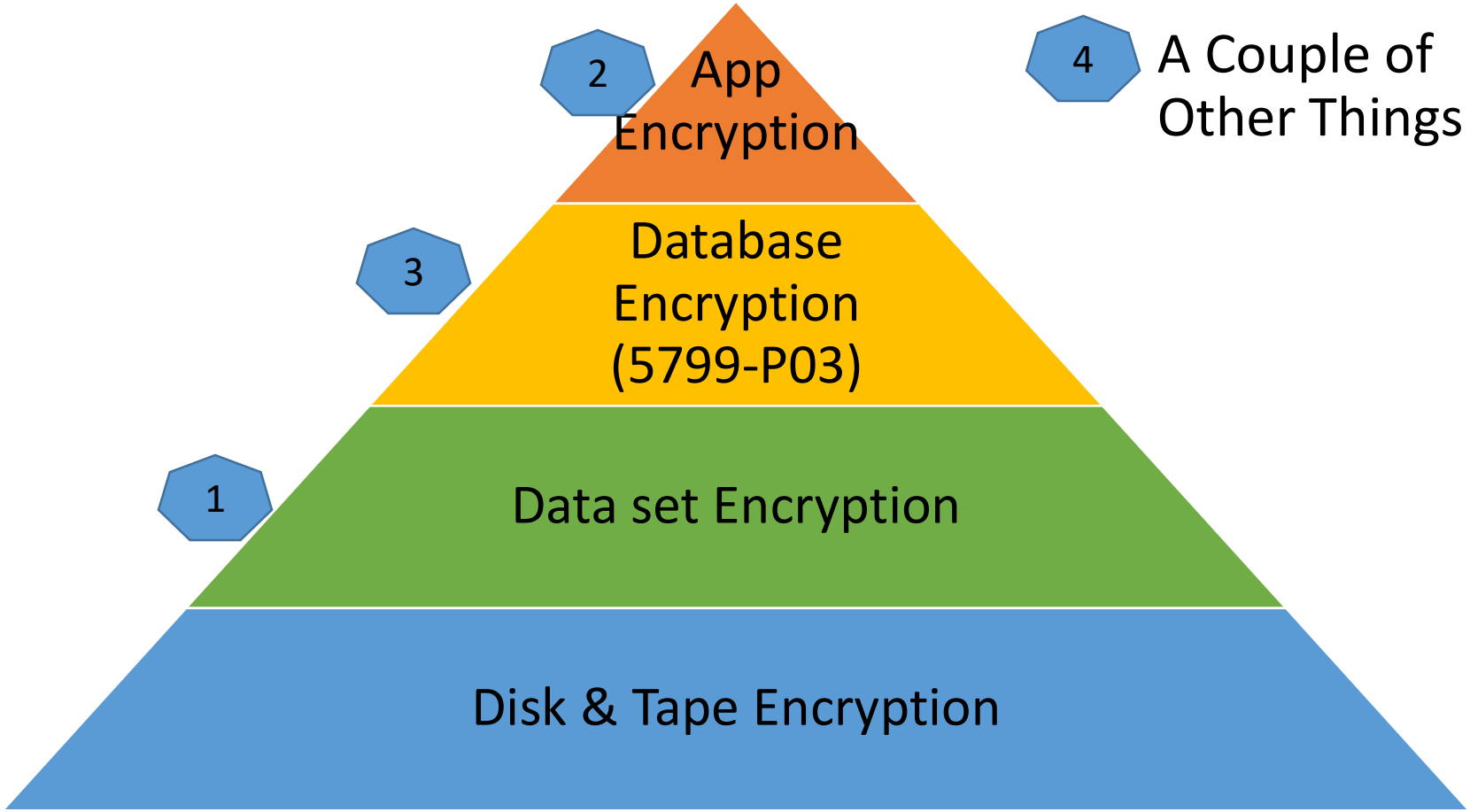
Greg Boyd
(gregboyd@mainframecrypto.com)

THE Z EXCHANGE

August 2021

Copyrights & Trademarks

Agenda – Database Encryption



DB2 Support of z/OS Data Set Encryption

- Relies on DFSMS data set encryption support – Extended Format
- Transparent encryption of data at rest without downtime or application changes
  - Encrypt user table spaces
  - Active and archive log datasets
  - Encrypt catalog and directory table spaces

Assigning the key label for DB2 system objects

- Security Admin assigns key label in the DFP segment of the SAF data set profile
- Database System Admin assigns key label using ENCRYPTION_KEYLABEL parameter (V12R1M502 only)
  - -SET SYSPARM command
  - Impacts all members of the data sharing group
  - Requires SYSADM or SECADM authority
  - DB2 DBM1 and MSTR address spaces must have SAF authority to the key label
- DBA or Storage Admin on the IDCAMS DEFINE
- Storage admin assigns key label in the DFSMS data class

Assigning key label for DB2 user objects (requires V12R1M502)

- Database Admin assigns key label using SQL
  - Includes explicitly or implicitly created base table space, auxiliary table spaces, XML table spaces and index spaces
  - Only for tables in a universal table space or a partitioned table space
  - CREATE/ALTER STOGROUP – assigns key label at the storage group level to encrypt all table spaces using the storage group
  - CREATE/ALTER TABLE – assigns key label at the table level, to encrypt all table spaces associated with the table
  - Enabled via APPLCOMPAT V12R1M502

Encrypting DB2 System Objects

- Catalog and directory table spaces
  - Execute REORG TABLESPACE utility to encrypt DSNDB06 and DSNDB01
  - Encrypt DSNDB01.SYSUTILIX – execute RECOVER utility followed by REBUILD INDEX ALL
- Active Logs
  - Encrypt new active logs
    - Define the active log data set as encrypted and issue SET LOG with NEWLOG option
  - Encrypt all active logs
    - Stop DB2 and copy the contents of the active log data set to an encrypted data set
    - Restart DB2
- Archive Logs
  - Automatically encrypted based on the key label setting

Encrypting User Objects (requires V12R1M502)

- Execute REORG utility to encrypt existing table spaces
- New tables spaces / partitions as they are populated

Utilities

- All IBM online utilities support table spaces and indices where the underlying VSAM data set is encrypted
- Online utilities
  - The userid invoking the utility will need SAF authority to the key label
- Stand alone utilities
  - The batch job executing the utility will need SAF authority to the key label

Compression

- DB2 Compression works seamlessly!

Futures

- Buffers encrypted?

DB2 Performance Implications

- Table row size has no impact on encryption/decryption
- 32K Page Size does cost more than a 4K page, but it is less than 8 times
- IBM internal benchmarks show that a z14 uses up to 7x less CPU than a z13
  - Your mileage **WILL** vary

DB2 Built-In Functions (Old)

- Under application control – you encrypt the fields that need to be secure
  - 'Password for Encryption' is hashed (using MD5) to generate a unique key
  - Hint can be used as a prompt for remembering the key
  - Encrypted field must be defined as VARCHAR (since it will contain binary data once its encrypted)
  - The encrypted field will be longer (next multiple of 8 bytes + 24 bytes of MetaData + 32 bytes for optional hint field)
  - TDES Only!

```
Encrypt_TDES (StringDataToEncrypt, PasswordOrPhrase,
PasswordHint)
Decrypt_Binary / Decrypt_Bit / Decrypt/Char / Decrypt_DB
(EncryptedData, PasswordOrPhrase)
```

DB2 Built-In Functions Example

```
CREATE TABLE EMPL
(EMPNO VARCHAR(64) FOR BIT DATA,
EMPNAME CHAR(20),
CITY CHAR(20),
SALARY DECIMAL(9,2))
IN DSNDB04.RAMATEST ;

COMMIT;

SET ENCRYPTION PASSWORD = 'PEEKAY' WITH HINT 'ROTTIE';

INSERT INTO EMPL(EMPNO, EMPNAME, SALARY)
VALUES (ENCRYPT('123456'),'PAOLO BRUNI',20000.00) ;

INSERT INTO EMPL(EMPNO, EMPNAME, SALARY)
VALUES (ENCRYPT('123457'),'ERNIE MANCILL',20000.00) ;
```

From Redbook SG24-7959, Security Functions of IBM DB2 10 for z/OS

New DB2 Built-In Functions (FL505 APAR PH09191)

- ENCRYPT_DATAKEY(expression, key-label-name,'AES256D'/'AES256R')
  - expression – the data to be encrypted
    - Data types
      - BIGINT, INTEGER, DECIMAL, CHAR, VARCHAR, GRAPHIC, VARGRAPHIC -> VARBINARY
      - CLOB, DBCLOB -> BLOB
  - key-label-name – the CKDS key label (stored as metadata)
  - AES256R or AES256D – CBC algorithm with a 256-bit AES key. CBC requires an Initialization Vector
    - Random IV – creates unique ciphertext for every cleartext value
    - Fixed (or Deterministic) IV – same cleartext results in same ciphertext

New DB2 Built-In Functions (FL505 APAR PH09191)

- DECRYPT_DATAKEY_type(encrypted-data)
  - Type – one of multiple types
  - Encrypted-data – the output from the ENCRYPT_DATAKEY function
  - Key label is not needed because that is stored as metadata by the ENCRYPT_DATAKEY BIF

DECRYPT_DATAKEY_type

- Integer
  - DECRYPT_DATAKEY_INTEGER(encrypted-data)
  - DECRYPT_DATAKEY_BIGINT(encrypted-data)
- Decimal
  - DECRYPT_DATAKEY_DECIMAL(encrypted-data,precision,scale)
- String
  - DECRYPT_DATAKEY_VARCHAR(encrypted-data,ccsid-constant)
  - DECRYPT_DATAKEY_CLOB(encrypted-data ,ccsid-constant)
  - DECRYPT_DATAKEY_VARGRAPHIC(encrypted-data ,ccsid-constant)
  - DECRYPT_DATAKEY_DBCLOB(encrypted-data ,ccsid-constant)
- Bit
  - DECRYPT_DATAKEY_BIT(encrypted-data)

Guardium Data Encryption Tool (5655-P03)

- DB2 EDIT Procedures (EDITPROC); IMS Segment Edit/Compression exit routine
  - No application changes required
  - Encrypted row same length as clear row
  - One key per table or segment
  - Indexes are not encrypted
- DB2 Field Procedures (FIELDPROC)
  - No application changes required
  - One key label specified in the FIELDPROC
  - Indexes can be encrypted
  - Columns must be < 254 bytes; Column names must be < 18 chars in length
- User-defined function (UDF)
  - No application changes required; Minimally disruptive, columns encrypted in place
  - Indexes can be encrypted
  - One key, label specified in the UDF
  - All data types supported by UDFs can be encrypted
  - VIEW/TRIGGER – provides access control to the cleartext

The Good and the Bad

- What gets encrypted?
  - Log records
  - Data buffers
  - Image copies
- Implementation
  - Unique EDITPROC / FIELDPROC / UDF or IMS Segment – based on key type (clear, secure, protected)
  - Driver routine – DECENADV
    - Compress, then encrypt
  - Subsystem required for a couple of IMS Segment exits
    - SYS1.PARMLIB(IEFSSNxx)
      - SUBSYS SUBNAME(xxxx) INITRTN(DECSSI20)
    - Dynamic definition
      - SETSSI ADD,SUB=xxxx,INITRTN=DECSSI20
    - Where xxxx is a unique name for your subsystem

First time encrypting or Reencipher

- Generate Key
- Prepare EDITPROC using Data Encryption Tool providing ICSF Keylabel
- Unload target table
- DROP / RECREATE table specifying EDITPROC
- LOAD table

EDITPROC restrictions

- LOB, DECFLOAT or XML columns not allowed
- ROWID columns not allowed
- Unicode columns not allowed for EBCDIC tables
- Accelerator-only table cannot be defined with an EDITPROC
- Tables using WITH ROW ATTRIBUTES clause
  - May not use ALTER ADD COLUMN
  - May not use ALTER RENAME COLUMN
  - May not contain:
    - A LOB, ROWID or XML Column
    - Identify column
    - Security label column
    - A column name > 18 EBCDIC bytes

FIELDPROC Restrictions

- Column must be built-in character or graphic string, not an LOB
- Column length must not be > 255 bytes
- Column name must be less than 18 bytes in length
- Column cannot specify a DEFAULT value
- Column must not be defined with both DEFAULT and FIELDPROC
- Cannot add to an existing column
- CREATE PERMISSION and CREATE MASK not allowed for columns with a FIELDPROC
- Cannot have Unicode columns in EBCDIC tables
- CREATE TABLE cannot contain both a FIELDPROC and a CCSID 1200 or CCSID 1208 clause
- Cannot be used with indices that are partitioned by range (PBR)
- Index key–expressions may not contain columns with a FIELDPROC

May not return correct data

Consider a table with an encrypted index and you select on that index

- … WHERE ACCOUNT NUMBER = 11111 …
  - Tool will encrypt 11111 and search for cells with that ciphertext, returning a match
- … WHERE ACCOUNT NUMBER BETWEEN 11111 AND 2222
  - Tool will encrypt 11111 and 22222 and search for cells that fall between the two ciphertext values

Decisions, Decisions … Which one should I use?

- Data set encryption
  - **+** Transparent
  - ***** Key Management
- Application encryption
  - **+** Most granular control
    - What gets encrypted
    - Key type (secure vs clear vs protected key)
  - **-** Performance impact
  - ***** Key Management

- Guardium Infosphere Database Encryption Tool for DB2 and IMS

Other DB2 Encryption

- Between DB2 databases
  - zIIP Assisted IPSec (VPN) on z/OS
- DASD Encryption
  - Protects the data when the DASD leaves your control, it does not protect the data from internal users
- Tape Encryption
  - Log files
  - Database unloads

Data set encryption for DB2 (PE)

- Encrypting your data with z/OS DFSMS data set encryption
  - https://www.ibm.com/support/knowledgecenter/SSEPEK_11.0.0/seca/src/tpc/db2z_dfsmsencryptionsupport.html
  - https://www.ibm.com/support/knowledgecenter/SSEPEK_12.0.0/seca/src/tpc/db2z_dfsmsencryptionsupport.html
- IBM Community
  - https://community.ibm.com/community/user/hybriddatamanagement/blogs/paul-mcwilliams1/2019/12/09/now-available-in-db2-12-db2-for-zos-support-for-zo

Announcement Letter 221-003 on 03/09/21

- IBM Db2 12 for z/OS and Db2 Tools for z/OS help optimize your investments with improved efficiency, scalability, and business readiness
  1. Function Level 505: Built-in functions for encryption using key labels,
  2. Function level 502: Provides more granular control over security to the database administration with support for transparent data encryption.
  3. Db2 12 for z/OS -> Idea: DB24ZOS-I-1024 z15: Dump security by encrypting buffer pool information
  4. Db2 12 for z/OS -> Idea: DB24ZOS-I-319 Show encryption on data-set level
  5. Db2 12 for z/OS -> Idea: DB24ZOS-I-239 Db2 SERVER_ENCRYPT enhancement
  6. Db2 12 for z/OS -> Idea: DB24ZOS-I-111 Db2 for ZOS DSNLEUSR proc should not require crypto card
  7. Db2 High Performance Unload for z/OS -> RFE: 116453 HPU 4.3 and support pervasive encryption

Data Encryption for Databases - Reference Materials

- SC19-3219 IBM Infosphere Guardium Data Encryption for DB2 and IMS Databases Version 1 Release 2 User Guide
- Redbooks
  - SG24-6465 DB2 UDB for z/OS Version 8 Performance Topics
  - SG24-7959 Security Functions of IBM DB2 10 for z/OS (Sept. 2011, doesn't cover FIELDPROCs and UDFs)
  - Securing and Auditing Data on DB2 for z-OS - SG24-7720
- Articles
  - Best Practices for Implementing IBM Data Encryption for DB2 and IMS Databases
    - http://publibfp.dhe.ibm.com/epubs/pdf/c2790010.pdf

Presentations

- Phoenix Share
  - DB2 for z/OS – Encrypting Your Data End to End by Gaya Chandran – Session #16444?
- Tom Hubbard Presentation – Database Encryption on z/OS
  - https://kiesslich-consulting.de/download/2016/C04_C12_Tom_Hubbard.pdf
- z15 Crypto Performance White Paper
  - https://www.ibm.com/downloads/cas/6K2653EJ

Questions?