IBM eserver zSeries and Total Storage Update (zSTSU)

Data Encryption for DB2

IBM Advanced Technical Support

Trademarks

The following are trademarks of the International Business Machines Corporation in the United States and/or other countries.

AIX*
AIX/ESA*
C/MVS
C/370
CICS*
CICS/ESA*
CICS/MVS*
COBOL/370

Database 2
DB2*
DB2 Connect
developerWorks*
DFSMS/MVS*
DFSMSdfp
DFSMSdss
DFSMShsm

e-business logo*
e(logo)server*
ESCON
FICON*
ibm.com*
IBMLink
MQSeries*
Multiprise*

MVS
MVS/DFP
MVS/ESA
OS/2*
OS/2 WARP*
OS/390*Parallel Sysplex*
Processor Resource/Systems Manager
PR/SM
RACF*

Resource Link
RMF
S/390*
S/390 Parallel Enterprise Server
WebSphere*
z/Architecture
z/OS*
z/VM*
zSeries*

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

Linux is a registered trademark of Linus Torvalds
Penguin (Tux) compliments of Larry Ewing
Java and all Java-related trademarks and logos are trademarks of Sun Microsystems, Inc., in the United States and other countries
UNIX is a registered trademark of The Open Group in the United States and other countries.
Microsoft, Windows and Windows NT are registered trademarks of Microsoft Corporation.
SET and Secure Electronic Transaction are trademarks owned by SET Secure Electronic Transaction LLC.
Intel, Intel Inside (logos), MMX and Pentium are trademarks of Intel Corporation in the United States, other countries, or both.

MasterCard is a registered trademark of MasterCard International
RSA BSAFE is a registered trademark of RSA Data Security
RSA is a registered trademark of RSA Inc.
Visa is a register trademark of Visa international

* All other products may be trademarks or registered trademarks of their respective companies.

Notes:

Performance is in Internal Throughput Rate (ITR) ratio based on measurements and projections using standard IBM benchmarks in a controlled environment. The actual throughput that any user will experience will vary depending upon 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 throughput improvements equivalent to the performance ratios stated here.

IBM hardware products are manufactured from new parts, or new and serviceable used parts. Regardless, our warranty terms apply.

All customer examples cited or described in this presentation are presented as illustrations of the manner in which some customers have used IBM products and the results they may have achieved. Actual environmental costs and performance characteristics will vary depending on individual customer configurations and conditions.

This publication was produced in the United States. IBM may not offer the products, services or features discussed in this document in other countries, and the information may be subject to change without notice. Consult your local IBM business contact for information on the product or services available in your area.

All statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only.

Information about non-IBM products is obtained from the manufacturers of those products or their published announcements. IBM has not tested those products and cannot confirm the 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.

Prices subject to change without notice. Contact your IBM representative or Business Partner for the most current pricing in your geography.

IBM Data Encryption for IMS and DB2 V1.1

- A solution offering targeting requirements driven by
  - Patriot Act
  - SEC 17a-4
  - Sarbanes-Oxley
  - Health Insurance Portability and Accountability Act (HIPAA) of 1996
  - Privacy Act
  - Graham-Leach-Bliley Act of 1999
- Not perfect, any solution has challenges in this environment
  - Performance overhead
  - Key management
  - Application changes
- Solution offered
  - PRPQ, program number 5799-GWD
  - Implemented using an IMS or DB2 exit
    - DECENC01 for IMS, Segment Edit/Compression exit specified in DBD COMPRTN parameter
    - DECENC00 for DB2, EDITPROC exit specified in EDITPROC clause of the SQL CREATE TABLE statement

IBM Data Encryption for IMS and DB2 V1.1

- Data is encrypted at the IMS segment and DB2 table level
  - Any restrictions are based on product specifics
    - DB2 cannot have DB2 indices encrypted
- Can have different encryption exits for different segments or tables
- Log records and image copies of data are also encrypted
- Data encryption is done via calls to ICSF from the exit which must be customized
  - ICSF callable services, CSNBENC and CSNBDEC, used
  - Requires DATA key defined in ICSF CKDS for each exit requiring unique key value
- Documentation
  - IBM Data Encryption for IMS and DB2 Databases User 's Guide
    - V1R1, SC18-7336-01

Facts

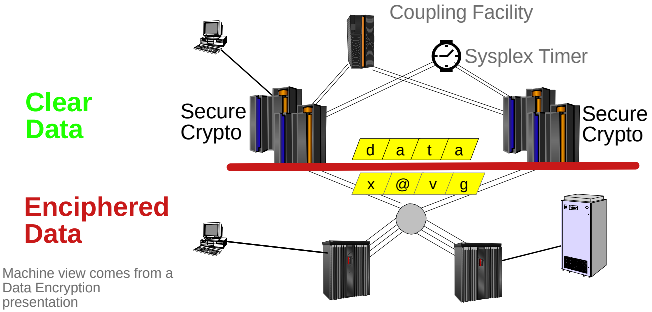
- Provides user-customizable, pre-coded exits for encrypting IMS and DB2 data
- Exploits zSeries and S/390 Crypto Hardware features, which results in low overhead encryption/decryption
  - CCF or PCIXCC required
  - Hardware provides speed, however, usage causes impact
- Uses DES/TDES
  - Recognition of need for AES
- Works at and is customizable at the IMS segment level or DB2 table level
- Usage causes impact??
  - Encrypting 4 bytes is just as much overhead as encrypting 80 or 1K bytes

Facts

- Column-level encryption was considered, however,
  - The cost to encrypt one column is roughly equivalent to the cost to encrypt the entire row
  - If this offering supported column level encryption, then two calls to the crypto hardware would be needed, thus doubling the performance overhead when compared to row-level encryption
- Size of the row has very little impact on performance overhead
- Therefore, encryption of the entire row provides lowest performance overhead possible
- So, what performance impact to expect
  - Roughly think 4 to 5 times the current overhead
  - Yeah, 400 - 500%, but that is less than using other crypto solutions
  - Will performance improvements come?
    - Probably
  - Solution performance degrades with z990/890

Protection Provided by Offering

- Data encryption on disk
  - Data on channel is encrypted (protects against channel/network sniffers)
  - Data in buffers and DB2 indexes is not encrypted
- Existing authorization controls accessing this data are unaffected
  - Assumption made that access (through the DBMS or direct access) invokes the DBMS data exits



Machine view comes from a Data Encryption presentation

Note:

- ✓ Data in transactions not covered by solution
- ✓ Data on disk is
- ✓ Enciphered Data to be processed on another machine must have crypto capability and access to same key value
- ✓ Testing in DB2 and IMS environments ensure compatibility with all interfaces

IMS Exit Processing

- IMS Encryption
  - Program passes a segment REPL, ISRT, or LOAD request to IMS control region
  - IMS determines, using the DBD, that a Segment Edit/Compression exit is required
  - IMS loads and calls the exit, passing it the unencrypted segment
  - Exit invokes ICSF services
  - IMS puts the encrypted segment into the database
- IMS Decryption
  - IMS application program passes segment GET request to the IMS control region
  - IMS determines, from the DBD, that a Segment Edit/Compression exit is required
  - IMS loads and calls the exit, passing it the encrypted segment
  - Exit invokes ICSF services
  - IMS passes the decrypted segment back to the application

IMS Exit Processing . . .

- IMS restrictions that apply
  - You cannot both encrypt and compress the database. (Note, however, that in DB2 this is allowed.)
  - An IMS segment can only be associated with one Segment Edit/Compression exit. If your IMS segment is already associated with a Segment Edit/Compression exit and you wish to implement this product, then you must code an alternative solution for your existing exit.
  - HIDAM index databases cannot be encrypted (the IMS DBD COMPRTN parameter doesn’t allow index databases to be specified on the Segment Edit/Compression exit)

DB2 Exit Processing

- DB2 Encryption
  - DB2 application program passes a row to DB2.
  - DB2 determines, by presence of EDITPROC on the table, exit is required
  - DB2 loads and calls the exit, passing it the unencrypted row
  - Exit invokes ICSF services
  - DB2 puts the encrypted row into the table
- DB2 Decryption
  - DB2 application program requests data from DB2
  - DB2 determines, by presence of EDITPROC on the table, exit is required
  - DB2 loads and calls the exit, passing it the unencrypted row
  - Exit invokes ICSF services
  - DB2 passes the decrypted row back to the application

DB2 Exit Processing . . .

- DB2 restrictions that apply
  - A DB2 table can only specify one EDITPROC exit. If your DB2 table already has an EDITPROC exit specified and you wish to implement this product, then you must code an alternative solution for your existing EDITPROC exit.
  - Indexes cannot be encrypted (the EDITPROC function doesn't support encryption of indexes).
  - Tables with ROWIDs or LOBs cannot be encrypted
- DB2 Considerations
  - In DB2, you can both encrypt and compress data using DB2's hardware compression. However, compression takes place after encryption, which greatly compromises the effectiveness of the compression. Because of this, you may want to disable DB2's hardware compression on objects that are encrypted
  - Presence of an EDITPROC can impact the size of a row in DB2
  - DB2 EDITPROC exits do not encrypt indexes

Pre-Requisites for Use

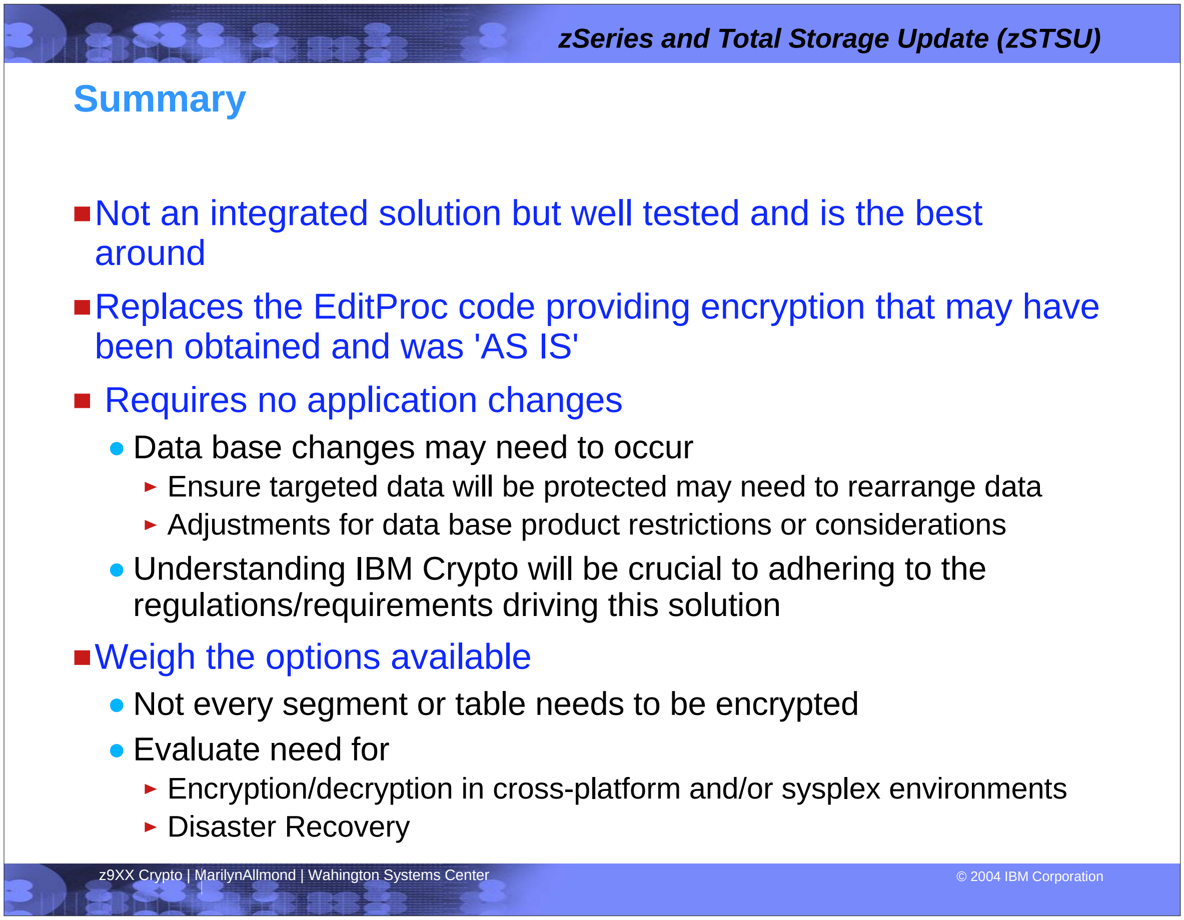
- Required Crypto hardware must be activated
  - CCF and PCIXCC
    - Configuration loaded
    - LPAR associations must be made
    - Master Keys must be loaded
- Required Software
  - IMS Version 6 or higher, and/or DB2 for OS/390 Version 6 or higher
  - ICSF must be activated
    - Base element of z/OS and OS/390
    - Key(s) to be used for data protection must be defined
      - ✓ Security issue!!
      - ✓ Do not look for step-by-step directions
      - ✓ Should understand this process in order to explain decisions to auditors
      - ✓ Should understand crypto concepts and IBM Crypto

Generic Installation Steps

- Set up and validate crypto hardware
  - Validate? If from ICSF menu, Utility option for Random Number works, ICSF and secure crypto hardware works
- Generate and then store (in the CKDS) a triple DES encryption key for use
  - Use ICSF Key Generation Utility Program, KGUP
  - Read ICSF Administrator's Guide
- Build the IMS or DB2 user exit, specifying the key name defined in step above
- Back up your data
- Unload your data
- Create/install the exit
- Reload the data, during which process the data is encrypted
- Validate your output

Specific Exit Installation Notes

- IMS
  - Provides sample job DECIMSJB in PDS smphlq.SDECSAMP
- DB2
  - Provides sample job DECDB2JB in PDS smphlq.SDECSAMP
- Generic to either database
  - Replace the yyyyyyyyyy (at the end of the job) with encryption key label built by the security analyst
  - Also, provides an alternate in the form of a panel interface to create the user exit
    - ex ’smphlq.SDECCEXE(DECENC04)”smphlq ’



Summary

- Not an integrated solution but well tested and is the best around
- Replaces the EditProc code providing encryption that may have been obtained and was 'AS IS'
- Requires no application changes
  - Data base changes may need to occur
    - Ensure targeted data will be protected may need to rearrange data
    - Adjustments for data base product restrictions or considerations
  - Understanding IBM Crypto will be crucial to adhering to the regulations/requirements driving this solution
- Weigh the options available
  - Not every segment or table needs to be encrypted
  - Evaluate need for
    - Encryption/decryption in cross-platform and/or sysplex environments
    - Disaster Recovery