DFSMS Dataset Encryption

With IBM Z Pervasive Encryption

Thomas Reed
IBM DFSMS Support
TReed@us.ibm.com

Please note

- IBM's statements regarding its plans, directions, and intent are subject to change or withdrawal without notice and at IBM's sole discretion.
- Information regarding potential future products is intended to outline our general product direction and it should not be relied on in making a purchasing decision.
- The information mentioned regarding potential future products is not a commitment, promise, or legal obligation to deliver any material, code or functionality. Information about potential future products may not be incorporated into any contract.
- The development, release, and timing of any future features or functionality described for our products remains at our sole discretion.
- Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput or performance that any user will experience will vary depending upon many factors, including considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve results similar to those stated here.

Agenda

- Pervasive Encryption
  - Role of z/OS data set encryption
- Client Value
- Considerations for data set encryption usage
- Implementation of data set encryption
- Getting Started
- Resources

Data protection and compliance are business imperatives

"It's no longer a matter of if, but when …"

28%

Likelihood of an organization having a data breach in the next 24 months [1]

European Union General Data Protection Regulation (GDPR)

Payment Card Industry Data Security Standard (PCI-DSS)

$3.6M

Average cost of a data breach in 2017 [2]

Of the **9 Billion** records breached since 2013 only **4%** were encrypted [3]

Health Insurance Portability and Accountability Act (HIPAA)

1, 2 Source: 2017 Ponemon Cost of Data Breach Study: Global Overview -- http://www.ibm.com/security/data-breach/
3 Source: Breach Level Index -- http://breachlevelindex.com/

Implementing encryption can be a complex process

Organizations struggle with questions such as:

1. What data should be encrypted?
2. Where should encryption occur?
3. Who is responsible for encryption?

Comprehensive data protection requires a huge investment to deploy point solutions and/or enable encryption directly in the applications.

Pervasive Encryption with IBM Z

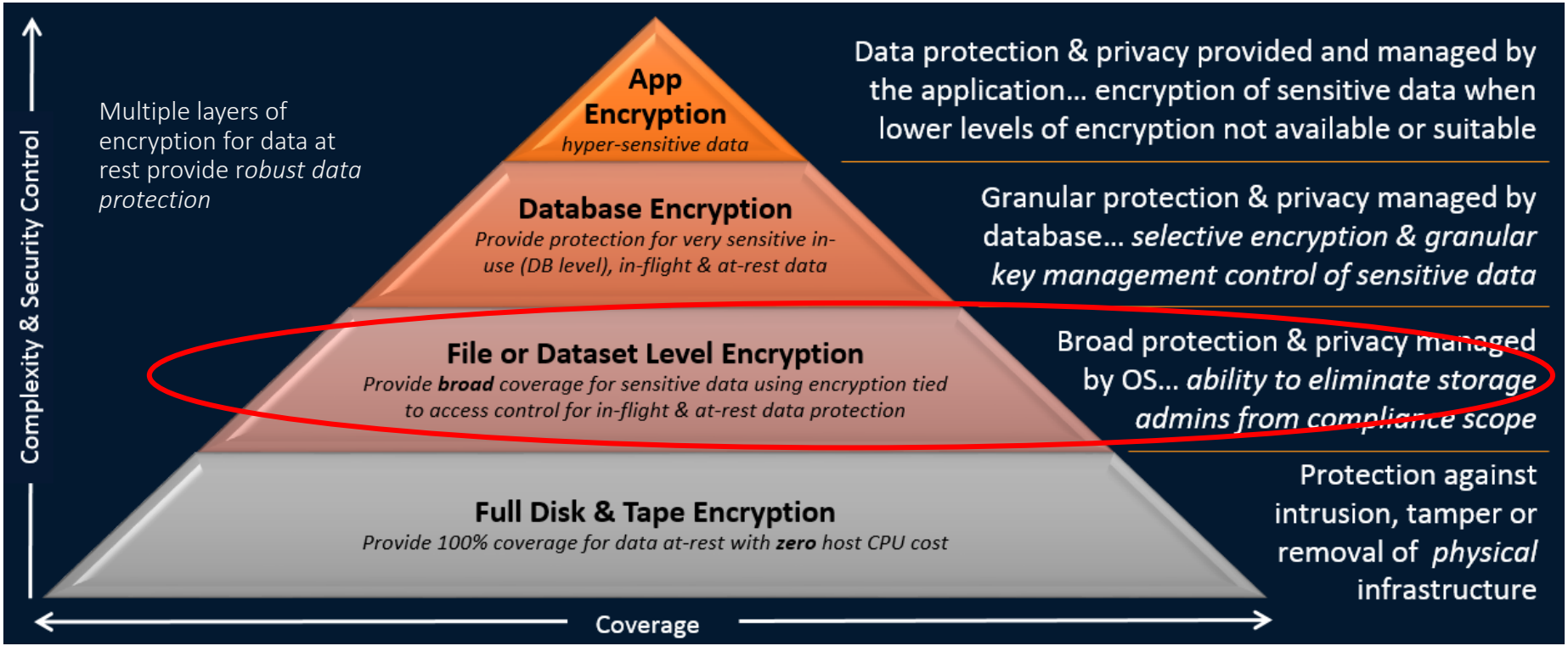
GUIDE SHARE EUROPE UK REGION

Integrated Crypto Hardware	Hardware accelerated encryption on every core, CPACF performance improvements of 7x Crypto Express6S – PCIe Hardware Security Module (HSM) & Cryptographic Coprocessor
Data at Rest	Broadly protect Linux file systems and z/OS data sets using policy controlled encryption that is transparent to applications and databases
Clustering	Protect z/OS Coupling Facility data end-to-end, using encryption that's transparent to applications
Network	Protect network traffic using standards based encryption from end to end, including encryption readiness technology to ensure that z/OS systems meet approved encryption criteria
Secure Service Container	Secure deployment of software appliances including tamper protection during installation and runtime, restricted administrator access, and encryption of data and code in-flight and at-rest
Key Management	The IBM Enterprise Key Management Foundation (EKMF) provides real-time, centralized secure management of keys and certificates with a variety of cryptographic devices and key stores

And we're just getting started …

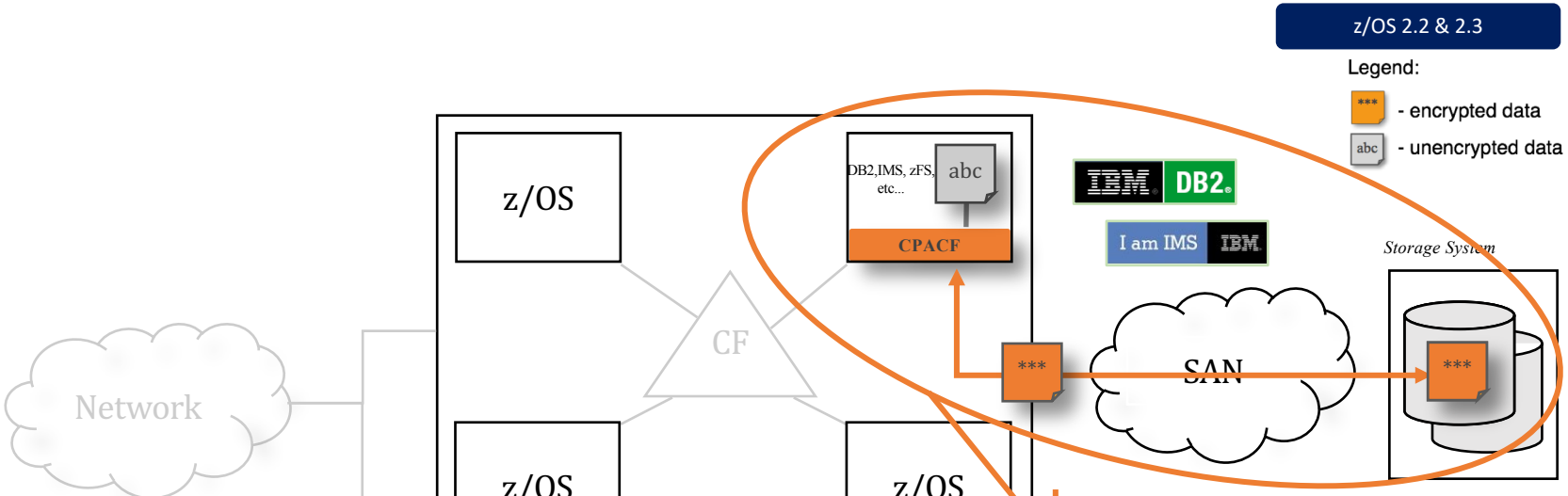
The Encryption Pyramid

...for data at rest



z/OS Dataset Encryption

Protection of data at rest



Client Value Proposition:
Reduced cost of encryption along with simple policy controls allows clients to enable extensive encryption to protect data in mission critical databases including DB2®, IMS™ and VSAM

z/OS Dataset Encryption:

- Application transparent & enabled by policy
- Encryption tied to fine grained access control
- Host encryption via CPACF as data written-to or read-from disk.
- Supports ext. format sequential & VSAM
- Includes HSM & DSS migration/backup of encrypted data sets
- Replicated data remains encrypted
- Supports: CICS®, DB2, IMS, Logger, & zFS

In-memory system or application data buffers will not be encrypted

Agenda

- Pervasive Encryption
  - Role of z/OS data set encryption
- **Client Value**
- Considerations for data set encryption usage
- Implementation of data set encryption
- Getting Started
- Resources

z/OS Data Set Encryption – Client Value

*Clients who are required to protect customer data can leverage the IBM Z hardware encryption for **data at rest** through existing **policy management**… **without application changes.***

1. – **No application changes required**
2. – **Data set level granularity**
3. – **Supports separation of access control for data set and encryption key label**
4. – **Enabled through RACF and / or SMS policy**
5. – **Audit readiness**

Designed to take advantage of the processing power of the z14

Agenda

- Pervasive Encryption
  - Role of z/OS data set encryption
- Client Value
- **Considerations for data set encryption usage**
- Implementation of data set encryption
- Getting Started
- Resources

Application transparency via access methods

- **Supported access methods/data set types:**
- **BSAM/QSAM**
  - Sequential data sets
    - Extended format only
- **VSAM and VSAM/RLS**
  - KSDS, ESDS, RRDS, VRRDS, LDS
    - Extended format only

- **Data set types that are *not* extended format**
  - **Basic and Large format sequential**
  - **PDS/PDSE**
  - **Tape data sets**
  - **BDAM**
- **Note:** The following sequential data sets cannot be extended format
  - **Temporary data sets**
  - **SORTWK data sets**

- Covers DB2, IMS, zFS, CICS/VSAM, Middleware, Logs, Batch, & ISV Solutions*. Refer to product documentation for information regarding support.
  - (*) For those applications that use the licensed Media Manager services, changes to Media Manager interfaces required to access encrypted data sets.

Transparent! No application changes or awareness that sequential or VSAM data is encrypted when accessed using the standard access method APIs.

Extended format data sets

- **Allocated with DSNTYPE keyword**
  - JCL DSNTYPE=EXTREQ or EXTPREF
  - SMS Data class DSNTYPE=EXTR or EXTP
- **SMS-managed DASD data sets**
- **Can be compressed format**
  - SMS Data class COMPACTION
    - Sequential: Generic, Tailored, zEDC
    - VSAM KSDS: Generic

❖Data sets that **can be** allocated as extended format
- ❖Db2 (table spaces and logs)
- ❖IMS (certain dbs, logs, trace data sets)
- ❖CICS/VSAM
- ❖zFS
- ❖Etc

Note: Review product documentation for support.

- **Restrictions:**
  - System data sets (such as Catalogs, SHCDS, HSM data sets) should not be created as extended format, unless otherwise specified.
  - Cannot be opened for EXCP processing
  - Sequential compressed format data sets cannot be opened for UPDATE processing

❖Data set types that are ***not*** extended format
- ❖Basic and Large format sequential
- ❖PDS/PDSE
- ❖BDAM
- ❖Tape data sets

❖Note: The following sequential data sets **cannot be** extended format
- Temporary data sets
- SORTWK data sets

After evaluating restriction, OK to convert to extended format, compression and encryption at the same time.

Additional Data Set Restrictions

- System data sets (such as Catalogs, SHCDS, HSM data sets) must not be encrypted, unless otherwise specified
- Data sets used before ICSF is started must not be encrypted
- Sequential (non-compressed) extended format data sets with a BLKSIZE of less than 16 bytes cannot be encrypted
- DFSMSdss REBLOCK keyword is ignored on COPY and RESTORE functions. DFSMSdss ADRREBLK installation exit will not be called for encrypted data sets
- DFSMSdss does not support VALIDATE processing when backing up encrypted indexed VSAM data sets. VALIDATE will be ignored.

Consider enabling data set level compression

GUIDE SHARE EUROPE UK REGION

Encrypted data does not compress downstream of encryption

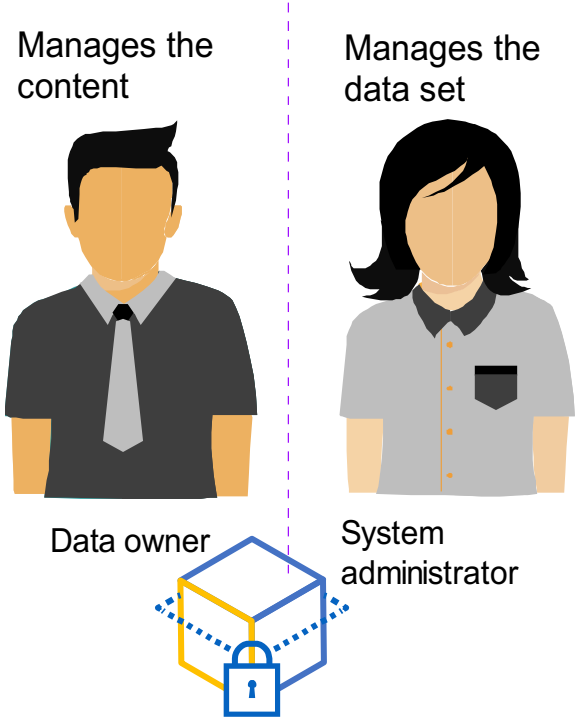
- Creating encrypted data sets may impact expected savings with disk or tape device compression.
- Backup and migration of encrypted data sets may impact expected savings with disk or tape device compression.
- Replicated data that is being compressed in the SAN infrastructure by DWDM technology will no longer be effective trying to compress encrypted data
- In addition, if deduplication of data is supported, data in encrypted data sets can prevent deduplication from working.

Where possible, convert to compressed format data sets

- When data set level compression requested, access methods handle compression before encryption for compressed format encrypted data sets.
  - Data class COMPACTION option
- Supported data set types and compression types:
  - Sequential extended format data sets
    - Generic, tailored, or zEDC compression
    - ***Restriction note:*** Sequential compressed format data sets cannot be opened for UPDATE.
  - VSAM extended format KSDS data sets
    - Generic compression

Access Control - Segregation of Duties

- Data owners that must access content will need authority access to the data set **AND** access to the encryption key label
- Storage administrators who only manage the data sets need access to the data set but **NOT** access to the key label (thus protecting access to the content)
- Different keys can be used to protect different data sets – ideal for multiple tenants or data set specific policies.
- Many utilities can process data preserving encrypted form



Limit access to data in clear! Remove certain roles from compliance scope….by controlling access to the data through SAF policies.

Backup, Migration and Replication

GUIDE SHARE EUROPE UK REGION

System services that manage the data set (as opposed to the data) ensure the data remains in encrypted form

- During DFSMSdss functions, COPY, DUMP and RESTORE
- During DFSMShsm functions, Migrate/Recall,Backup/Recover, Abackup/Arecover, Dump/Data Set Restore, FRBACKUP/FRRECOV DSNAME.
- Encrypting with data set encryption ensures security across all storage tiers, including cloud storage with Transparent Cloud Tiering
- During track based copy (PPRC, XRC, FlashCopy, Concurrent Copy, etc) operations since read track will get the track image which has the already encrypted data.
    - Key material must be available on target systems to access encrypted data sets

Storage admins (or others) that perform these system services would not require access to the key label.

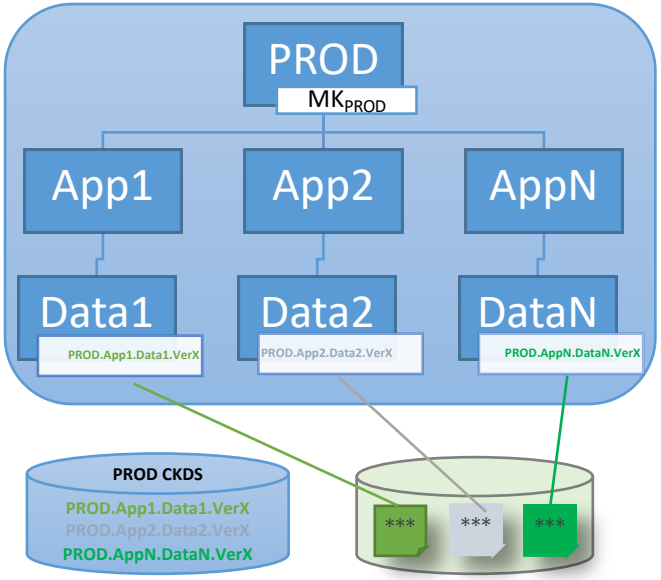
Transmitting data

- System services that transmit data will typically retrieve the data using the access methods, thus the data in encrypted data sets is decrypted within these services prior to transmit.
- When transmitting sensitive data, as today, use the secure versions of these services.
  - Connect: Direct
  - FTPS
  - XMIT

Users/System admins performing these functions will require access to the key label.

Naming Conventions & Granular Access Control

Leveraging naming conventions & z Security to enforce separation across application instances



- **Naming conventions can be used to segment applications, data, and keys, e.g.**
  - Environment: PROD, QA, TEST, DEV
  - Application: App1, App2,…, AppN
  - Data-Type: Account, Payroll, Log
  - Version: Ver1, Ver2,…,Verx
- **Application resources (data sets, encryption keys) can be assigned names based on naming conventions, e.g.**
  - PROD.APP2.LOG.VER10
  - PROD.APP1.PAYROLL.KEY.VER7
- **Security rules can be used to enforce separation with granular access control for application resources and encryption keys**

Flexible! Data set encryption is designed to be flexible in allowing as much granularity as desired when identifying key labels for data sets. There is no limit as to how many key labels and encryption keys are used across the data sets…however, planning for key management is critical.

Creating encrypted data sets via policy

- **A data set is defined as 'encrypted' when a key label is supplied on allocation of a new sequential or VSAM extended format data set**

- **A key label supplied in any of the following (using order of precedence as follows):**
  - RACF Data set profile DFP segment
  - JCL, Dynamic Allocation, TSO Allocate, IDCAMS DEFINE
  - SMS Construct: Data Class
    - Note: Can specify data class on ISPF 3.2 to allocate an encrypted data set

Ease of use! Easy to create an encrypted data set just by specifying a key label. Even easier when enabled via RACF or SMS policy.

3

Creating encrypted data sets – choosing key label source

- **RACF Data set profile DFP segment**
  - Provides ability to support data set encryption via a security policy, beneficial for audit purposes
  - Identifies data sets via a discrete or generic HLQ
  - Enables security administrator to have control over protection of data, including which data sets are encrypted and which key label is used.
- **JCL, Dynamic Allocation, TSO Allocate, IDCAMS DEFINE**
  - Provides ability for a specific job to identify specific data sets to be encrypted.
  - Useful for initial testing
- **Data Class**
  - Provides ability to support data set encryption via SMS policy
  - The encryption data class could be explicitly specified for a data set, or it could be determined by defaults defined by an ACS routine.
  - ACS routines are flexible such that the encryption data class could be determined for data sets according to allocation parameters, data set sizes, object or data set names, and other variables.
  - The storage administrator should work with security administrator to understand which data sets are to be encrypted and which key labels are to be used.

Security Admin

User

Storage Admin

In choosing a source for key label, consider how to control which data sets are to be encrypted and which key labels are to be used. Many clients prefer to place this control under the role of the security administrator.

z/OS Data Set Encryption – Encryption keys

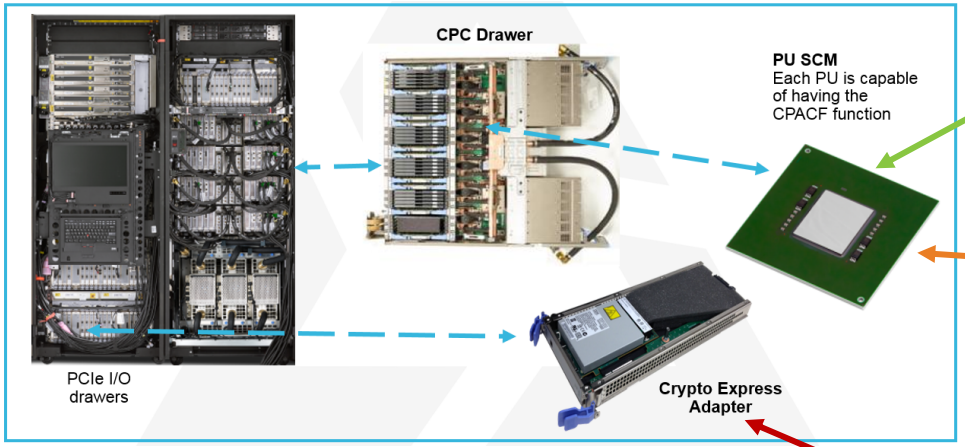
- **Key label: 64-byte label of a key in the ICSF Cryptographic Key Data Set (CKDS)**
  - Required to access an encrypted data set
- **Encryption data key:**
  - Require AES-256 bit key
  - Must be set up in CSFKEYS as a protected key
  - Recommend secure keys (protected by Crypto Express AES Master Key)
- **Encryption mode:**

– DFSMS uses XTS mode

Key management is critical for a robust security strategy

Understanding Clear, Secure and Protected Keys

- **Using secure keys ensures that key values stored in the ICSF Key Data Sets are protected with encryption.**



High speed

Clear Key

Key values are not encrypted. Crypto operations may be performed in CPACF or on a Crypto Express adapter

Speed & Security Hybrid

Protected Key

Key values are encrypted under a CPACF wrapping key. **Crypto operations are performed only using CPACF**

High security

Secure Key

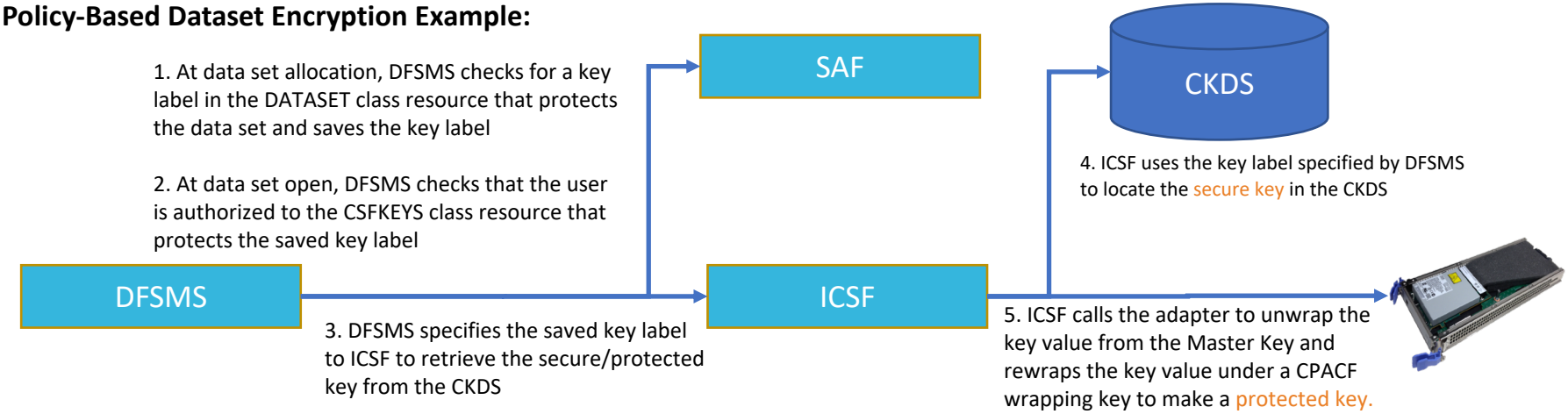
Key values are encrypted under a Master Key. Crypto operations are performed only on a Crypto Express adapter

Note: With z/OS data set encryption, protected keys are implicitly created from secure keys.

- *Only protected keys created*
- *from secure keys should be*
- *used for Pervasive Encryption.*

Understanding Key Labels

- Every record in the CKDS has an associated key label.
- When user applications or z/OS components invoke ICSF callable services (i.e. APIs), the application can specify a key label as a parameter to identify the key for the callable service to use.
- **System Authorization Facility (SAF) policies control which users can use which keys (and callable services).**
- The **CSFKEYS class** controls access to cryptographic keys in the ICSF CKDS and PKDS and enables/disables the use of protected key.
- The **CSFSERV class** controls access to ICSF callable services and ICSF TSO panel utilities.
- **Policy-Based Dataset Encryption Example:**



New data set allocation via policy based storage mgmt

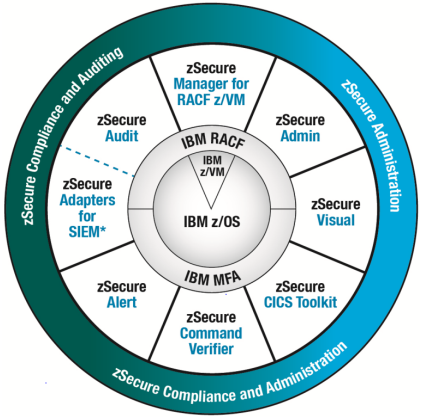
- **DFSMS Storage Management Subsystem (SMS) derives key label (from one or more sources) to be used for the encrypted data set**
  - Derived key label stored in Catalog
    - New encryption cell (non-VSAM NVR, VSAM VVR)
      - Once key label stored in catalog for a data set, **NO** ability to alter it. Any subsequent change to RACF Data set profile or Data Class will not affect existing data sets
  - Encryption indicator set in volume table of contents (VTOC)
    - Format 1/Format 8 DSCB flag (**DS1ENCRP**)
  - New allocation message indicating data set is an encrypted data set with derived key label

IGD17150I DATA SET dsname IS ELIGIBLE FOR ACCESS METHOD ENCRYPTION.

KEY LABEL IS (key_label)

Audit readiness

- **Auditor can rely on system interfaces, not individuals, for compliance.**
- Encryption attributes displayed in various system interfaces
  - SMF records
  - DCOLLECT records
  - LISTCAT
  - IEHLIST LISTVTOC
  - Catalog Search Interface (CSI)
  - ISITMGD



zSecure also collects, formats and enriches data set encryption information that is sent to SIEMs including IBM QRadar® for enhanced enterprise-wide security intelligence.

Simplifies compliance! Allows enhanced tooling to help simplify the audit process.

Agenda

- Pervasive Encryption
  - Role of z/OS data set encryption
- Client Value
- Considerations for data set encryption usage
- Implementation of data set encryption
- **Getting Started**
- Resources

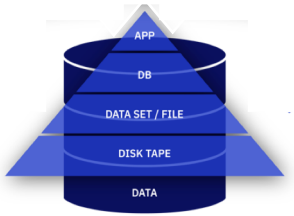
z/OS Data Set Encryption - Getting Started

1) **Choose an application**
2) **Prepare test environment**
3) **Enable encryption (4 steps)**
4) **Test & verify**
5) **Plan for production rollout**

Pervasive encryption client advocacy program

z/OS Data Set Encryption – 1) Choose an application

- **Questions:**
- Is your enterprise driving a top down encryption initiative?
  - e.g. GDPR, PCI DSS, etc..
- What do you expect to be the first use case for data set encryption?



- CICS/VSAM application
- DB2 database
- IMS database
- Batch workload
- Log data sets (system logger)

Note: Data set encryption supports extended format sequential and VSAM

z/OS Data Set Encryption – 2) Prepare test environment

- **Hardware**
  - CPACF protected key (z196 or later for AES-XTS mode)
  - Crypto Express3 or later required for secure key
  - Recommend use of Crypto Express in test to validate crypto operational procedures (e.g. master key loading, master key change, etc…)
- **Setup & Configure ICSF**
  - Load AES master key
  - Recommend installing latest ICSF web deliverable (HCR77C1)
    (Can generate AES DATA keys using CKDS Browser)
- **Install/Update Base Software**
  - DFSMS: z/OS 2.2 + service or z/OS 2.3
  - RACF: z/OS 2.2 + service or z/OS 2.3
  - ICSF: HCR77A0-B1 + service or HCR77C0-C1
- **Install/Update Exploitation Software**
  - DB2, IMS, logger… vendor products?
- **Review fixcat to obtain all the latest maintenance**

Category	Description	Keyword
IBM.Function.DataSetEncryption	Fixes to enable and support the z/OS Data Set Encryption function.	DSENCRYPT/K

Questions:

- Is anyone still running z/OS 2.1 or earlier?

z/OS Data Set Encryption – 3) Enable Encryption (4 steps)

GUIDE SHARE EUROPE UK REGION

1 — ICSF Admin

Generate an encryption key and key label, store it in the CKDS .

2 — Security Admin

Setup RACF for use of key label

Allow secure key to be used as protected key via ICSF segment
- SYMCPACFWRAP
- SYMCPACFRET

– AND –

Grant access to key label

In RACF, permit access to new resource in FACILITY class

3 — Security Admin, Storage Admin, User

Associate the key label with the desired data set(s).

In RACF, alter DFP segment in data set profile - DATAKEY()

– OR –

In DFSMS, assign to data class

4 — Storage Admin, DBA, User

Migrate to encrypted data

DB2: Online Reorg — Non-disruptive

IMS HA Database: Online Reorg — Non-disruptive

zFS Container: zfsadmin encrypt — Non-disruptive

VSAM or Seq data set:
1. Stop application
2. Copy data
3. Restart application

Defining a robust key management strategy is critical!

z/OS Data Set Encryption – 4) Test and verify

z/OS Data Set Encryption – 5) Plan for production rollout

Questions:

- Is ICSF environment configured for Parallel Sysplex?
- Is ICSF environment configured for DR?
- Is an Enterprise Key Management system deployed?

- Configure ICSF & key store for high availability
- Configure ICSF & key store for DR
- Configure periodic logical back up of key store
- Deploy Enterprise Key Management system for backup & recovery
- Consider use of host based compression
- Plan key label naming convention and access control
- Evaluate encryption overhead

Enterprise Key Management

Encryption of data at enterprise scale requires robust key management

- **The current key management landscape can be characterized by clients who have …**
  - … already deployed an enterprise key management solution
  - … developed a self-built key management solution
  - … not deployed an enterprise key management solution

Key management for pervasive encryption must provide …

- Policy based key generation
- Policy based key rotation
- Key usage tracking
- Key backup & recovery

EKMF

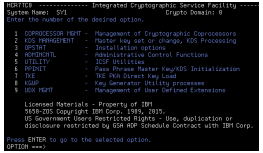
The IBM Enterprise Key Management Foundation (EKMF) provides real-time, centralized secure management of keys and certificates in an enterprise with a variety of cryptographic devices and key stores.

What IBM tools are available to manage keys?

GUIDE SHARE EUROPE UK REGION

Integrated Cryptographic Services Facility (ICSF)

ICSF provides callable services and utilities that generate, store, and manage keys, and also perform cryptographic operations.



Supports Master Keys and Operational Keys

Trusted Key Entry (TKE) Workstation

TKE securely manages multiple Cryptographic Coprocessors and keys on various generations of IBM Z from a single point of control.

Supports Master Keys and Operational Keys

Enterprise Key Management Foundation (EKMF)

EKMF securely manages keys and certificates for cryptographic coprocessors, hardware security modules (HSM), cryptographic software, ATMs, and point of sale terminals.

Supports Operational Keys

Security Key Lifecycle Manager (SKLM)

SKLM v2.7 provides key storage, key serving and key lifecycle management for IBM and non-IBM storage solutions using the OASIS Key Management Interoperability Protocol (KMIP) and IBM Proprietary Protocol (IPP).

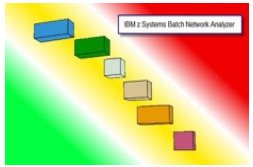
Supports Operational Keys <u>for Self Encrypting Devices (SEDs)</u>

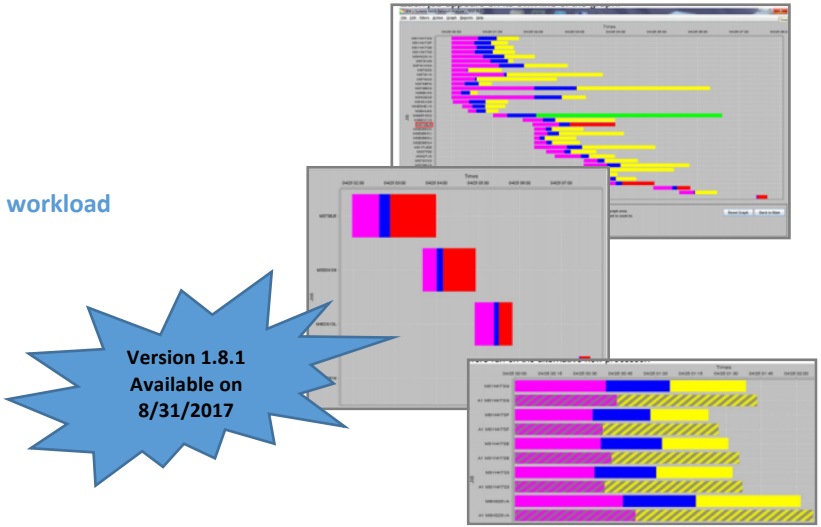
z/OS Data Set Encryption – Evaluate impact

Estimating CPU Cost of Data Protection

- ***z Batch Network Analyzer (zBNA)***
- **zBNA Background:**
- A no charge, “as is” tool originally designed to analyze batch windows
- PC based, and provides graphical and text reports
- Available on techdocs for customers, business partners, and IBMers
  http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/PRS5132
- Previously enhanced for zEDC to identify & evaluate compression candidates
- **zBNA Encryption Enhancements:**
- **Enhanced to help clients estimate encryption CPU overhead based on actual client workload SMF data**
- Ability to select z13 or z14 as target machine
- Support provided for
  - z/OS data set encryption
  - Coupling Facility encryption

zBNA 1.8.1



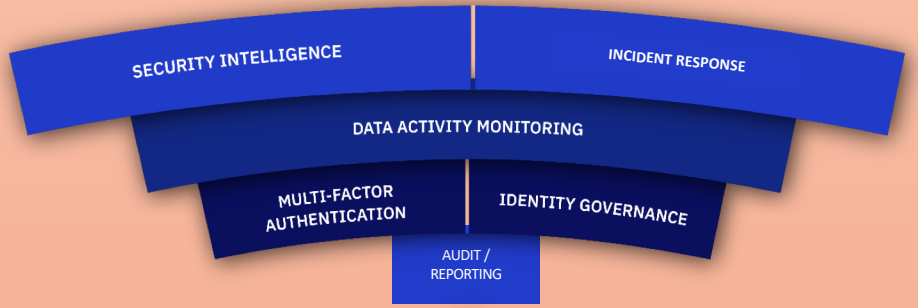


Version 1.8.1 Available on 8/31/2017

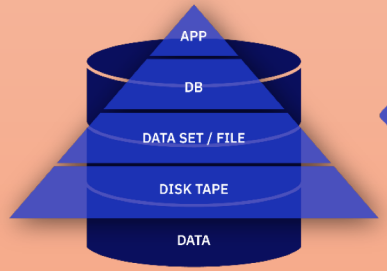
Note: z/OS Capacity Planning tool zCP3000 also updated to provide encryption estimates
http://w3-03.ibm.com/support/americas/wsc/cpsproducts.html

Protecting data at the core of the enterprise

Encryption is the solid foundation of a layered cybersecurity strategy



System hardening



IBM Z pervasive encryption

Traditional workloads and APIs:

- DB2
- CICS / VSAM
- IMS
- MQ

Relevant IBM Security Solutions:

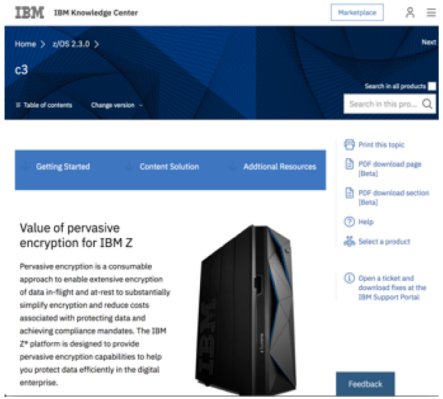
- IBM Security zSecure Suite
- IBM Security QRadar
- IBM Security Guardium Family
- IBM Multi-factor Authentication
- IBM Security Identity Governance
- Enterprise Key Management

Agenda

- Pervasive Encryption
  - Role of z/OS data set encryption
- Client Value
- Considerations for data set encryption usage
- Implementation of data set encryption
- Getting Started
- **Resources**

Resources

- **New:** Getting Started with z/OS Data Set Encryption Redbook
  http://www.redbooks.ibm.com/redpieces/abstracts/sg248410.html?Open
- **New:** IBM Z pervasive encryption landing page
  https://www.ibm.com/support/knowledgecenter/SSLTBW_2.3.0/com.ibm.zos.v2r3.izs/pervasiveEncryption.html
- IBM Z pervasive encryption solution guide (Knowledge Center)
  https://www.ibm.com/support/knowledgecenter/en/SSLTBW_2.3.0/com.ibm.zos.v2r3.izs/izs.htm
- IBM Z pervasive encryption FAQ:
  https://www-01.ibm.com/common/ssi/cgi-bin/ssialias?htmlfid=ZSQ03116USEN
- IBM Crypto Education page:
  https://ibm.biz/BdiAah
- zPET Test Reports:
  https://www.ibm.com/developerworks/community/groups/service/html/communitystart?communityUuid=43ea8e78-acbe-49f5-9290-379e4f4569cb
- MOP demo white paper:
  http://www-03.ibm.com/support/techdocs/atsmastr.nsf/WebIndex/WP102734
- Youtube Videos:
  - Data Set Encryption: https://www.youtube.com/watch?v=zdSXRUSmkb4
  - CF Encryption: https://www.youtube.com/watch?v=lTmsFWuJwJU
  - zERT: https://www.youtube.com/watch?v=1CgEcCTX_o8
  - MOP MPL Bank: https://www.youtube.com/watch?v=EP488nLdGts

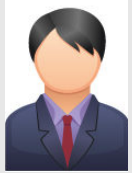
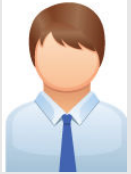


Agenda

- Pervasive Encryption
  - Role of z/OS data set encryption
- Client Value
- Considerations for data set encryption usage
- **Implementation of data set encryption**
- Getting Started
- Resources

Implementing encryption at the data set level

GUIDE SHARE EUROPE UK REGION

Role	ICSF Admin 	Security Admin 	Security Auditor 	Systems Prog 	Storage Admin (Data Mgr) 	User (Data Owner) 
Objective	Responsible for key mgmt. (defining keys, key labels), working with key mgmt. system; Manages ICSF and key changes	Identify data sets that need to be encrypted; Tie encryption to user access; Responsible for creating RACF profiles, assigning access to key labels	Update audit reports; Ensure audit and reporting compliance	Ensures system (hw/sw) supports encryption; work with Security Admin to determine if migration action needed to allow encryption	Assigns encryption to specific data classes; manage backup, migration and replication	Automatically create encrypted data sets; Runs applications, submits jobs
How	Defines key labels in CKDS associated with secure AES256 keys	Update key label in RACF data set profile; Modify user profiles with key labels and access permissions to files	List the catalog, etc to display encryption status	Ensure all systems that may need to access the data have the CKDS	Set key labels for data class using storage mgmt. panels (ISMF); Updates ACS rtns	Add key label to JCL or IDCAMS DEFINE CLUSTER;
Benefit	Manages key repository	Encrypt sensitive data; Prevent unauthorized access to data based on profiles	Determine encryption status to meet compliance	Manages HW/SW level on systems to support encryption	Manages SMS constructs that enable encryption	Automate creation of encrypted files without code changes

Not intended to be a complete list of responsibilities

z/OS data set encryption – High Level Steps

GUIDE SHARE EUROPE UK REGION

1 — ICSF Admin

Generate an encryption key and key label, store it in the CKDS .

Defining a robust key management strategy is critical!

2 — Security Admin

Setup RACF for use of key label

Allow secure key to be used as protected key via ICSF segment
- SYMCPACFWRAP
- SYMCPACFRET

– AND –

Grant access to key label

In RACF, permit access to new resource in FACILITY class

3 — Security Admin, Storage Admin, User

Associate the key label with the desired data set(s).

In RACF, alter DFP segment in data set profile - DATAKEY()

– OR –

In DFSMS, assign to data class

4 — Storage Admin, DBA, User

Migrate to encrypted data

DB2: Online Reorg — **Non-disruptive**

IMS HA Database: Online Reorg — **Non-disruptive**

zFS Container: zfsadmin encrypt — **Non-disruptive**

VSAM or Seq data set:
1. Stop application
2. Copy data
3. Restart application

1

Prepare ICSF CKDS for use

Generate an encryption key and key label, store it in the CKDS .

GUIDE SHARE EUROPE UK REGION

- **ICSF Admin must ensure keys exist**
  - Key labels defined in CKDS associated with secure AES256 keys
  - Various methods available to create key label and data keys, for example
    - ICSF CKDS Keys Panel (HCR77C1)
    - ICSF APIs (CSNBKGN, CSNBKRC2)
    - ICSF KGUP
    - EKMF
  - Use Crypto Express to protect keys in the CKDS as secure keys!

ICSF Admin

Data keys must be accessible EVERYWHERE that the encrypted data sets must be accessed.

z/OS data set encryption – High Level Steps

GUIDE SHARE EUROPE UK REGION

1 — ICSF Admin

Generate an encryption key and key label, store it in the CKDS .

Defining a robust key management strategy is critical!

2 — Security Admin

Setup RACF for use of key label

Allow secure key to be used as protected key via ICSF segment
- **SYMCPACFWRAP**
- **SYMCPACFRET**

– AND –

Grant access to key label

In RACF, permit access to new resource in FACILITY class

3 — Security Admin, Storage Admin, User

Associate the key label with the desired data set(s).

In RACF, alter DFP segment in data set profile - DATAKEY()

– OR –

In DFSMS, assign to data class

4 — Storage Admin, DBA, User

Migrate to encrypted data

DB2: Online Reorg — **Non-disruptive**

IMS HA Database: Online Reorg — **Non-disruptive**

zFS Container: zfsadmin encrypt — **Non-disruptive**

VSAM or Seq data set:
1. Stop application
2. Copy data
3. Restart application

2a

Prepare for access method access to ICSF CKDS Key provisioning service*

GUIDE SHARE EUROPE UK REGION

Security Admin

- **Setup SAF resources for ICSF service**

– **Security Admin** sets up access to the ICSF CKDS Key Record Read2 (CSNBKRR2) service

  - Define the RACF profile such that no one has access to the ICSF services. For example,
    - **RDEFINE CSFSERV * UACC(NONE)**
  - Allow everyone to have access to the callable service CSNBKRR2. For example,
    - **RDEFINE CSFSERV CSFKRR2 UACC(READ)**

OR

    - **PERMIT CSFKRR2 CLASS(CSFSERV) ID(*) ACCESS(READ)**
  - ***Note:*** The above are examples intended to show how an installation might set up CSFSERV profiles.

(*) Note: The above step is only required if CHECKAUTH(YES) is specified on the ICSF installation options data set. CHECKAUTH(NO) is the default.

2b

Setup access to key labels

GUIDE SHARE EUROPE UK REGION

Setup SAF resources for key-label

- **Security Admin** sets up profiles in the CSFKEYS general resource class based on installation requirements.
  - ***Any user that must access data in the clear must have access to the key label***
  - ***See examples on next page***

- **Security Admin** must also update the ICSF segment of the covering profile to allow ICSF to return a protected key: **SYMCPACFWRAP(YES) SYMCPACFRET (YES)**

Security Admin

Allows security admin to control who can get to data in the clear.

2b

Setup access to key labels

GUIDE SHARE EUROPE UK REGION

Setup SAF resources for key-label

– The following are ***examples:***

- Define the RACF profile such that no one has access to key-label

  RDEFINE CSFKEYS key-label UACC(NONE)

- Add the ICSF segment keywords to use the key label for a protected key

  RALTER CSFKEYS key-label ICSF(SYMCPACFWRAP(YES) SYMCPACFRET (YES))

- To allow key label to be used by JOHN when accessed by any application

  PERMIT key-label CLASS(CSFKEYS) ID(JOHN) ACCESS(READ)

- To allow key label to be used by MIKE only when accessed by DFSMS

  PERMIT key-label CLASS(CSFKEYS) ID(MIKE) ACCESS(READ) WHEN(CRITERIA(SMS(DSENCRYPTION)))

- To allow key label to be used by any user only when accessed by DFSMS

  PERMIT key-label CLASS(CSFKEYS) ID(*) ACCESS(READ) WHEN(CRITERIA(SMS(DSENCRYPTION)))

Security Admin

Allows security admin to control who can get to data in the clear.

2c

Prepare system to allow data set encryption

GUIDE SHARE EUROPE UK REGION

- **Set up SAF resource to enable data set encryption based on key label specification**
  - **Security Admin** must consider whether migration action should prevent creation of encrypted data sets via new resource in FACILITY class: **STGADMIN.SMS.ALLOW.DATASET.ENCRYPT**
    - Ensure all systems that may need to access the data have the CKDS with key material required to decrypt the data sets AND are at the correct HW/SW levels.
      - **RDEFINE FACILITY STGADMIN.SMS.ALLOW.DATASET.ENCRYPT UACC(NONE)**
    - To allow the system to create encrypted data sets when the key label is specified via a method outside of the DFP segment in the RACF data set profile, the user must have at least READ authority to the new resource in the FACILITY class.
      - **RALTER FACILITY STGADMIN.SMS.ALLOW.DATASET.ENCRYPT UACC(READ)**

Security Admin

Allows security admin to control who can create encrypted data sets.

2c

Prepare system to allow data set encryption

- **Set up SAF resource to enable data set encryption based on key label specification**
  - **Security Admin** must consider whether allocation of non-extended format data sets with key label should result in allocation failure via new resource in FACILITY class: **STGADMIN.SMS.FAIL.INVALID.DSNTYPE.ENC**
    - Default is to allow successful allocation for non-encrypted non-extended format data set.
      - Info message issued.
    - To fail the allocation, the user must have at least **READ** authority to the new resource in the **FACILITY** class.

Security Admin

Allows security admin to control whether key label should be ignored for unsupported data set types.

z/OS data set encryption – High Level Steps

GUIDE SHARE EUROPE UK REGION

1 — ICSF Admin

Generate an encryption key and key label, store it in the CKDS .

2 — Security Admin

Setup RACF for use of key label

Allow secure key to be used as protected key via ICSF segment
- SYMCPACFWRAP
- SYMCPACFRET

– AND –

Grant access to key label

In RACF, permit access to new resource in FACILITY class

3 — Security Admin, Storage Admin, User

Associate the key label with the desired data set(s).

In RACF, alter DFP segment in data set profile - DATAKEY()

– OR –

In DFSMS, assign to data class

4 — Storage Admin, DBA, User

Migrate to encrypted data

DB2: Online Reorg — **Non-disruptive**

IMS HA Database: Online Reorg — **Non-disruptive**

zFS Container: zfsadmin encrypt — **Non-disruptive**

VSAM or Seq data set:
1. Stop application
2. Copy data
3. Restart application

Defining a robust key management strategy is critical!

3

Creating encrypted data sets – supplying key labels

- **A data set is defined as 'encrypted' when a key label is supplied on allocation of a new sequential or VSAM extended format data set.**
- **Options for assigning key label (with order of precedence):**
- **RACF Data set profile DFP segment**
  - **Security Admin** can update RACF DS profile to request encryption by adding key label: DATAKEY
    - Note: Key label specified in the DFP segment is used regardless of the ACSDEFAULTS(xx) setting specified in SYS1.PARMLIB(IGDSMSxx)
- **JCL, Dynamic Allocation, TSO Allocate, IDCAMS DEFINE**
  - **User** can modify JCL or program to allocate specific data sets as encrypted by adding key label: JCL DSKEYLBL, Dynalloc DALDKYL, DEFINE KEYLABEL
- **Data Class**
  - **Storage Admin** can update specific data class(es) via ISMF to request encryption by adding key label: Data Set Key Label
  - **Storage Admin** can update ACS routines via ISMF to select data classes enabled for encryption

Security Admin

User

Storage Admin

3

Creating encrypted data sets – choosing key label source

GUIDE SHARE EUROPE UK REGION

- **RACF Data set profile DFP segment**
  - Provides ability to support data set encryption via a security policy, beneficial for audit purposes
  - Identifies data sets via a discrete or generic HLQ
  - Enables security administrator to have control over protection of data, including which data sets are encrypted and which key label is used.
- **JCL, Dynamic Allocation, TSO Allocate, IDCAMS DEFINE**
  - Provides ability for a specific job to identify specific data sets to be encrypted.
  - Useful for initial testing
- **Data Class**
  - Provides ability to support data set encryption via SMS policy
  - The encryption data class could be explicitly specified for a data set, or it could be determined by defaults defined by an ACS routine.
  - ACS routines are flexible such that the encryption data class could be determined for data sets according to allocation parameters, data set sizes, object or data set names, and other variables.
  - The storage administrator should work with security administrator to understand which data sets are to be encrypted and which key labels are to be used.

Security Admin

User

Storage Admin

In choosing a source for key label, consider how to control which data sets are to be encrypted and which key labels are to be used. Many clients prefer to place this control under the role of the security administrator.

3

Creating encrypted data sets – supplying key labels

- **A data set is defined as 'encrypted' when a key label is supplied on allocation of a new sequential or VSAM extended format data set.**
- **Options for assigning key label (with order of precedence):**
- **RACF Data set profile DFP segment**
  - **Security Admin** can update RACF DS profile to request encryption by adding key label: DATAKEY
    - Note: Key label specified in the DFP segment is used regardless of the ACSDEFAULTS(xx) setting specified in SYS1.PARMLIB(IGDSMSxx)
- ~~**JCL, Dynamic Allocation, TSO Allocate, IDCAMS DEFINE**~~
  - ~~**User** can modify JCL or program to allocate specific data sets as encrypted by adding~~

Note: To only allow new encrypted data sets through RACF policy (and thus controlled by security admin), do not provide users read access to resource STGADMIN.SMS.ALLOW.DATASET.ENCRYPT

RDEFINE FACILITY STGADMIN.SMS.ALLOW.DATASET.ENCRYPT UACC(NONE)

  - ~~**Storage Admin** can update ACS routines via ISMF to select data classes enabled for encryption~~

Security Admin

User

Storage Admin

Prepare for extended format on new data set allocation

Options for DSNTYPE

- **Setup SMS policy to request extended format via data class**
  - **Storage admin** can update specific data class(es) via ISMF to request extended format via DSNTYPE option
    - **SMS Data class DSNTYPE=EXTR or EXTP**
  - **Storage admin** can update ACS routines via ISMF to select data classes enabled for extended format
- **Setup job(s) to request extended format on JCL**
  - **User** can modify JCL to allocate specific data sets as extended format by adding DSNTYPE
    - **JCL DSNTYPE=EXTREQ or EXTPREF**
- ***Restriction note:*** Sequential extended format data sets cannot be opened for EXCP.

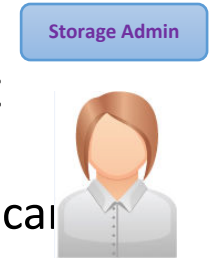
Storage Admin

User

Data set encryption requires extended format

Optionally, prepare for compression on new data set allocation

- **Setup SMS policy to request compression**
  - **Storage admin** can update specific data class(es) via ISMF to request compression via **COMPACTION** option
    - Sequential extended format data sets support generic, tailored, or zEDC compression
    - VSAM extended format KSDS supports generic compression (Only KSDS can be compressed format)
  - **Storage admin** can update ACS routines via ISMF to select data classes enabled for compression
- ***Restriction note:*** Sequential compressed format data sets cannot be opened for UPDATE.



z/OS data set encryption – High Level Steps

GUIDE SHARE EUROPE UK REGION

1 — ICSF Admin

Generate an encryption key and key label, store it in the CKDS .

2 — Security Admin

Setup RACF for use of key label

Allow secure key to be used as protected key via ICSF segment
- SYMCPACFWRAP
- SYMCPACFRET

– AND –

Grant access to key label

In RACF, permit access to new resource in FACILITY class

3 — Security Admin, Storage Admin, User

Associate the key label with the desired data set(s).

In RACF, alter DFP segment in data set profile - DATAKEY()

– OR –

In DFSMS, assign to data class

4 — Storage Admin, DBA, User

Migrate to encrypted data

DB2: Online Reorg — Non-disruptive

IMS HA Database: Online Reorg — Non-disruptive

zFS Container: zfsadmin encrypt — Non-disruptive

VSAM or Seq data set:
1. **Stop application**
2. **Copy data**
3. **Restart application**

Defining a robust key management strategy is critical!

4a

Converting existing data sets to encryption

- **Storage admin(*)** or **user** can copy an existing data set to a new target data set allocated as encrypted.
  - No utility available to perform a conversion without decrypting data from source and re-encrypting data onto target
  - Standard utilities can be used to perform the copy, for example
    - ISPF 3.3 Copy data set
    - IDCAMS REPRO
    - IEBGENER
- **DB admin:** For high availability, DB2 and IMS provide non-disruptive migration to encryption with DB online reorg function

The above could also be used to re-key an existing encrypted data set or DB to a new key.

4b

Accessing data in encrypted data sets

- **User can access data in encrypted data sets**
  - When accessed via BSAM, QSAM, VSAM or VSAM/RLS
    - Transparent access… ***no application changes***
      - Transparent to any applications or middleware making use of VSAM, QSAM, BSAM access methods.
      - Refer to individual product documentation to confirm support of z/OS data set encryption.
    - Data encrypted on writes and decrypted on reads
      - For those applications that use the licensed Media Manager services, changes to Media Manager interfaces required to access encrypted data sets.

4b

How can I be sure the data is encrypted?

GUIDE SHARE EUROPE UK REGION

- **Encryption attributes displayed in various system interfaces**
  - SMF records
  - DCOLLECT records
  - LISTCAT
  - IEHLIST LISTVTOC
  - Catalog Search Interface (CSI)
  - ISITMGD
- ***Note:*** To view encrypted data, can use **DFSMSdss PRINT Tracks**

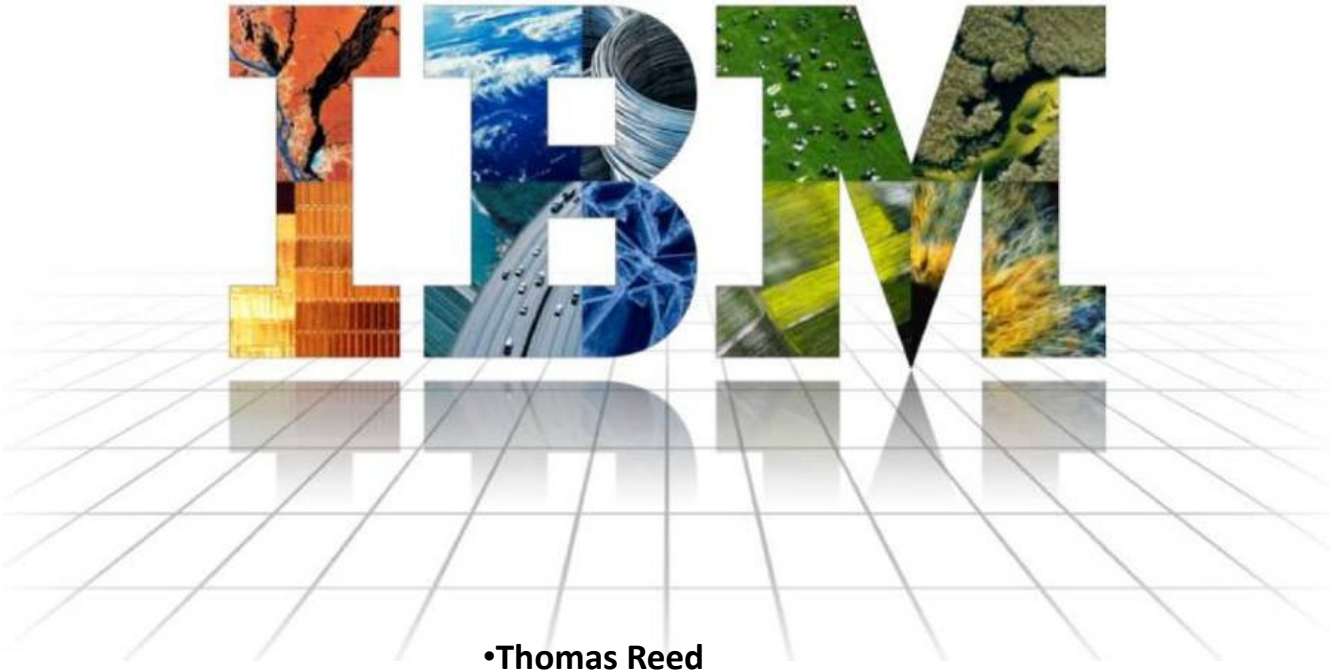
Notices and disclaimers

- © 2018 International Business Machines Corporation. No part of this document may be reproduced or transmitted in any form without written permission from IBM.
- **U.S. Government Users Restricted Rights — use, duplication or disclosure restricted by GSA ADP Schedule Contract with IBM.**
- Information in these presentations (including information relating to products that have not yet been announced by IBM) has been reviewed for accuracy as of the date of initial publication and could include unintentional technical or typographical errors. IBM shall have no responsibility to update this information. **This document is distributed "as is" without any warranty, either express or implied. In no event, shall IBM be liable for any damage arising from the use of this information, including but not limited to, loss of data, business interruption, loss of profit or loss of opportunity.** IBM products and services are warranted per the terms and conditions of the agreements under which they are provided.
- IBM products are manufactured from new parts or new and used parts. In some cases, a product may not be new and may have been previously installed. Regardless, our warranty terms apply."
- **Any statements regarding IBM's future direction, intent or product plans are subject to change or withdrawal without notice.**
- Performance data contained herein was generally obtained in a controlled, isolated environments. Customer examples are presented as illustrations of how those
- customers have used IBM products and the results they may have achieved. Actual performance, cost, savings or other results in other operating environments may vary.
- References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business.
- Workshops, sessions and associated materials may have been prepared by independent session speakers, and do not necessarily reflect the views of IBM. All materials and discussions are provided for informational purposes only, and are neither intended to, nor shall constitute legal or other guidance or advice to any individual participant or their specific situation.
- It is the customer's responsibility to insure its own compliance with legal requirements and to obtain advice of competent legal counsel as to the identification and interpretation of any relevant laws and regulatory requirements that may affect the customer's business and any actions the customer may need to take to comply with such laws. IBM does not provide legal advice or represent or warrant that its services or products will ensure that the customer follows any law.

Notices and disclaimers continued

- Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products about this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products. Questions on the capabilities of non-IBM products should be addressed to the suppliers of those products. IBM does not warrant the quality of any third-party products, or the ability of any such third-party products to interoperate with IBM's products. **IBM expressly disclaims all warranties, expressed or implied, including but not limited to, the implied warranties of merchantability and fitness for a purpose.**
- The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents, copyrights, trademarks or other intellectual property right.
- IBM, the IBM logo, ibm.com and [names of other referenced IBM products and services used in the presentation] are trademarks of International Business Machines Corporation, registered in many jurisdictions worldwide. Other product and service names might be trademarks of IBM or other companies. A current list of IBM trademarks is available on the Web at "Copyright and trademark information" at: www.ibm.com/legal/copytrade.shtml.
- .

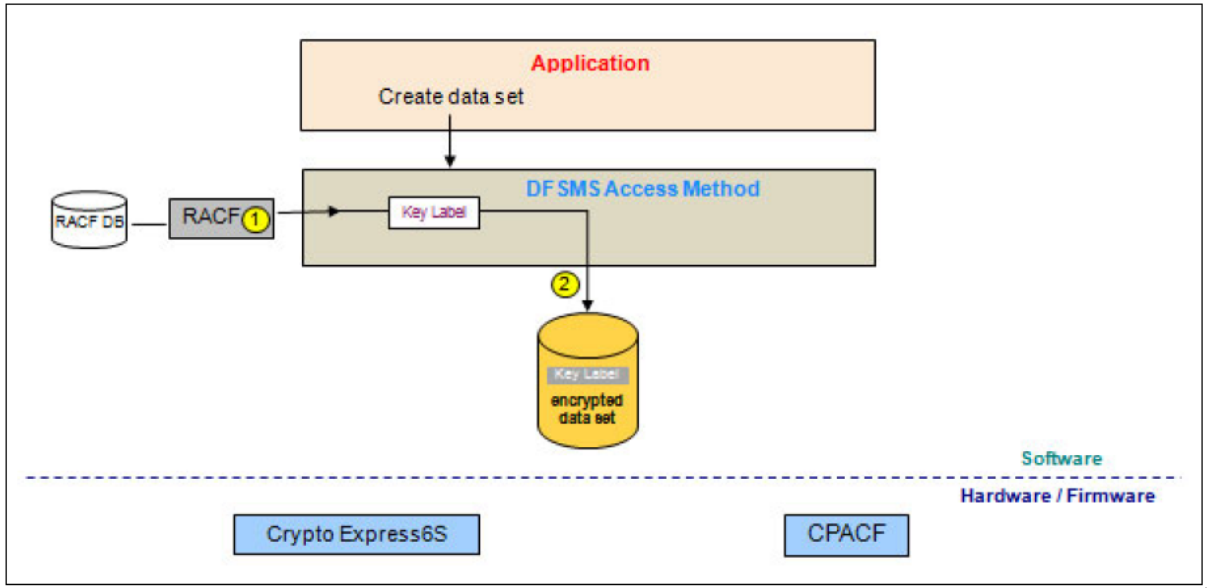
Thank you!!



- Thomas Reed
- z/OS DFSMS Support VSAM RLS/PDSE
- TReed@us.ibm.com

BACKUP

Example: Processing key label during data set create

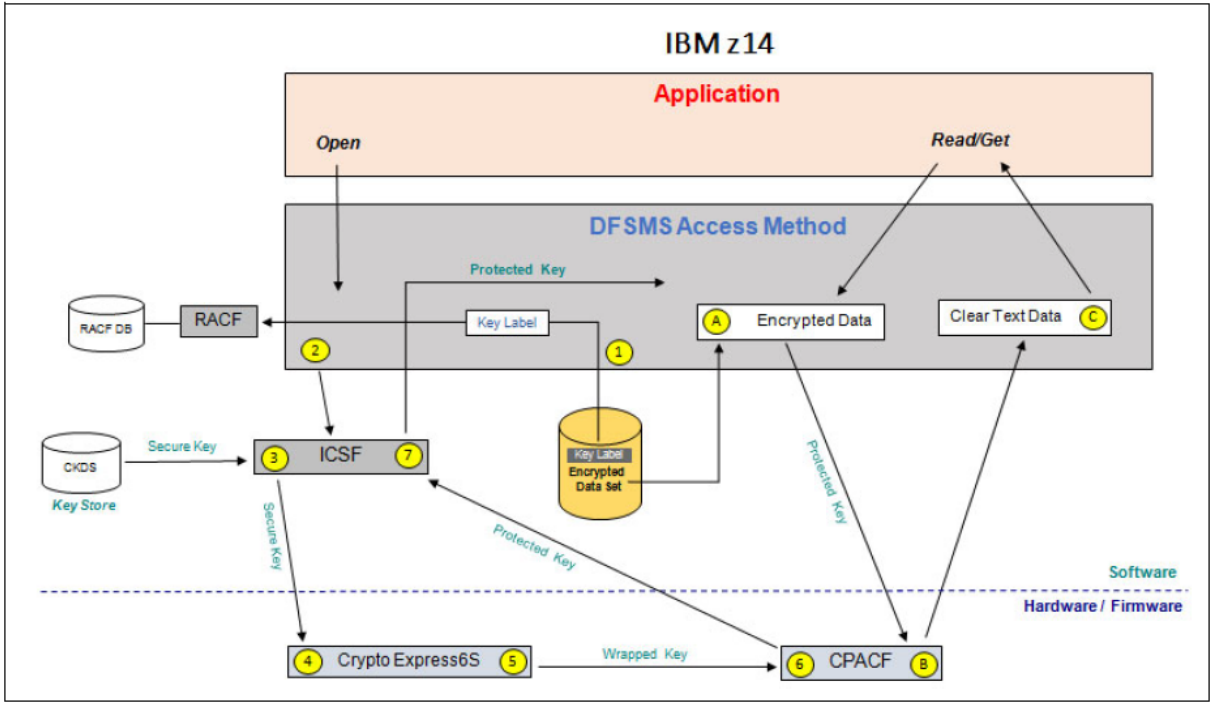


In this example,

- SMS finds key label specified in RACF data set profile DFP segment
- SMS validates the key label
- Key label stored in new encryption cell in VVDS

NOTE: No call to ICSF, Crypto Express or CPACF at data set create

Example: Processing keys during input



In this example, Open…

- Retrieves key label from encryption cell
- Checks SAF authority to key label
- Calls ICSF to retrieve protected key
  - Crytpo Express invoked for secure key
- Protected key saved in virtual storage for access requests

In this example, Read/Get…

- Reads data off disk
- Calls CPACF to decrypt
- Returns clear text to application

The application is unaware that the data has been decrypted to read

IBM Z pervasive encryption Performance

IBM Z

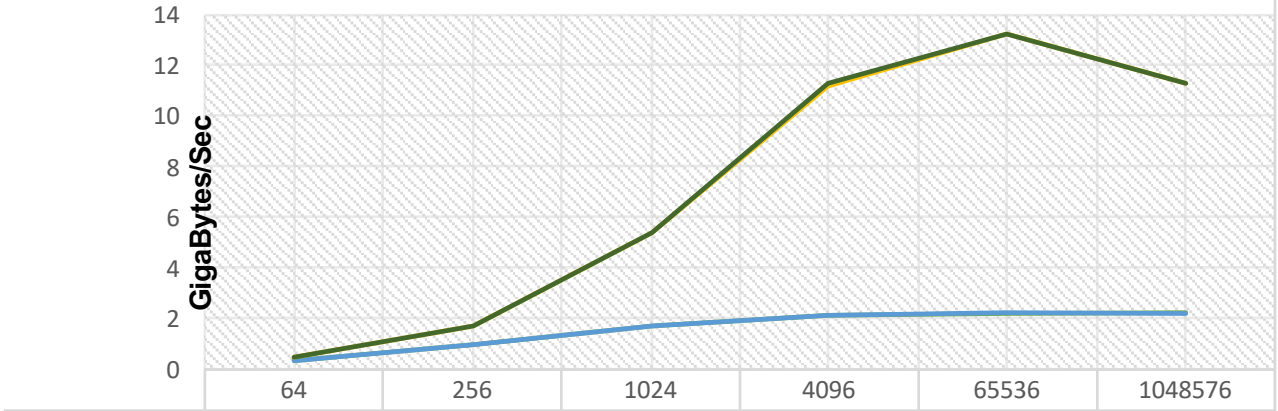
you IBM

z14 CPACF - Performance Measurement Considerations

- **CPACF on chip encryption/decryption support has been greatly optimized on the z14 machine.**
  - Early primitive testing validates up to 7X faster
- **Performance benefits for z14 are better when working with large blocks of data.**
  - Data set encryption will benefit from large block sizes
- **Encryption overhead is expected to be low for most OLTP workloads.**
  - I/O generally avoided where possible (buffer pools, caching)
  - Encryption overhead very small relative to the total workload CPU
- **Encryption cost will be higher for extremely I/O Intensive workloads**
  - Lots of I/O (lots of data to encrypt/decrypt)
  - Encryption overhead higher relative to total workload CPU

z14 CPACF - Initial Crypto Primitive Measurement Results

AES-256 XTS Protected Key



	64	256	1024	4096	65536	1048576
z13 Encrypt	0.35	0.97	1.72	2.13	2.22	2.21
z13 Decrypt	0.35	0.97	1.71	2.13	2.21	2.22
z14 Encrypt	0.49	1.77	5.44	11.25	13.26	11.26
z14 Decrypt	0.49	1.77	5.44	11.26	13.26	11.26

AES-256 XTS mode with Protected Key used by Data Set Encryption

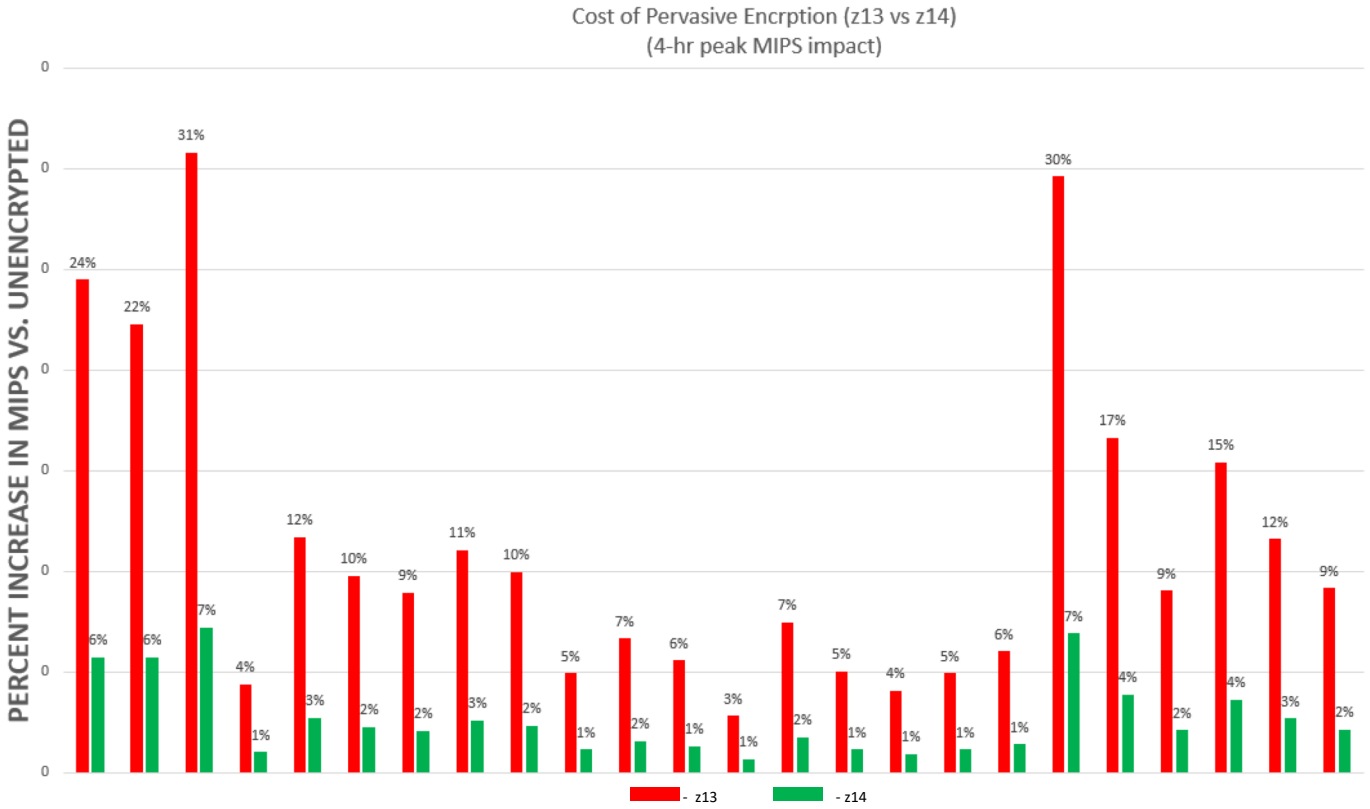
z14 measurement of primitives show much better performance for both encrypt & decrypt

Results show approaching 7x with larger blocks of data

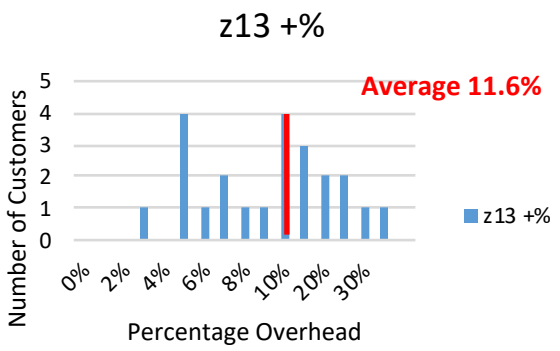
Similar results for GCM mode

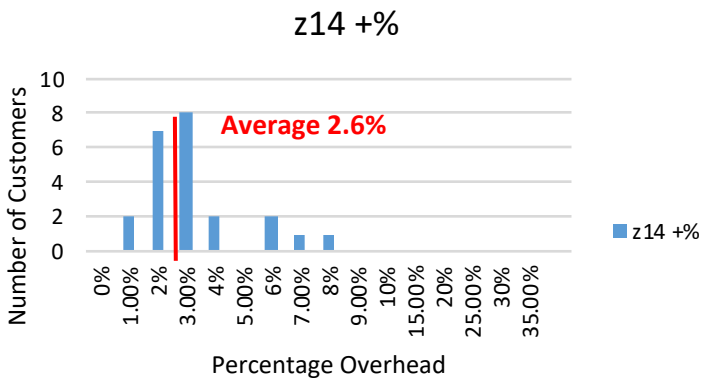
z/OS Data Set Encryption – z14 vs z13 Comparison

Percent CPU MIPS increase for enabling z/OS data set encryption



MIPS percentage overhead for data set encryption for different customers on z13 and z14



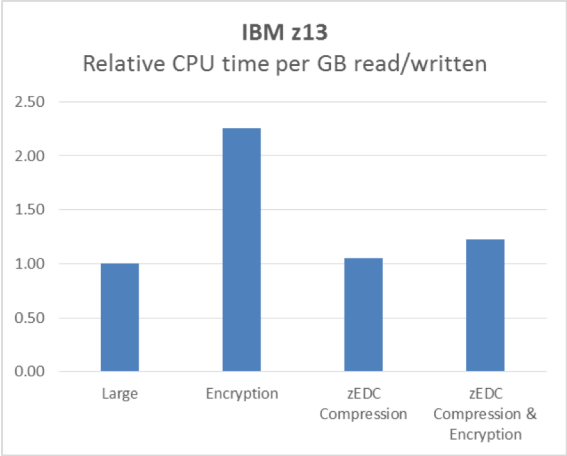


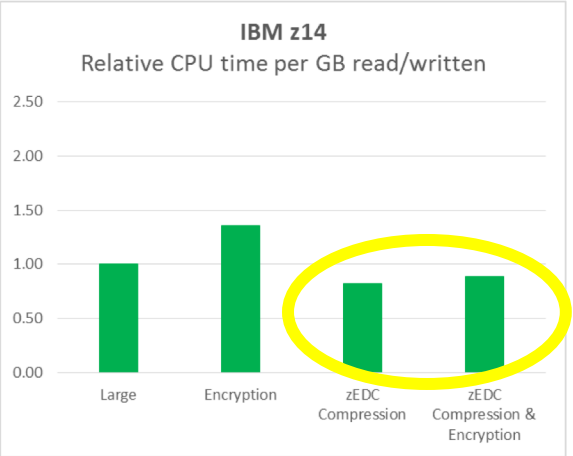
z14 overhead is generally <3-4% (average 2.6%)
z13 overhead is generally < 12 – 18% (average 11.6%)
zEC12 overhead is generally 2 times higher than z13 overhead

Use zEDC compression to enhance the efficiency of dataset encryption.

Very Highly I/O Intensive Batch

- zEDC offloads most of the CPU cost of dataset compression
- zEDC can reduce dataset size by up to 5x, reducing the size of the data to be encrypted
- On z14, zEDC is so efficient that it **reduces the CPU cost, even with encryption!**





Specifying a key label

DFP segment in RACF data set profile

- Label of an existing key in the ICSF CKDS used by access methods for encrypting/decrypting sequential and VSAM data
- Provides granularity for different key labels to be used based on RACF profiles

```
ALTDSD 'PROJECTA.DATA.*' UACC(NONE) DFP(RESOWNER(iduser1)DATAKEY(Key-Label))
```

Command Keyword	Meaning
DATAKEY(Key-Label)	Identifies the KEY LABEL in ICSF CKDS used to encrypt/decrypt the data
NODATAKEY	Removes a key label if defined to the RACF DPF segment

Key label only used for new data set create
Any subsequent change to RACF Data set profile will not affect existing data sets

JCL, Dynamic Allocation and TSO Allocate

- New keyword to be used for DASD data sets
  - **DSKEYLBL=key-label**
    - Key label of an existing key in ICSF CKDS used by access methods for encrypting/decrypting sequential and VSAM data

```
//DD1    DD
DSN=DSN1,DISP=(NEW,CATLG),DATACLAS=DSN1DATA,MGMTCLAS=DSN1MGMT,
//              STORCLAS=DSN1STOR,DSKEYLBL='LABEL.FOR.DSN1'
```

- For dynamic allocation text unit: DALDKYL
- For TSO allocate: DSKEYLBL(label-name)

DSKEYLBL is effective only if the new data set is on DASD. It is ignored for device types other than DASD, including DUMMY.

Key label only used for new data set create

Creating a new VSAM data set via IDCAMS

- New parameter on DEFINE for CLUSTER
  - **KEYLABEL=key-label**
    - Key label of an existing key in ICSF CKDS used by access methods for encrypting/decrypting sequential and VSAM data
    - Used for both cluster and any alternate index

```
DEFINE CLUSTER -
(NAME(DSN1.EXAMPLE.ESDS1) -
RECORDS(100 500) -
RECORDSIZE(250 250) -
KEYLABEL (LABEL.FOR.DSN1) -
NONINDEXED )
```

SMS Construct: Data Class

Data Class identifies key label to be used when creating a new data set.

- Key label of an existing key in ICSF CKDS used by access methods for encrypting/decrypting sequential and VSAM data

```
                                DATA CLASS ALTER                    Page 5 of 6
Command ===>

SCDS Name . . . : IBMUSER.ENCSCDS
Data Class Name : ENCRLS64

To ALTER Data Class, Specify:

  Tape Encryption Management
    Key Label 1 . . .      (1 to 64 characters or blank)

    Key Label 2 . . .

    Encoding for Key Label 1  . . . . .              (L, H or blank)
    Encoding for Key Label 2  . . . . .              (L, H or blank)

  DASD Data Set Level Encryption Management
    Data Set Key Label . . .  (1 to 64 characters or blank)
    PROTKEY.AES.SECURE.KEY.32BYTE

Use ENTER to Perform Verification; Use UP/DOWN Command to View other Panels;
Use HELP Command for Help; Use END Command to Save and Exit; CANCEL to Exit.
```

Key label only used for new data set create

Verifying data set encryption status

Identifying an encrypted data set by data set attributes

1) Volume

- **LISTVTOC** – displays volume level information
  - Data set info includes new encryption attribute under field ‘SMS.IND’

```
---------------DATA SET NAME---------------  SER NO  SEQNO  DATE.CRE  DATE.EXP
SYSPLEX.RLSENC17.KSDS01.DATA                  XP0301      1  2017.026    00.000
SMS.IND   LRECL  KEYLEN  INITIAL ALLOC  2ND ALLOC    EXTEND       LAST BLK(TTTT-
S   E   N     0            TRKS CONTIG        1
EATTR
NS
            EXTENTS  NO  LOW(C-H)   HIGH(C-H)    NO  LOW(C-H)   HIGH(C-H)    NO
```

Identifying an encrypted data set by data set attributes

2) Catalog

- **LISTCAT** – displays catalog level information
  - Data set info displays key label and Encryption flag

```
    LISTCAT ALL ENTRIES('SYSPLEX.RLSENC17.KSDS01')
CLUSTER ------- SYSPLEX.RLSENC17.KSDS01
     IN-CAT --- PDSESHR.CATALOG
     HISTORY
       DATASET-OWNER-----(NULL)      CREATION--------2017.034
       RELEASE----------------2      EXPIRATION------0000.000
     SMSDATA
       STORAGECLASS ---SXPXXS04      MANAGEMENTCLASS---(NULL)
       DATACLASS ------KSX00001      LBACKUP ---0000.000.0000
       CA-RECLAIM---------(YES)
       EATTR-------------(NULL)
       BWO STATUS------00000000      BWO TIMESTAMP---00000 00:00:
       BWO---------------(NULL)
     RLSDATA
       LOG -------------------ALL    RECOVERY REQUIRED --(NO)
       VSAM QUIESCED -------(NO)     RLS IN USE --------(YES)
       LOGSTREAMID-------------------IGWTVS.FR.LOG001
       RECOVERY TIMESTAMP LOCAL-----X'0000000000000000'
       RECOVERY TIMESTAMP GMT-------X'0000000000000000'
     ENCRYPTIONDATA
       DATA SET ENCRYPTION ---- (YES)
       DATA SET KEY LABEL ----- PROTKEY.AES.SECURE.KEY.32BYTE
       PROTECTION-PSWD-----(NULL)    RACF-------------------(NO)
```

Identifying an encrypted data set by data set attributes

3) SMS policy

- **ISMF** Data set list panel
  - Encryption flag/type

```
DGTLGP41                          DATA SET LIST                  VIEW WAS SUCCESSFUL
Command ===>                                                      Scroll ===> PAGE
                                                                  Entries 1-6 of 6
Enter Line Operators below:                                       View in Use

     LINE                                                         ENCRYPTION
     OPERATOR                 DATA SET NAME                       INDICATOR
    ---(1)----  --------------------(2)--------------------- ---(43)---
                SYSPLEX.RLSENCLP.KSDS01                          ---
                SYSPLEX.RLSENCLP.KSDS01.DATA                     YES
                SYSPLEX.RLSENCLP.KSDS01.INDEX                    YES
                SYSPLEX.RLSENCLP.KSDS02                          ---
                SYSPLEX.RLSENCLP.KSDS02.DATA                     NO
                SYSPLEX.RLSENCLP.KSDS02.INDEX                    NO
    ----------  ------  -----------  BOTTOM  OF  DATA  -----  -------  ---- ----
```

Note: In order to display the Encryption Indicator, make sure “Acquire Data from Volume – Yes” is selected in DATA SET SELCTION ENTRY PANEL

Identifying an encrypted data set by SMF

- **SMF records**
  - **SMF Type 14/15** (Sequential data sets)
    - New DASD encryption section with key label and encryption type fields

Offsets		Name	Length	Format	Description
4	4	SMF14DEF	1	binary	Flag byte. Indicators: Bit (Name) Meaning when set 0 (SMF14DSE) Data set encrypted 1 (SMF14DSEB) The system honors user requested access method to bypass decryption on reads
					2-7 Reserved
5	5		1	binary	Flag byte. Reserved
6	6	SMF14DET	2	binary	Encryption type
8	8	SMF14DKL	64	EBCDIC	DASD data set key labels

Identifying an encrypted data set by SMF

- **SMF records**
  - **SMF Type 62** (VSAM data sets)
    - New DASD encryption information with key label and encryption type fields

12	C	SMF62DEF	1	binary	Fourth ACB MACRF flag byte: Bit (Name) Meaning when set 0 (SMF62DSENC) DASD data set encrypted 2-7 Reserved
13	D	SMF62DET	2	binary	Encryption type
15	F	SMF62DKL	64	EBCDIC	DASD data set key label

Identifying an encrypted data set by DCOLLECT

- **DFSMS Data Collection Facility**
  - **DCOLLECT** – system/data level information
    - Data class definition record Type 'DC': New key label field

Offset	Type	Length	Name	Description
302(X'12E')	BITSTRING	1	DDCSPECC	ADDITIONAL SPECIFICATION FLAGS
	.....			
	...1		DDCFKLBL	DASD Data Set Key label specified
	.....			
470(X'1D6')	CHARACTER	66	DDCDKYBL	DASD Data Set Key label
470(X'1D6')	SIGNED	2	DDCDKLBL	DASD Data Set Key Label length
472(X'1D8')	CHARACTER	64	DDCDKLBN	DASD Data Set Key Label name

Identifying an encrypted data set by DCOLLECT

- **DFSMS Data Collection Facility**
  - **DCOLLECT** – system/data level information
    - Data set info record Type 'D' : New key label field

Offset	Type	Length	Name	Description
	.....			
386 (X'182')	CHARACTER	66	DCDENCR	ENCRYPTION INFORMATION
386 (X'182')	UNASSIGNED	2	DCDTYPE	ENCRYPTION TYPE
388 (X'184')	CHARACTER	64	DCDKLBL	ENCRYPTION KEY LABEL

Identifying an encrypted data set by DCOLLECT

- **DFSMS Data Collection Facility**
  - **DCOLLECT** – system/data level information
    - HSM migration/backup record: Encryption flag

Offset	Type	Length	Name	Description
	.....			
184 (B8)	BITSTRING	1	UMFLAG2	INFORMATION FLAG 2
	.....			
	1		UMENCRP	IF SET TO 1, DATA SET IS ENCRYPTED
	.....			
185 (B9)	BITSTRING	1	UBFLAG3	INFORMATION FLAG 3
	.....			
	..1 .... 1... xxx		UBENCRP *	ONLY VALID WHEN UBF_RETAIN_SPCD IS SET TO 1. WHEN SET TO 1, DATA SET IS ENCRYPTED RESERVED

Identifying encryption SW support by Programming Interfaces

- **DFSMS Features Area (DFA)**
  - **DFAENCRYPT** New flag to indicate DFSMS data set encryption SW installed

60 (3C)	Bit string	4	DFAFEAT9	Features byte 9
	1...		DFAJ3AA	JES3_ALLOC_ASSIST ENABLED
	.1..		DFAMEMUX	Reserved
	…1. ….		DFAPDSEG	PDSE Generation support is installed
	…1 ….		DFAZEDCCMP	zEDC Compression support is installed
	…. xxx.			
	.... …1		DFAENCRYPT	Data set encryption support is installed
…				

Identifying an encrypted data set by Programming Interfaces

1) Catalog

- CSI (catalog search interface)
  - Key label, Encryption flag/type, Encryption cell

Catalog Field Names

Table 1 shows the catalog field names.

Table 1. Catalog Field Names

Rep	Type	Length	Name	Description
.....				
no	Binary	1	ENCRYPTF	The field name for the encryption flag. • X'00' - Not encrypted. • X'01' - Encrypted.
no	Fixed	2	ENCRYPTT	A 2 byte integer for the encryption type. It is initialized to x'0100'. If the data set is not encrypted, hex 'FFFF' is returned. Encryption type is intended for possible future types of encryption.
no	Character	96	ENCRYPTA	All of the encryption fields as one field. It returns 96 bytes of information as formatted in the encryption cell: • 2 bytes for the encryption type • 64-byte key label • 8 bytes for the saved ICV (first half) • 1 byte for the encryption mode • 16 bytes for a verification value • 5 bytes reserved • If the data set is not encrypted, 96 bytes of hex 'FF's are returned.
.....				
no	Character	64	KEYLABEL	The field name for key label and the data returned is 64 characters in length. If the data set is not encrypted, 64 bytes of hex 'FF's are returned.
.....				

Identifying an encrypted data set by Programming Interfaces

2) BSAM/QSAM macro

- **ISITMGD** – returns attributes related to sequential data sets
  - Encryption flag **ISMENCRP** ON if the DASD data set is encrypted by the access methods.