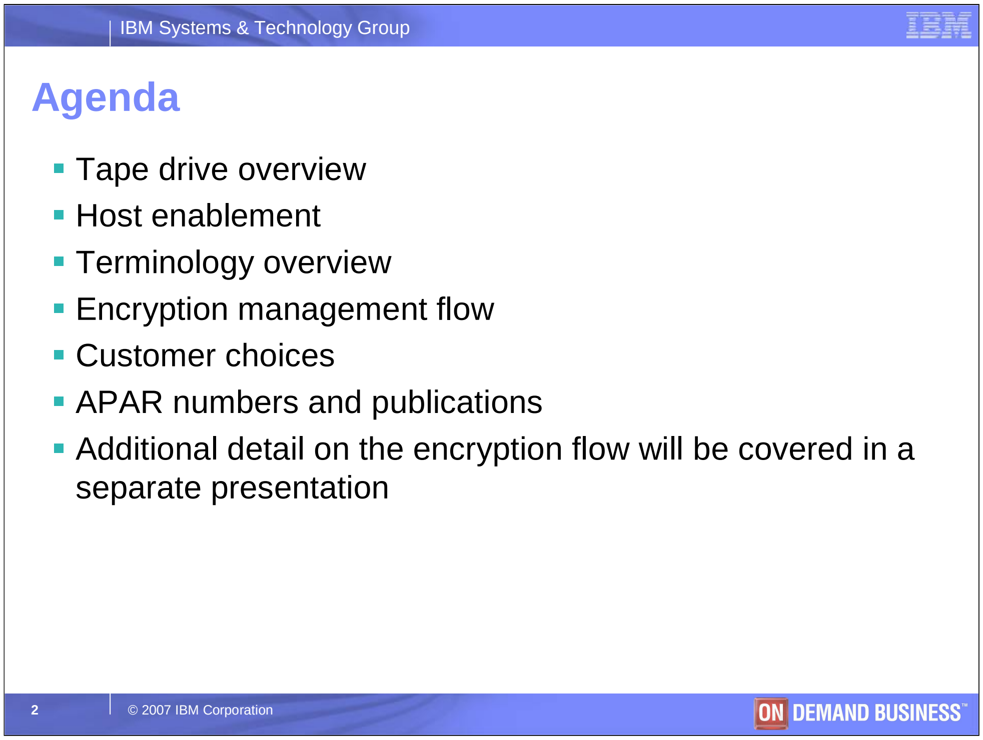
IBM Systems & Technology Group

z/OS & DFSMS

TS1120 Tape Encryption Support

Overview

This presentation provides an overview of the z/OS and DFSMS encryption support for the IBM System Storage TS1120 Tape Drive also referred to as the 3592 Model E05.

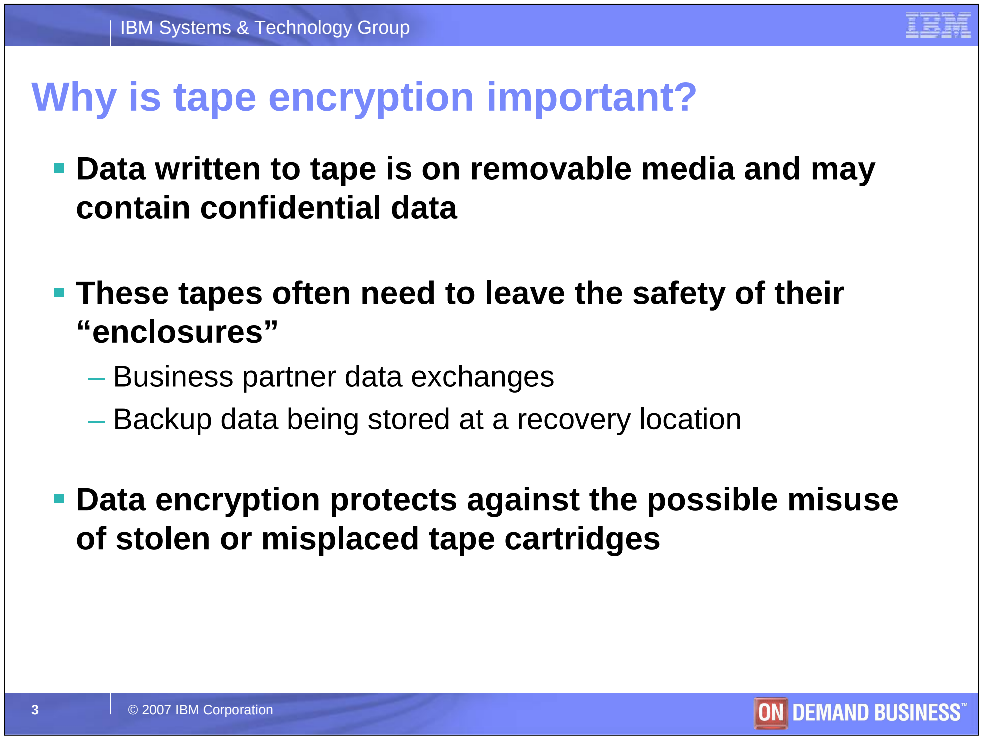


Agenda

- Tape drive overview
- Host enablement
- Terminology overview
- Encryption management flow
- Customer choices
- APAR numbers and publications
- Additional detail on the encryption flow will be covered in a separate presentation

ON DEMAND BUSINESS™

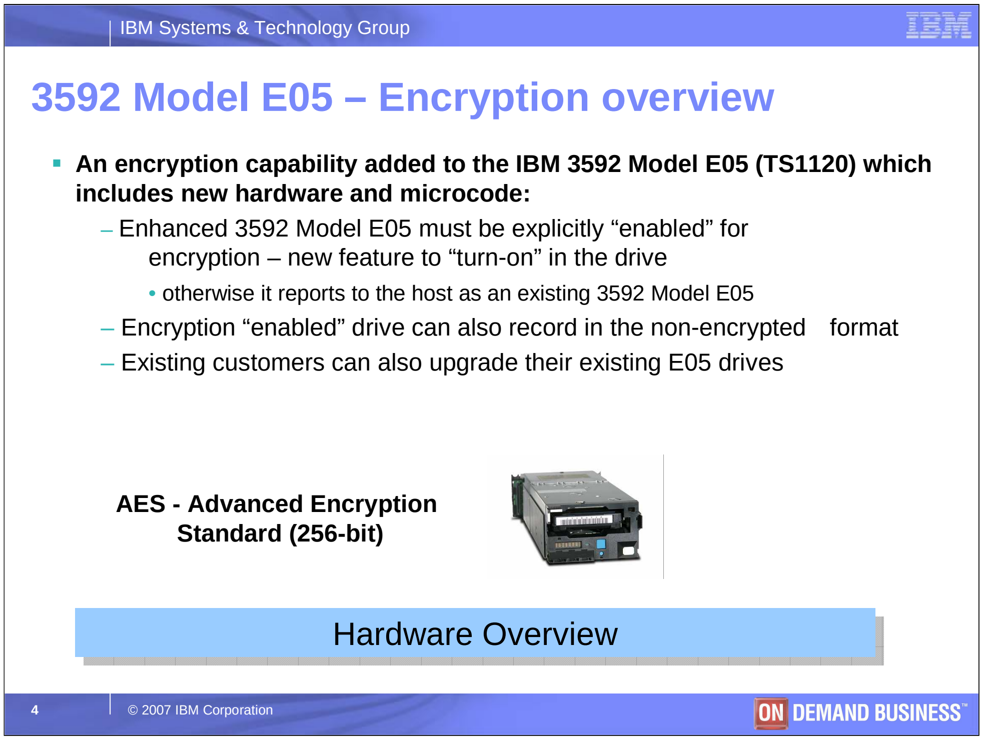
This module starts with an overview of the encryption support that is provided in the tape drive. Then it will discuss how encryption is requested at the host, followed by a discussion of the terminology used in the encryption solution. It will also describe the encryption management flow between the drive and the encryption key manager and the role of the new proxy interface in IOS. Then it will discuss the items you will need to consider when implementing the solution, followed by the list of applicable APAR numbers and publications. Another presentation will go into more detail about the encryption flow and terminology.



Why is tape encryption important?

- **Data written to tape is on removable media and may contain confidential data**
- **These tapes often need to leave the safety of their "enclosures"**
  - Business partner data exchanges
  - Backup data being stored at a recovery location
- **Data encryption protects against the possible misuse of stolen or misplaced tape cartridges**

So first you may ask, why is tape encryption even important?   Since data directed to tape is written on media that can be easily removed from its "enclosure", care must be taken with the data, especially if it contains confidential information. These tapes often need to leave the safety of their "enclosure". For example, for business partner data exchanges or for backup data going off-site to a recovery location.   Encrypting your tape data protects against the possible misuse of stolen or misplaced tape cartridges.



IBM Systems & Technology Group

3592 Model E05 – Encryption overview

- **An encryption capability added to the IBM 3592 Model E05 (TS1120) which includes new hardware and microcode:**
  - Enhanced 3592 Model E05 must be explicitly “enabled” for encryption – new feature to “turn-on” in the drive
    - otherwise it reports to the host as an existing 3592 Model E05
  - Encryption “enabled” drive can also record in the non-encrypted format
  - Existing customers can also upgrade their existing E05 drives

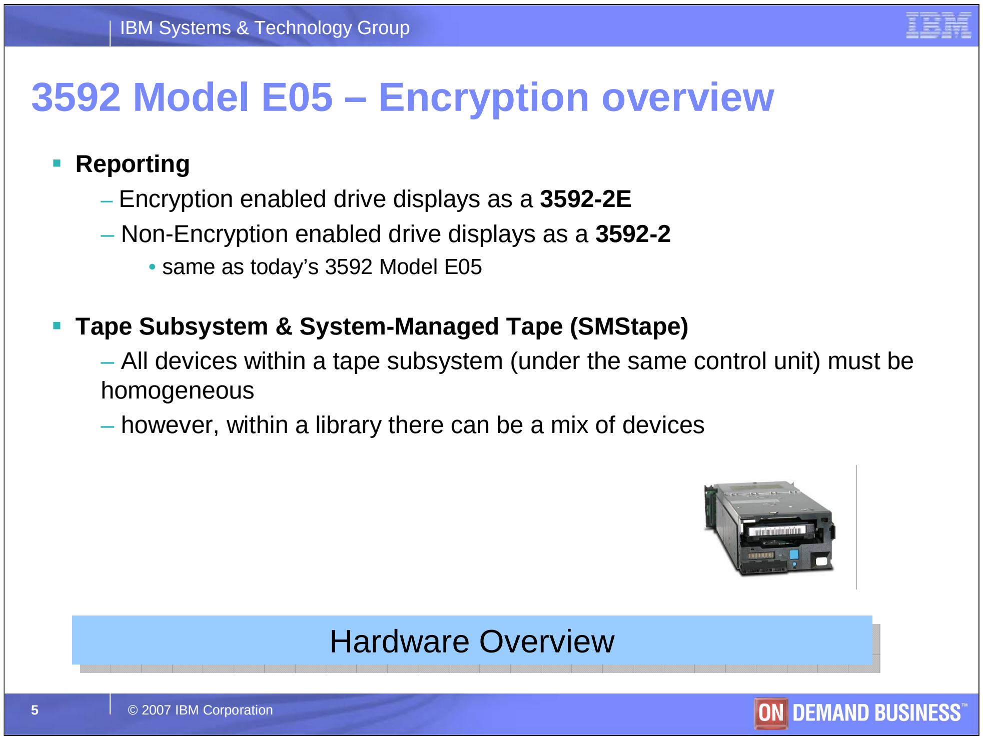
AES - Advanced Encryption Standard (256-bit)

Hardware Overview

4 © 2007 IBM Corporation

ON DEMAND BUSINESS™

A new encryption capability has been added to the IBM System Storage TS1120 Tape Drive, also referred to as the 3592 Model E05. This new encryption capability in the drive includes both hardware and microcode changes. Even though each new TS1120 delivered after the encryption support became available is capable of tape encryption, the encryption feature in the tape drive must be explicitly enabled in the drive for the encryption support to be used. Once the drive has the encryption feature enabled, it can then record in the new encryption format and the non-encryption formats. How the drive is instructed to record in which recording format will be discussed later in this presentation. For existing TS1120 tape drives, there is also an upgrade path available that will enable you to add the new encryption capability into the drive. Encryption of the data then takes place at the drive using the Advanced Encryption Standard (AES) with 256-bit data keys.

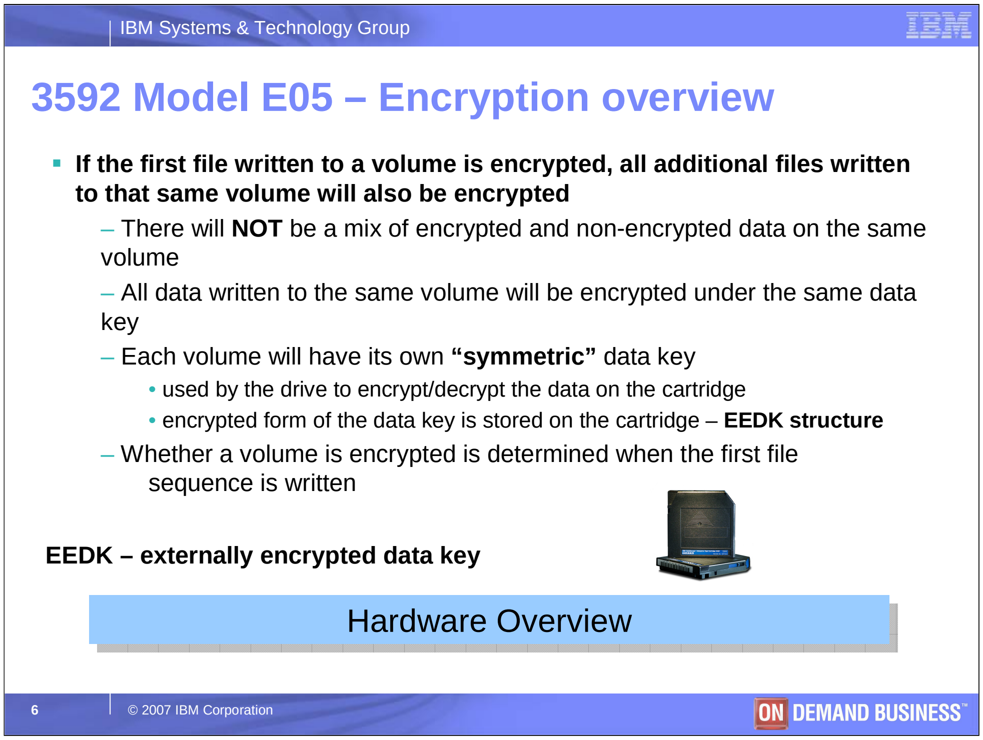


3592 Model E05 – Encryption overview

- **Reporting**
  - Encryption enabled drive displays as a **3592-2E**
  - Non-Encryption enabled drive displays as a **3592-2**
    - same as today's 3592 Model E05
- **Tape Subsystem & System-Managed Tape (SMStape)**
  - All devices within a tape subsystem (under the same control unit) must be homogeneous
  - however, within a library there can be a mix of devices

Hardware Overview

If the 3592 Model E05 has the encryption feature enabled, it will display at the host as a 3592-2E and if the feature is not installed or enabled, it will display as a 3592-2. The "2" at the end indicates that this is the 2nd generation 3592 tape drive. Important to note is that with our system-managed tape support (SMStape), all devices under the same tape control unit must be homogenous. So if the encryption feature is enabled in one of the 3592 Model E05 drives, it needs to be enabled in all of the drives under that same control unit. This way all devices under the same control unit can handle the same request. However, within a tape library there can be a mix of devices with different capabilities as long as there is more than one control unit supporting that mix.



IBM Systems & Technology Group

3592 Model E05 – Encryption overview

- **If the first file written to a volume is encrypted, all additional files written to that same volume will also be encrypted**
  - There will **NOT** be a mix of encrypted and non-encrypted data on the same volume
  - All data written to the same volume will be encrypted under the same data key
  - Each volume will have its own **"symmetric"** data key
    - used by the drive to encrypt/decrypt the data on the cartridge
    - encrypted form of the data key is stored on the cartridge – **EEDK structure**
  - Whether a volume is encrypted is determined when the first file sequence is written

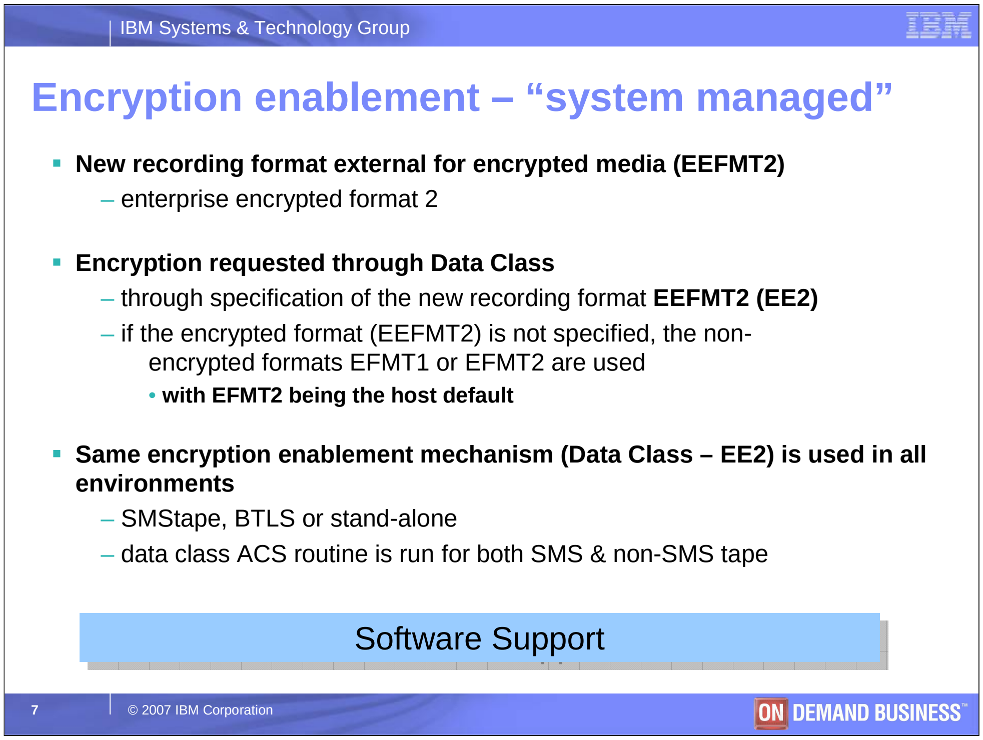
EEDK – externally encrypted data key

Hardware Overview

6 © 2007 IBM Corporation

ON DEMAND BUSINESS™

Whether a tape volume is encrypted or not is determined when the first file sequence is written to the tape. If the first file sequence is encrypted, all additional files written to that same volume will also be encrypted. A mix of encrypted and non-encrypted data is not supported on the same volume. All data written to the same volume will be encrypted under the same data key with each volume having its own symmetric data key. The symmetric data key is used by the drive to encrypt and decrypt the data that is stored on the tape cartridge. An encrypted form of this data key is also stored on the tape cartridge in a structure referred to as the externally encrypted data key (EEDK).



IBM Systems & Technology Group

Encryption enablement – “system managed”

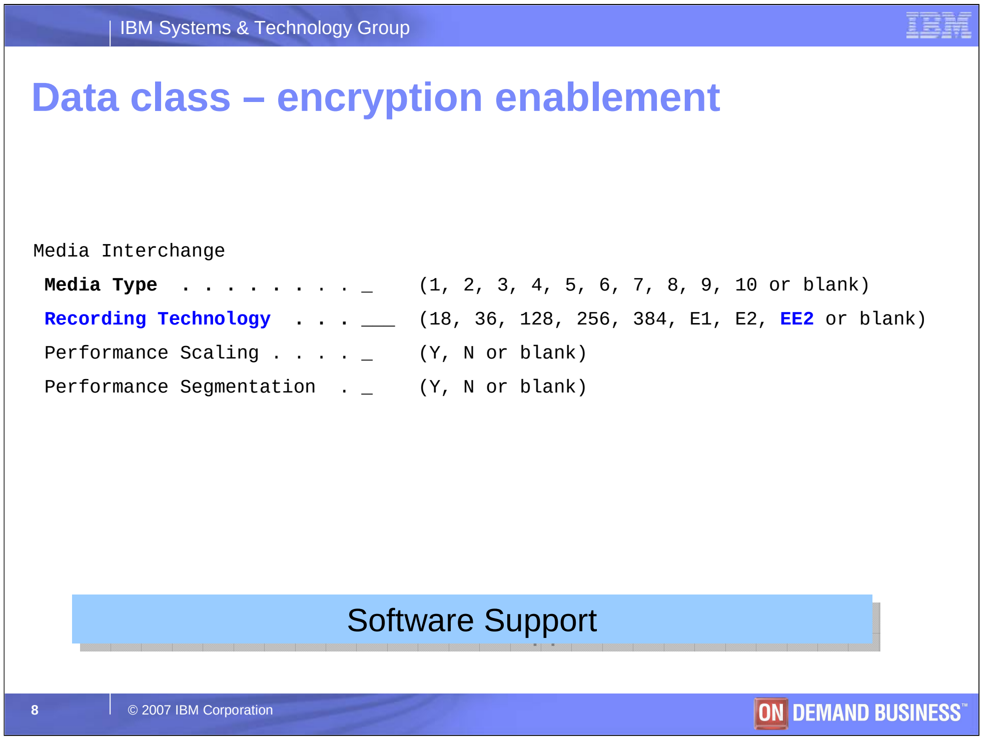
- **New recording format external for encrypted media (EEFMT2)**
  - enterprise encrypted format 2
- **Encryption requested through Data Class**
  - through specification of the new recording format **EEFMT2 (EE2)**
  - if the encrypted format (EEFMT2) is not specified, the non-encrypted formats EFMT1 or EFMT2 are used
    - **with EFMT2 being the host default**
- **Same encryption enablement mechanism (Data Class – EE2) is used in all environments**
  - SMStape, BTLS or stand-alone
  - data class ACS routine is run for both SMS & non-SMS tape

Software Support

7 © 2007 IBM Corporation

ON DEMAND BUSINESS™

In support of the tape encryption solution, there is a new recording format EEFMT2 for enterprise encrypted format 2. Encryption is requested through the SMS data class ACS routine through specification of the new recording format EEFMT2 or EE2. If the encryption format is not selected, the non-encrypted formats EFMT1 or EFMT2 can be selected through data class with EFMT2 being the host default. Since data class is applicable with or without system-managed tape (SMStape), the same enablement mechanism is used to request encryption in any environment (SMStape, BTLS or stand-alone).



IBM Systems & Technology Group

Data class – encryption enablement

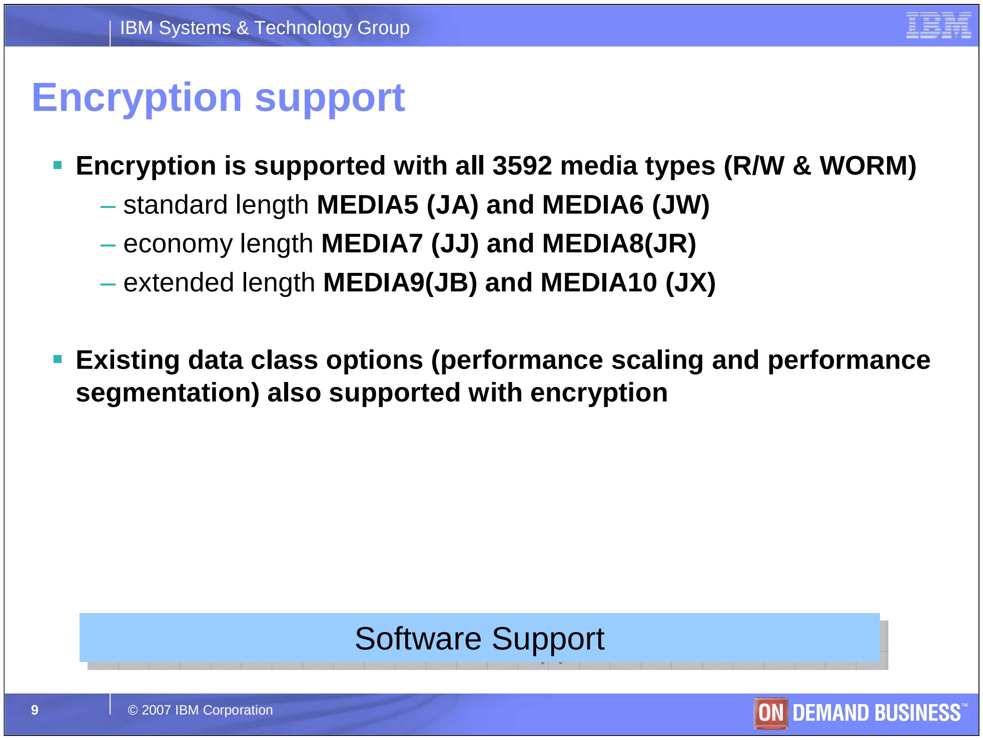
```
Media Interchange
 Media Type  . . . . . . . . . _     (1, 2, 3, 4, 5, 6, 7, 8, 9, 10 or blank)
 Recording Technology   . . . ___    (18, 36, 128, 256, 384, E1, E2, EE2 or blank)
 Performance Scaling . . . . . _     (Y, N or blank)
 Performance Segmentation  . . _     (Y, N or blank)
```

Software Support

8

ON DEMAND BUSINESS™

The data class “media interchange” section shows the new encryption format EEFMT2 (EE2) that can be specified in the recording technology parameter to request encryption.   The non-encryption formats EFMT1 (E1) or EFMT2 (E2) can also be selected.   If a recording technology is not selected and a 3592 Model E05 is allocated, EFMT2 (E2) is the host default.   A specific media type can also be specified in the “media type” parameter.



IBM Systems & Technology Group

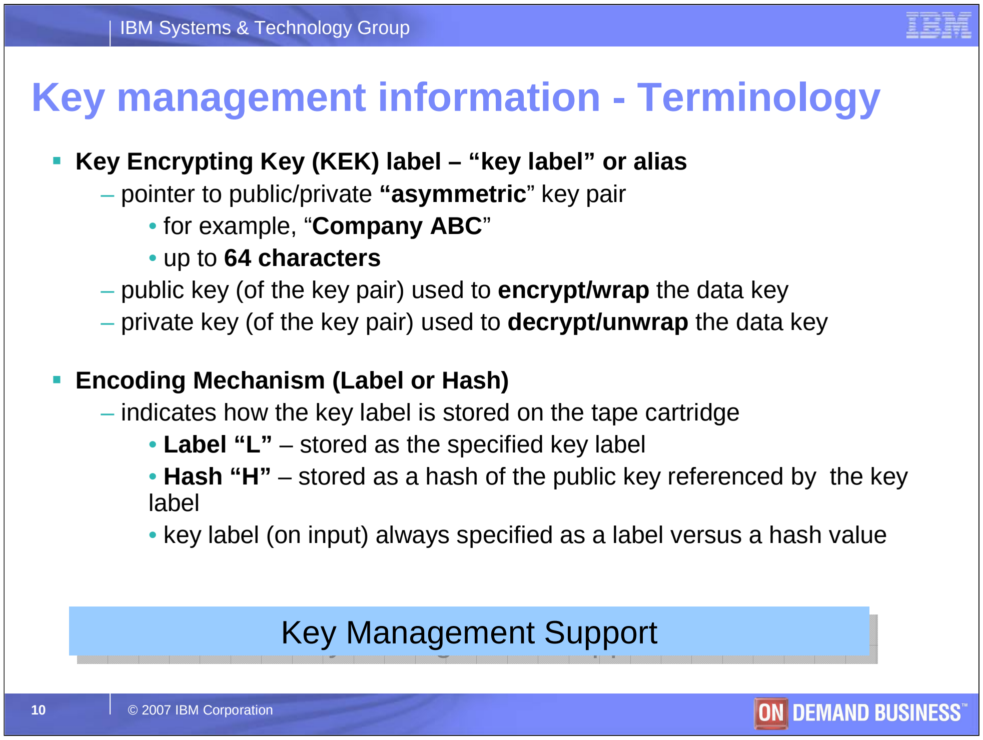
Encryption support

- **Encryption is supported with all 3592 media types (R/W & WORM)**
  - standard length **MEDIA5 (JA) and MEDIA6 (JW)**
  - economy length **MEDIA7 (JJ) and MEDIA8(JR)**
  - extended length **MEDIA9(JB) and MEDIA10 (JX)**
- **Existing data class options (performance scaling and performance segmentation) also supported with encryption**

Software Support

9 © 2007 IBM Corporation

Encryption is supported with all existing 3592 media types: standard, economy and extended length; both with read/write (R/W) and with write-once-read-many (WORM) cartridges. Encryption can also be used with the existing data class parameters for performance scaling and performance segmentation. These are additional options that are available with media types: MEDIA5 and MEDIA9.



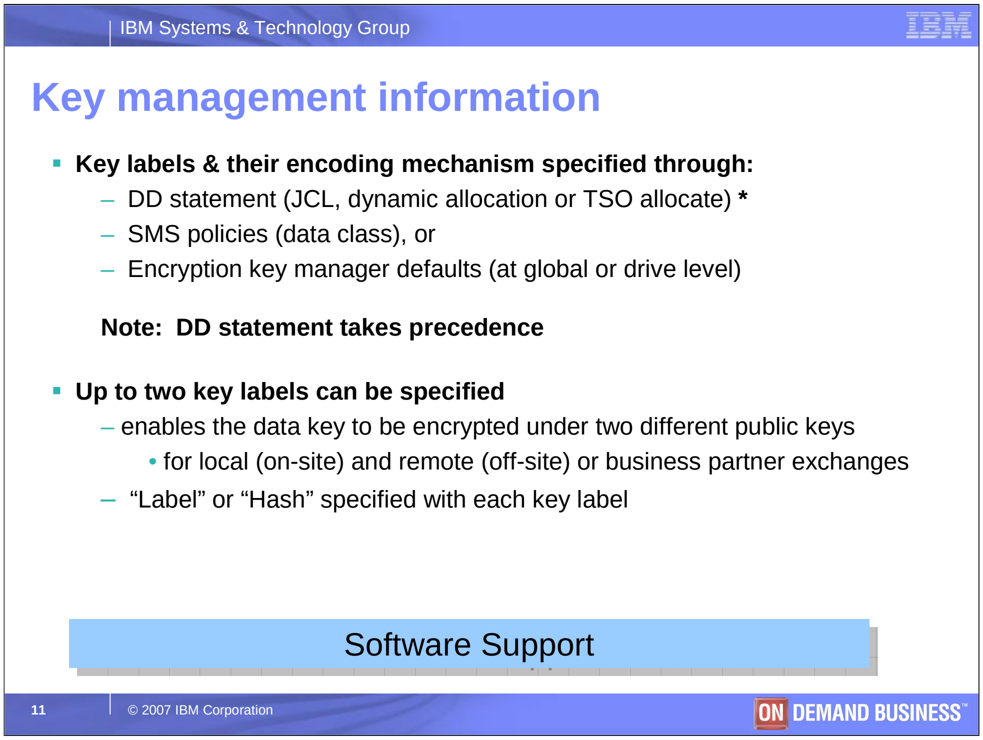
Key management information - Terminology

- **Key Encrypting Key (KEK) label – “key label” or alias**
  - pointer to public/private **“asymmetric**” key pair
    - for example, “**Company ABC**”
    - up to **64 characters**
  - public key (of the key pair) used to **encrypt/wrap** the data key
  - private key (of the key pair) used to **decrypt/unwrap** the data key

- **Encoding Mechanism (Label or Hash)**
  - indicates how the key label is stored on the tape cartridge
    - **Label “L”** – stored as the specified key label
    - **Hash “H”** – stored as a hash of the public key referenced by the key label
    - key label (on input) always specified as a label versus a hash value

Key Management Support

Lets discuss some of the terminology that is applicable to this support. A key encrypting key (KEK) is an asymmetric key pair that is used to encrypt and decrypt the data key. The public key of this key pair is used to encrypt the data key and the private key of this key pair is used to decrypt the data key. The data key is what the tape drive uses to encrypt and decrypt the data on the tape cartridge. So the key encrypting key (KEK) is used to protect or “wrap” the data key associated with a tape cartridge. A key label or alias is what is used to externally reference the key encrypting key. So for example, the key encrypting key may be referenced by key label “Company ABC”. An encoding mechanism of “label” or “hash” indicates how the specified key label is stored on the tape cartridge in the EEDK structure. An encoding mechanism of “label’ stores the key label as specified and an encoding mechanism of “hash” stores the key label as a hash of the public key. The key label is always entered as a label, however the encoding mechanism provides instructions to the encryption key manager on how the key label is then stored on the tape cartridge. The next couple of slides provide additional detail on why you would choose one encoding mechanism over the other.



Key management information

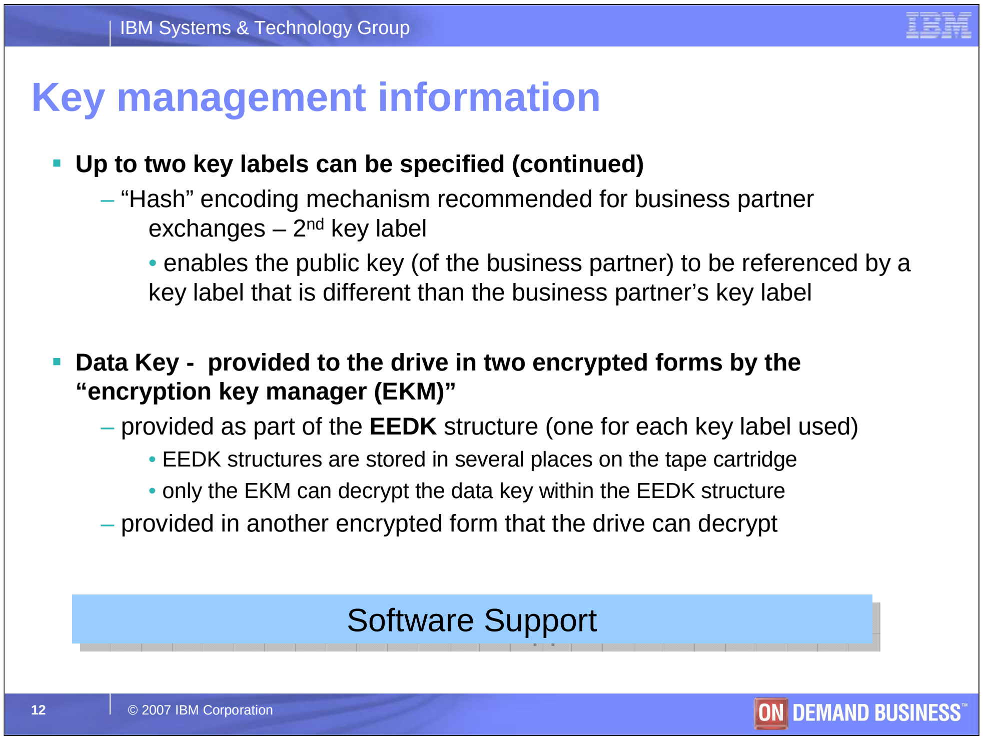
- **Key labels & their encoding mechanism specified through:**
  - DD statement (JCL, dynamic allocation or TSO allocate) *****
  - SMS policies (data class), or
  - Encryption key manager defaults (at global or drive level)

  Note: DD statement takes precedence

- **Up to two key labels can be specified**
  - enables the data key to be encrypted under two different public keys
    - for local (on-site) and remote (off-site) or business partner exchanges
  - "Label" or "Hash" specified with each key label

Software Support

The key labels and their encoding mechanism can be specified in several ways. They can be specified through the DD statement: job control language (JCL), dynamic allocation or TSO allocate. They can also be specified through the SMS data class policy and lastly they can be specified through encryption key manager defaults at the global or drive level. Any key labels specified through the DD statement take precedence. Even though the key labels can be specified in several ways, encryption itself can only be requested through data class by specifying the encryption recording format EEFMT2 (EE2). Up to two key labels can be specified enabling the data key to be encrypted with two different public keys. One of the key labels may be used for local (on-site) usage and the other may be used for remote (off-site) or business partner data exchanges. "Label" or "Hash" can be specified with each of the key labels.

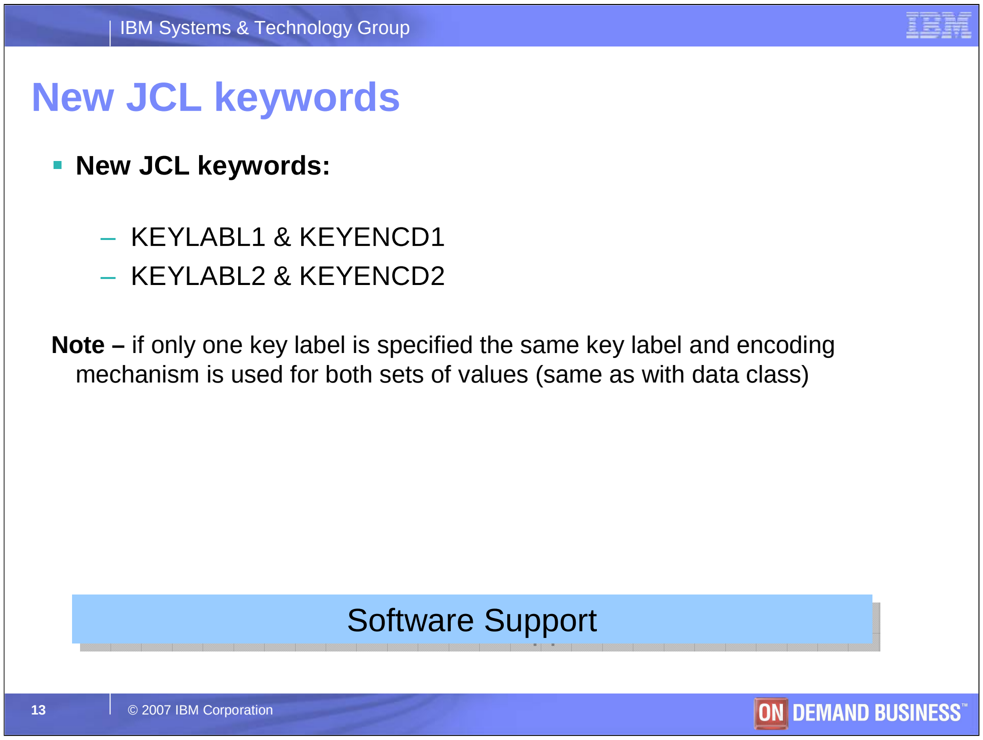


Key management information

- **Up to two key labels can be specified (continued)**
  - “Hash” encoding mechanism recommended for business partner exchanges – 2nd key label
    - enables the public key (of the business partner) to be referenced by a key label that is different than the business partner’s key label
- **Data Key - provided to the drive in two encrypted forms by the “encryption key manager (EKM)”**
  - provided as part of the **EEDK** structure (one for each key label used)
    - EEDK structures are stored in several places on the tape cartridge
    - only the EKM can decrypt the data key within the EEDK structure
  - provided in another encrypted form that the drive can decrypt

Software Support

The “hash” encoding mechanism is recommended for business partner data exchanges. This enables the public key of the business partner to be referenced by a key label (in your key store) that is different than the business partner’s key label (in their key store). The data key that is used to encrypt and decrypt the actual data on the tape cartridge is provided to the drive in two encrypted forms by the encryption key manager (EKM). It is provided to the drive as part of the EEDK structure that is stored on the tape cartridge for subsequent usage. It is also provided to the drive in another encrypted form that the drive can decrypt and use for current operations.



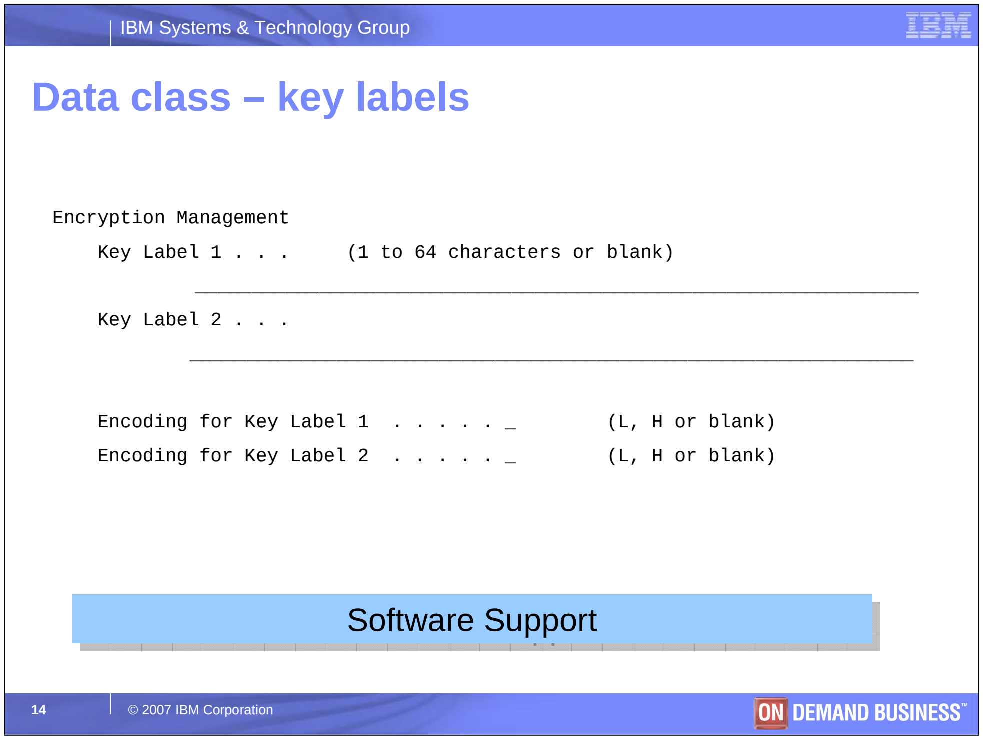
New JCL keywords

- **New JCL keywords:**
  - KEYLABL1 & KEYENCD1
  - KEYLABL2 & KEYENCD2

Note – if only one key label is specified the same key label and encoding mechanism is used for both sets of values (same as with data class)

Software Support

There are new job control language (JCL) keywords that can be specified on the DD statement for the key label (KEYLABL1 and KEYLABL2) and its encoding mechanism (KEYENCD1 and KEYENCD2). If only one key label is specified the same key label and encoding mechanism is sent to the drive for both key labels.   This behavior is the same regardless of where the key labels are specified at the host; the DD statement or through data class.

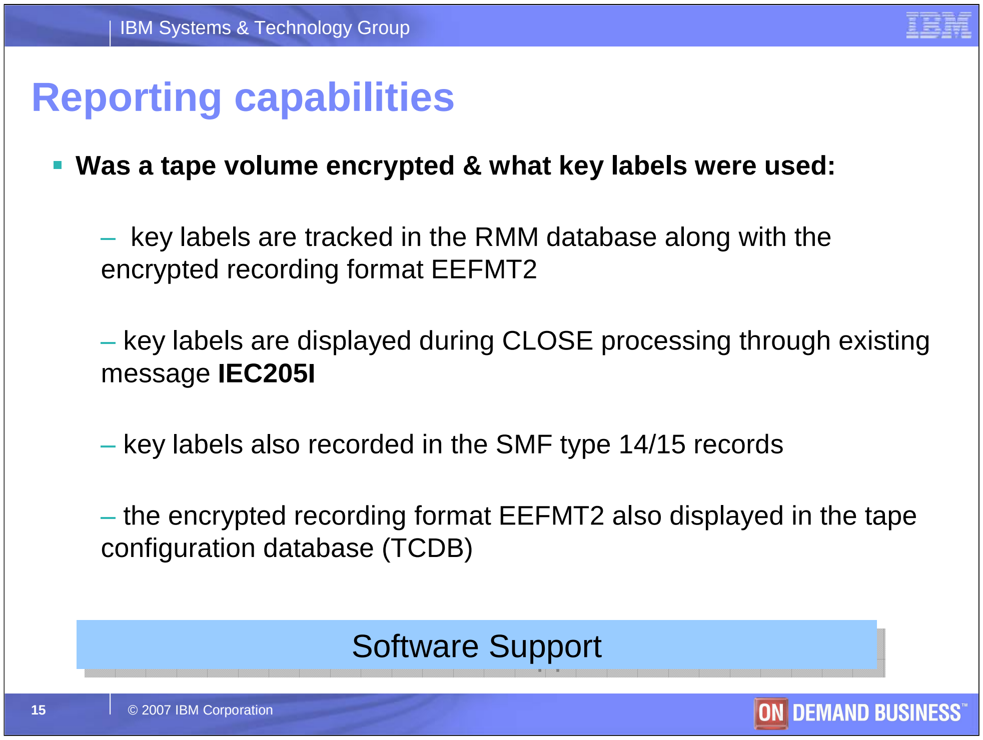


Data class – key labels

```
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)
```

Software Support

This data class panel shows the new “encryption management” section that was added for key label and encoding mechanism specification. The key label and encoding mechanism provides instructions to the encryption key manager (EKM) on how the specified key label will be encoded in the EEDK structure that is written to the tape. The key labels themselves will always be entered as a label versus a hash value.

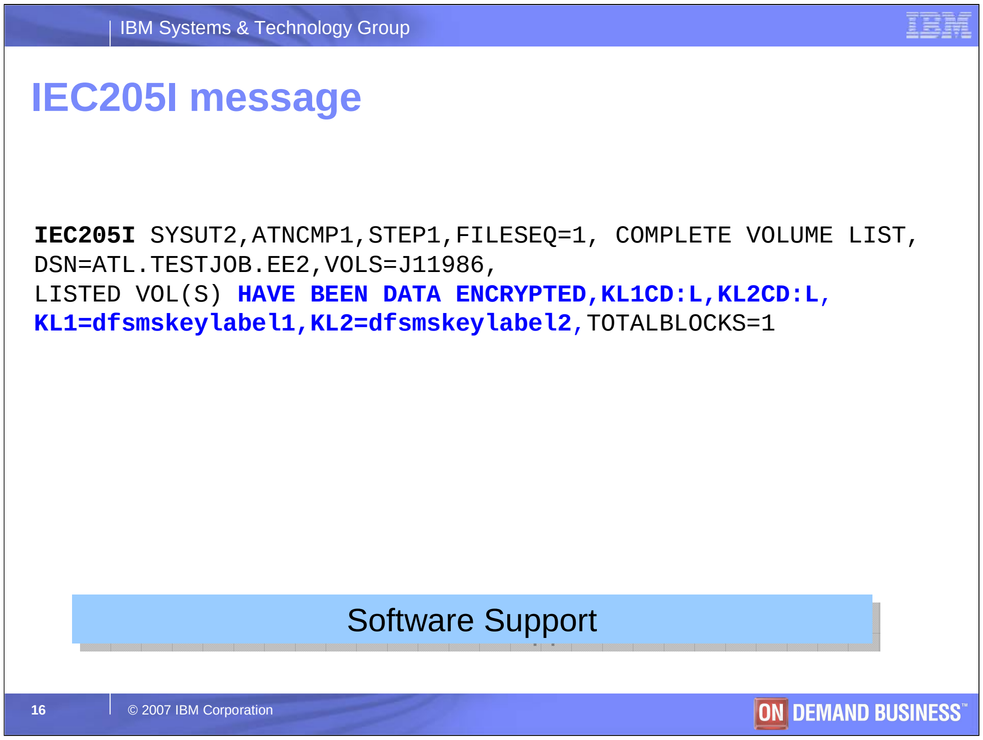


Reporting capabilities

- **Was a tape volume encrypted & what key labels were used:**
  - key labels are tracked in the RMM database along with the encrypted recording format EEFMT2
  - key labels are displayed during CLOSE processing through existing message **IEC205I**
  - key labels also recorded in the SMF type 14/15 records
  - the encrypted recording format EEFMT2 also displayed in the tape configuration database (TCDB)

Software Support

Now you know how to request encryption in data class through specification of the encryption format EEFMT2 (EE2) and you know how to specify key labels and their encoding mechanism. How do you know whether the data on the tape cartridge was encrypted and what key labels were actually used? If you are using DFSMSrmm as your tape management system, the key labels and their encoding mechanism are tracked in their database along with recording format EEFMT2. The key labels and their encoding mechanism are also displayed during tape dataset close processing through existing message IEC205I. Lastly, the key labels and their encoding mechanism are also tracked in the existing SMF 14/15 record through a new subtype 7. And if you are using system-managed tape (SMStape), the recording format EEFMT2 is also tracked in the tape configuration database (TCDB).

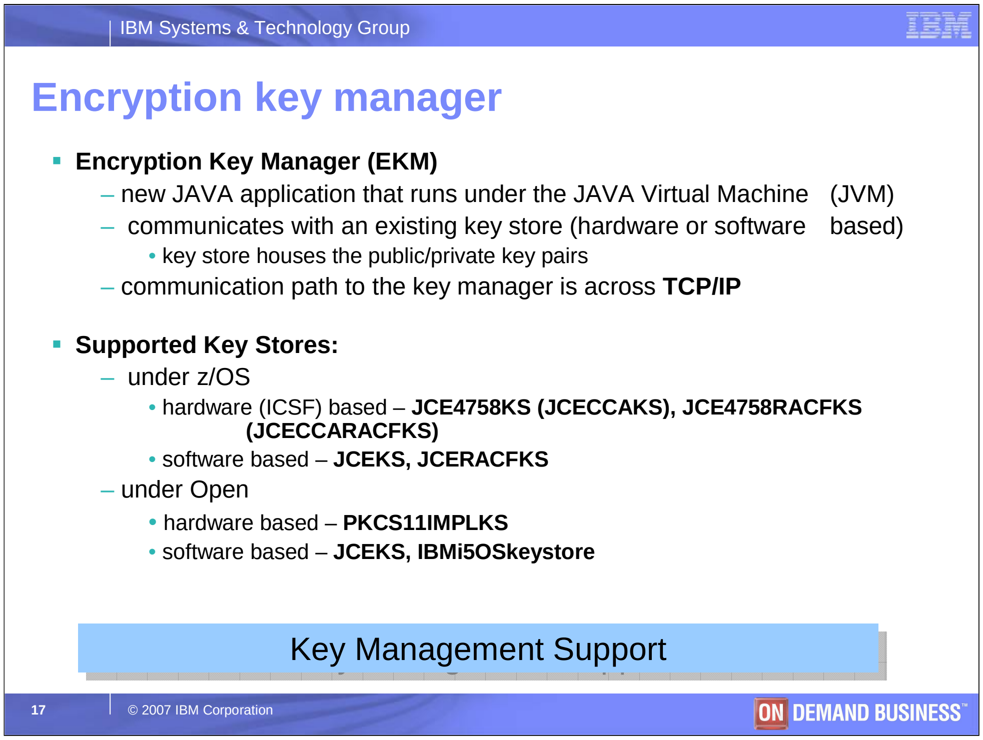


IEC205I message

```
IEC205I SYSUT2,ATNCMP1,STEP1,FILESEQ=1, COMPLETE VOLUME LIST,
DSN=ATL.TESTJOB.EE2,VOLS=J11986,
LISTED VOL(S) HAVE BEEN DATA ENCRYPTED,KL1CD:L,KL2CD:L,
KL1=dfsmskeylabel1,KL2=dfsmskeylabel2,TOTALBLOCKS=1
```

Software Support

The IEC205I message issued during tape dataset close processing displays whether the data was encrypted and if it was, it also displays what key labels and encoding mechanism were used.   The example above shows that “dfsmskeylabel1” and dfsmskeylabel2” were the key labels used and an encoding mechanism of “label” was used for each key label.   This message can be used to verify that the end result was as you expected.



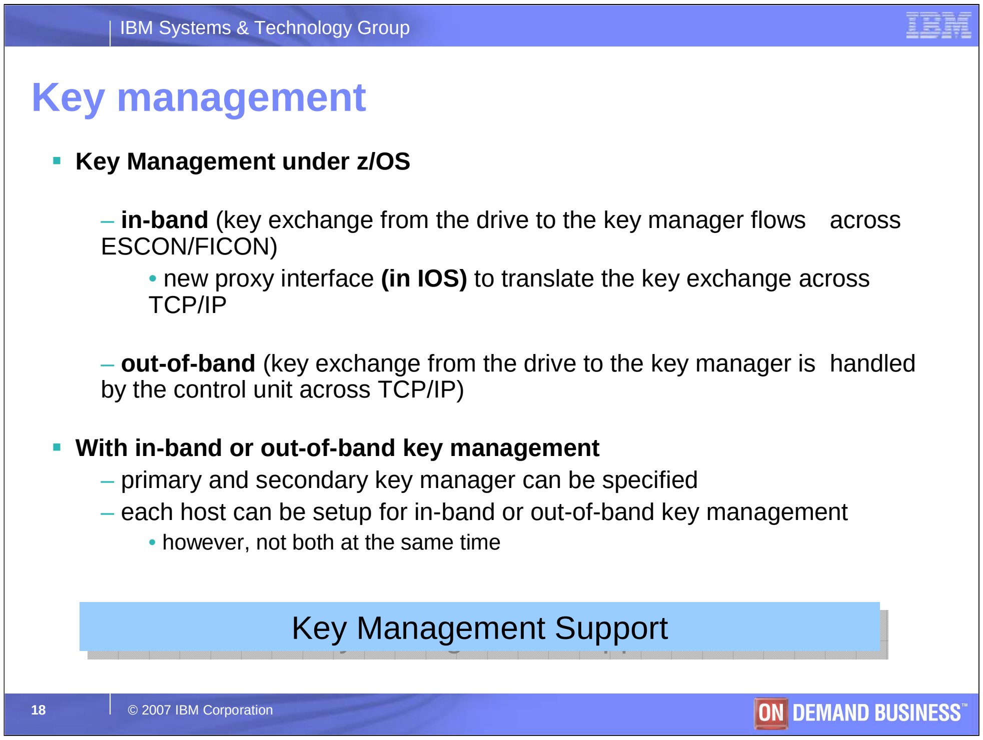
IBM Systems & Technology Group

Encryption key manager

- **Encryption Key Manager (EKM)**
  - – new JAVA application that runs under the JAVA Virtual Machine (JVM)
  - – communicates with an existing key store (hardware or software based)
    - key store houses the public/private key pairs
  - – communication path to the key manager is across **TCP/IP**
- **Supported Key Stores:**
  - – under z/OS
    - hardware (ICSF) based – **JCE4758KS (JCECCAKS), JCE4758RACFKS (JCECCARACFKS)**
    - software based – **JCEKS, JCERACFKS**
  - – under Open
    - hardware based – **PKCS11IMPLKS**
    - software based – **JCEKS, IBMi5OSkeystore**

Key Management Support

Previously we've referenced the encryption key manager (EKM). What role does the encryption key manager play? The EKM is a new JAVA application that runs under the JAVA virtual machine (JVM) on any IBM server: i, p, x or z and also Windows, Sun Solaris or HP-UX. It communicates with a key store that can be hardware or software based to access the public/private key pairs defined in the key store. When the tape drive needs a data key to encrypt or decrypt data, it communicates with the encryption key manager. The communication path to the EKM, regardless of where it resides is across TCP/IP. However, as we'll discuss later, there can be an in-band or an out-of-band path from the drive to the EKM. If the key manager resides on z/OS there are two software based and two hardware based key stores that are supported. The hardware based key store on z/OS uses the Integrated Cryptographic Services Facility (ICSF) and can also use RACF. RACF can also be used as one of the software based key stores. Then if the key manager resides on an Open system platform, a software or hardware based key store is also supported.



IBM Systems & Technology Group

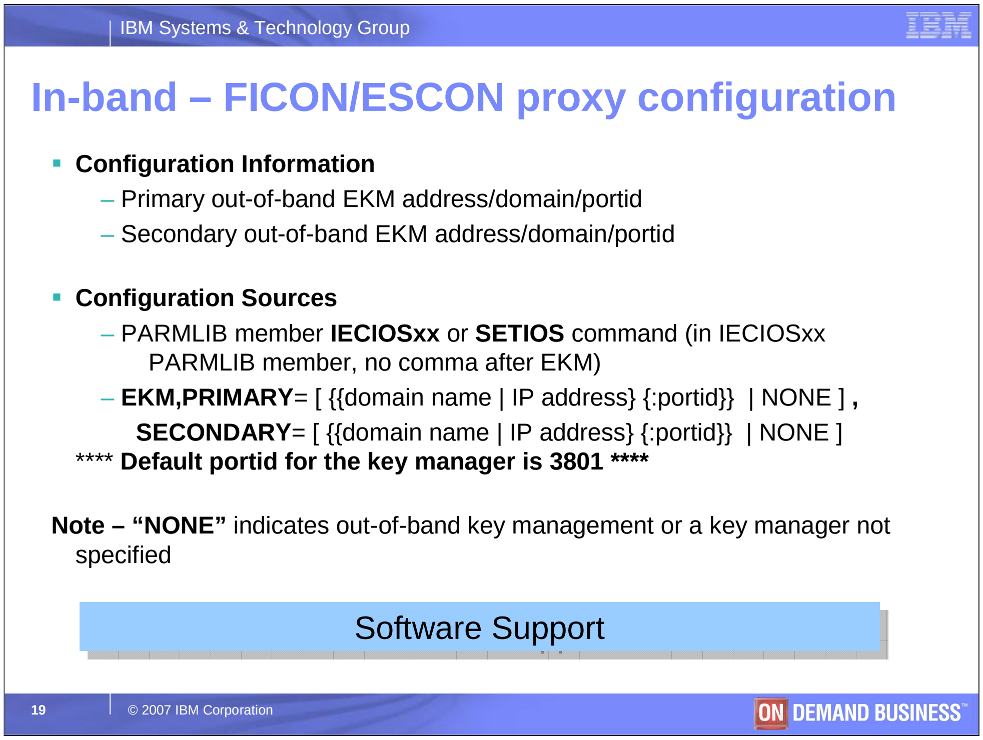
Key management

- **Key Management under z/OS**
  - **in-band** (key exchange from the drive to the key manager flows across ESCON/FICON)
    - new proxy interface **(in IOS)** to translate the key exchange across TCP/IP
  - **out-of-band** (key exchange from the drive to the key manager is handled by the control unit across TCP/IP)
- **With in-band or out-of-band key management**
  - primary and secondary key manager can be specified
  - each host can be setup for in-band or out-of-band key management
    - however, not both at the same time

Key Management Support

18

Communication with the encryption key manager (EKM) itself is always across TCP/IP. For in-band key management, the key exchange between the drive and the EKM flows across ESCON/FICON with a new proxy interface in IOS (I/O Supervisor component of z/OS). The IOS proxy translats the key exchange across TCP/IP to the EKM. For out-of-band key management, the key exchange between the drive and the EKM is handled by the control unit across TCP/IP. The expectation is that System z hosts, like z/OS, that support an in-band flow to the EKM, will be setup for in-band communication. For System z hosts that do not have proxy support (VM, VSE and TPF), they will be setup to use the control unit's out-of-band support. With either in-band or out-of-band key management, a primary and a secondary key manager is supported so if communication cannot be made to the primary key manager, the alternate or secondary is tried. Each z/OS host can be setup for in-band or out-of-band key management, but not both at the same time. When using both a primary and a secondary key manager, it is critical that each key manager has access to the same public/private key pairs.



In-band – FICON/ESCON proxy configuration

- **Configuration Information**
  - Primary out-of-band EKM address/domain/portid
  - Secondary out-of-band EKM address/domain/portid

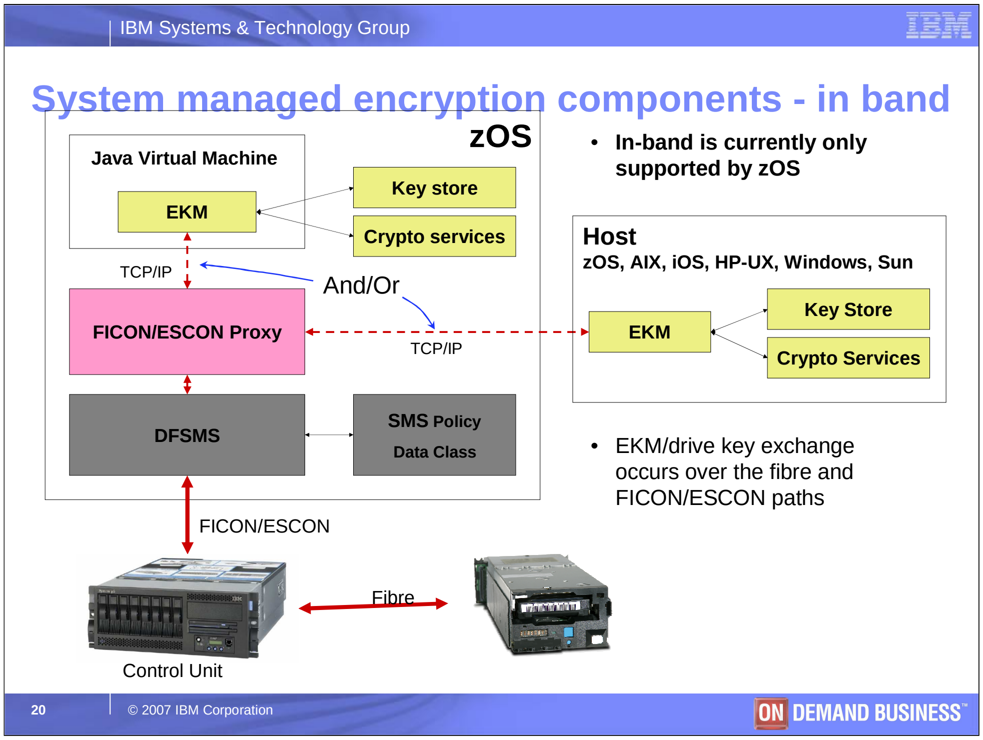
- **Configuration Sources**
  - PARMLIB member **IECIOSxx** or **SETIOS** command (in IECIOSxx PARMLIB member, no comma after EKM)
  - **EKM,PRIMARY**= [{{domain name | IP address} {:portid}} | NONE] **,**
    SECONDARY= [{{domain name | IP address} {:portid}} | NONE]

**** **Default portid for the key manager is 3801 ****

Note – "NONE" indicates out-of-band key management or a key manager not specified

Software Support

For in-band communication with the encryption key manager, the TCP/IP address of the key manager can be specified in the IOS PARMLIB member IECIOSxx, or it can be established using the SETIOS command. There is a new "EKM" parameter that takes as input a PRIMARY and a SECONDARY EKM. A key manager of "NONE", which is the default, indicates that either a key manager has not been established for in-band communication or the request is for out-of-band key management. By default, the portid for the EKM is 3801, however this can be changed to another value.



IBM Systems & Technology Group

System managed encryption components - in band

zOS

Java Virtual Machine

EKM

Key store

Crypto services

TCP/IP

And/Or

FICON/ESCON Proxy

TCP/IP

DFSMS

SMS Policy

Data Class

Host

zOS, AIX, iOS, HP-UX, Windows, Sun

EKM

Key Store

Crypto Services

- **In-band is currently only supported by zOS**
- EKM/drive key exchange occurs over the fibre and FICON/ESCON paths

FICON/ESCON

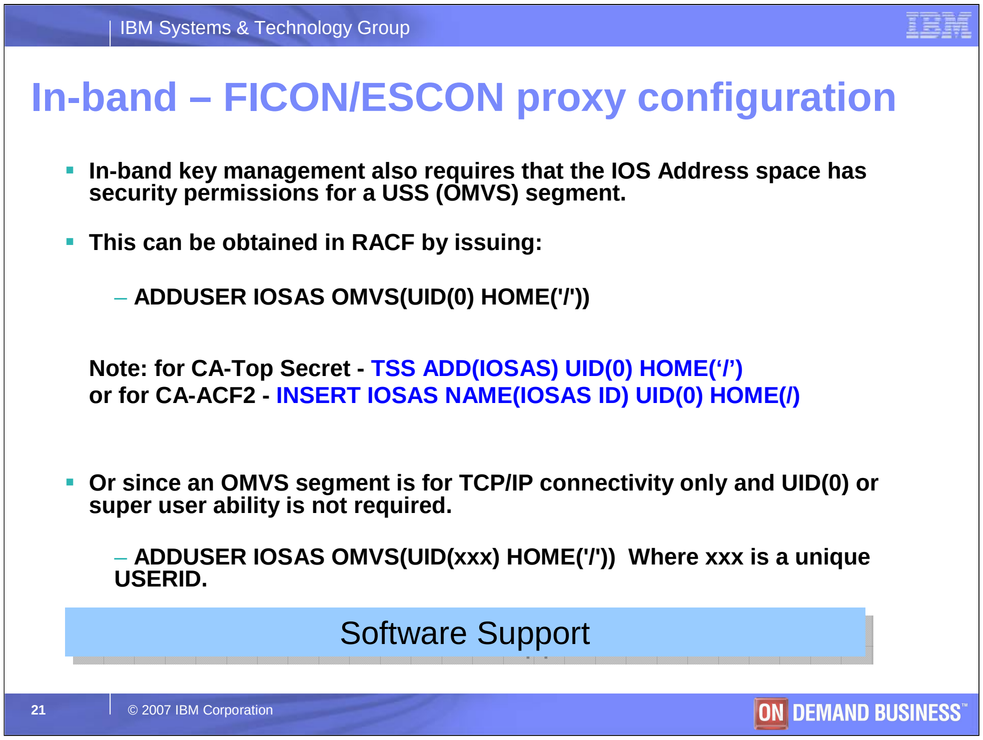
Fibre

Control Unit

20

ON DEMAND BUSINESS™

The diagram above shows the in-band communication path between the drive and the key manager with the IOS (ESCON/FICON) proxy layer being in the middle. The IOS proxy will then communicate across TCP/IP to a key manager that is running on the same system that is reading or writing to tape or on another system.



IBM Systems & Technology Group

In-band – FICON/ESCON proxy configuration

- **In-band key management also requires that the IOS Address space has security permissions for a USS (OMVS) segment.**
- **This can be obtained in RACF by issuing:**
  - **ADDUSER IOSAS OMVS(UID(0) HOME('/'))**

Note: for CA-Top Secret - TSS ADD(IOSAS) UID(0) HOME('/')
or for CA-ACF2 - INSERT IOSAS NAME(IOSAS ID) UID(0) HOME(/)

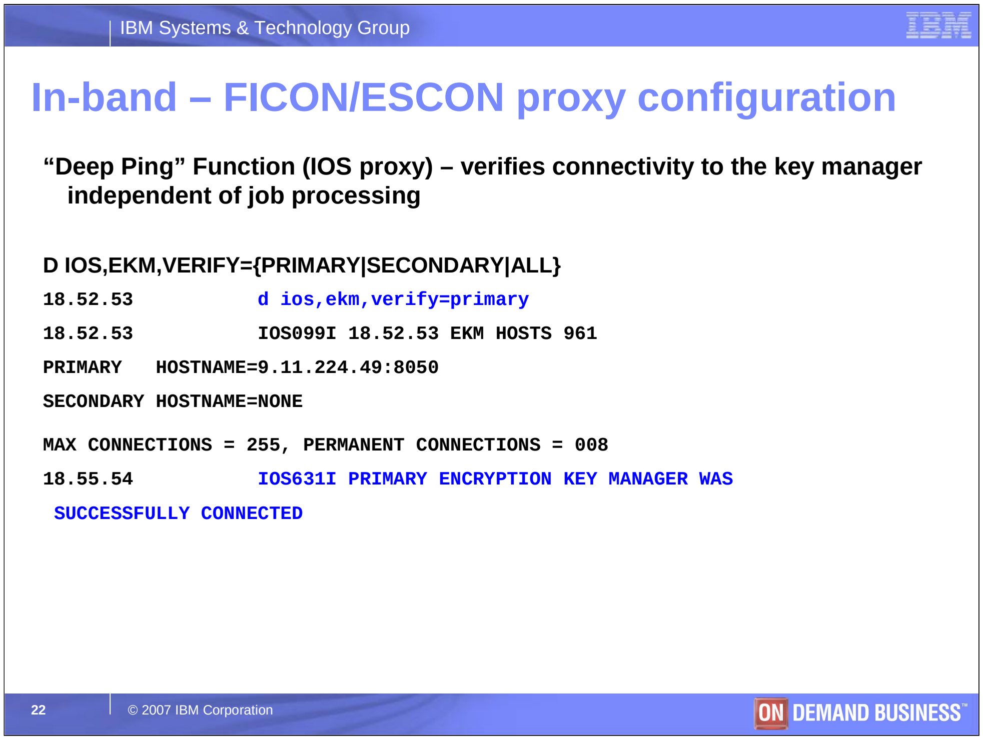
- **Or since an OMVS segment is for TCP/IP connectivity only and UID(0) or super user ability is not required.**
  - **ADDUSER IOSAS OMVS(UID(xxx) HOME('/')) Where xxx is a unique USERID.**

Software Support

21 © 2007 IBM Corporation

ON DEMAND BUSINESS™

In-band key management on z/OS also requires that the IOS address space has security permissions for a Unix System Services (USS) or Open-MVS (OMVS) segment. The syntax above shows what is needed to establish the security permissions for the IOS address space. Also note that an IPL is needed after the security permissions have been established for the IOS address space.



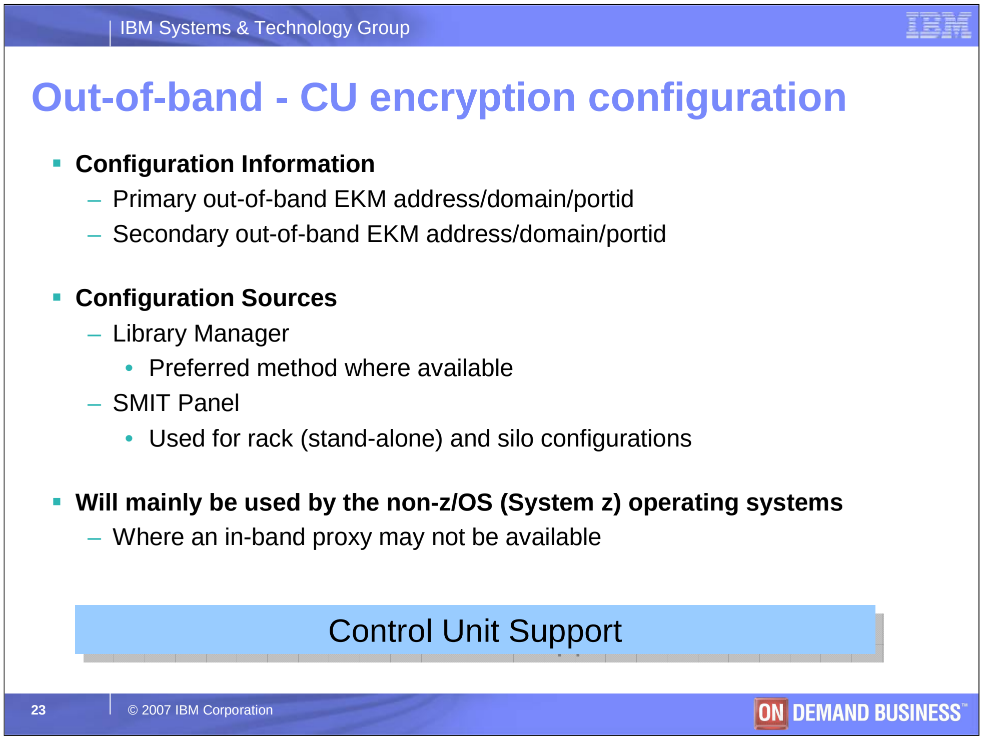
In-band – FICON/ESCON proxy configuration

“Deep Ping” Function (IOS proxy) – verifies connectivity to the key manager independent of job processing

D IOS,EKM,VERIFY={PRIMARY|SECONDARY|ALL}

```
18.52.53           d ios,ekm,verify=primary
18.52.53           IOS099I 18.52.53 EKM HOSTS 961
PRIMARY   HOSTNAME=9.11.224.49:8050
SECONDARY HOSTNAME=NONE
MAX CONNECTIONS = 255, PERMANENT CONNECTIONS = 008
18.55.54           IOS631I PRIMARY ENCRYPTION KEY MANAGER WAS
 SUCCESSFULLY CONNECTED
```

The IOS proxy also has a “deep ping” capability that can be used to verify whether the IOS proxy can communicate with the encryption key manager (EKM). The VERIFY option of the DISPLAY IOS,EKM command can be used to verify whether the IOS proxy can communicate with the primary and secondary key manager.   The example above shows the syntax of the command and the output for successful communication with the primary key manager.



IBM Systems & Technology Group

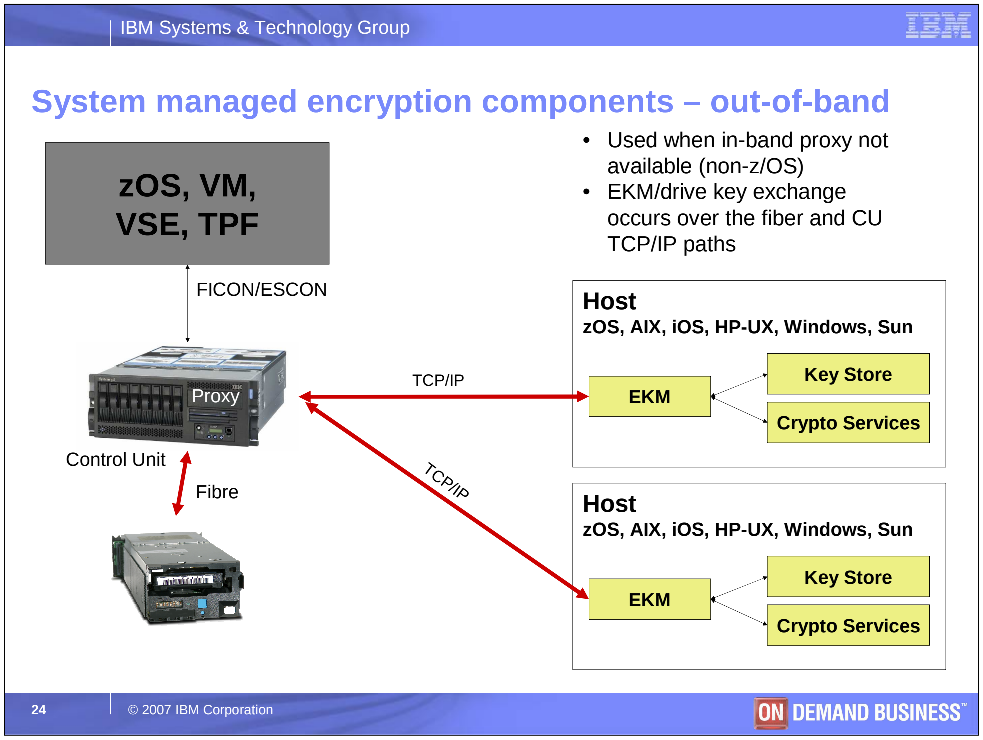
Out-of-band - CU encryption configuration

- **Configuration Information**
  - Primary out-of-band EKM address/domain/portid
  - Secondary out-of-band EKM address/domain/portid
- **Configuration Sources**
  - Library Manager
    - Preferred method where available
  - SMIT Panel
    - Used for rack (stand-alone) and silo configurations
- **Will mainly be used by the non-z/OS (System z) operating systems**
  - Where an in-band proxy may not be available

Control Unit Support

23

For out-of-band key management, the TCP/IP address of the encryption key manager is established through the tape library or the control unit as applicable for the installation.   Out-of-band key management will mainly be used by the System z operating systems that do not have an in-band proxy available such as VM, VSE and TPF.   Though it is also available as an option under z/OS.



System managed encryption components – out-of-band

- Used when in-band proxy not available (non-z/OS)
- EKM/drive key exchange occurs over the fiber and CU TCP/IP paths

zOS, VM, VSE, TPF

FICON/ESCON

Proxy

Control Unit

Fibre

TCP/IP

TCP/IP

Host
zOS, AIX, iOS, HP-UX, Windows, Sun

EKM

Key Store

Crypto Services

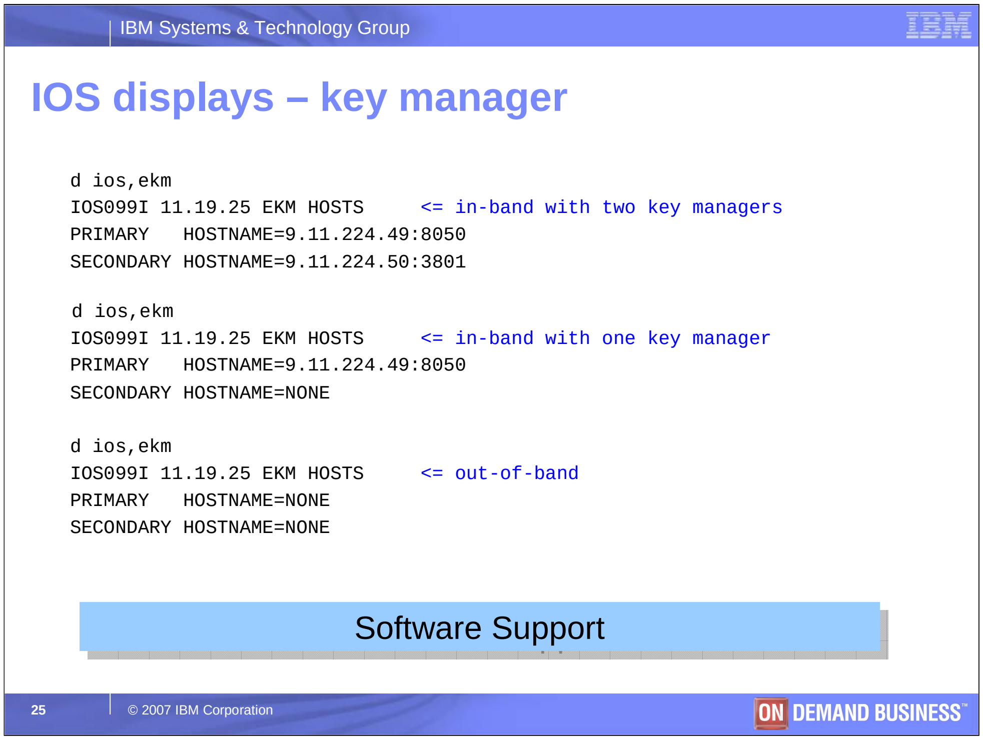
Host
zOS, AIX, iOS, HP-UX, Windows, Sun

EKM

Key Store

Crypto Services

The diagram above shows the out-of-band communication path between the drive and the key manager with the tape control unit being in the middle. The tape control unit will then communicate across TCP/IP to a key manager that is running on the same system that is reading or writing to tape or on another system.



IBM Systems & Technology Group

IOS displays – key manager

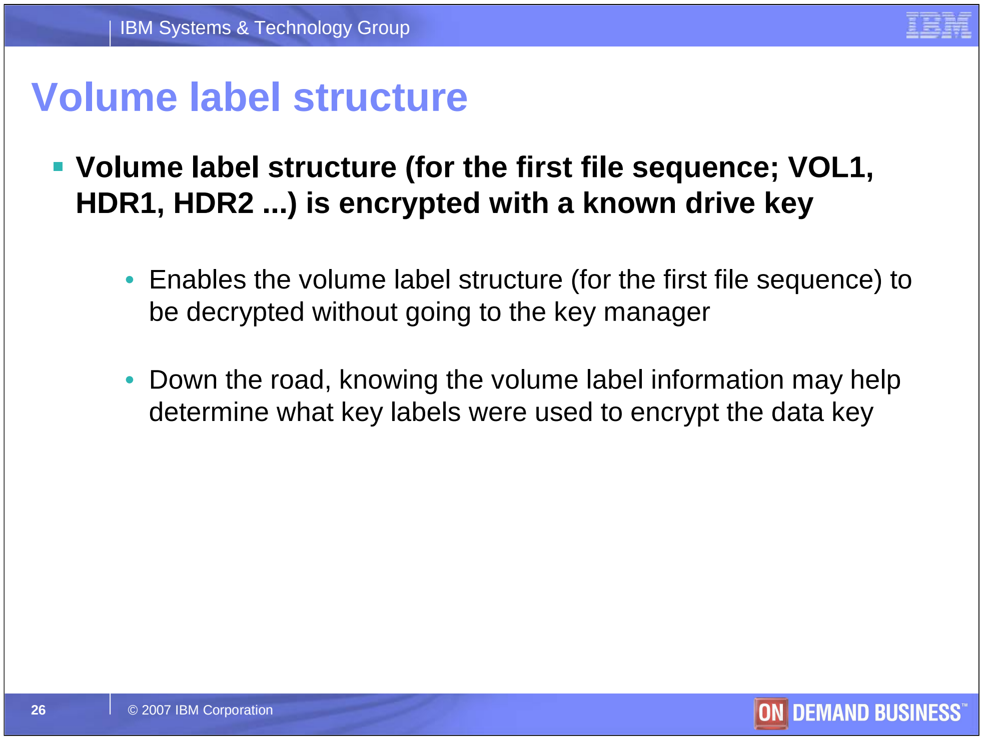
```
d ios,ekm
IOS099I 11.19.25 EKM HOSTS     <= in-band with two key managers
PRIMARY   HOSTNAME=9.11.224.49:8050
SECONDARY HOSTNAME=9.11.224.50:3801

d ios,ekm
IOS099I 11.19.25 EKM HOSTS     <= in-band with one key manager
PRIMARY   HOSTNAME=9.11.224.49:8050
SECONDARY HOSTNAME=NONE

d ios,ekm
IOS099I 11.19.25 EKM HOSTS     <= out-of-band
PRIMARY   HOSTNAME=NONE
SECONDARY HOSTNAME=NONE
```

Software Support

25

ON DEMAND BUSINESS™

The examples above show the DISPLAY IOS,EKM command with different setups established. The first display indicates that in-band key management is established for both a primary and a secondary key manager. The second display indicates that just a primary key manager has been established and the third display shows that in-band key management has not been established. In this last example, if an encryption request is sent to the tape drive, the communication path will be out-of-band, from the control unit to the EKM.



IBM Systems & Technology Group

Volume label structure

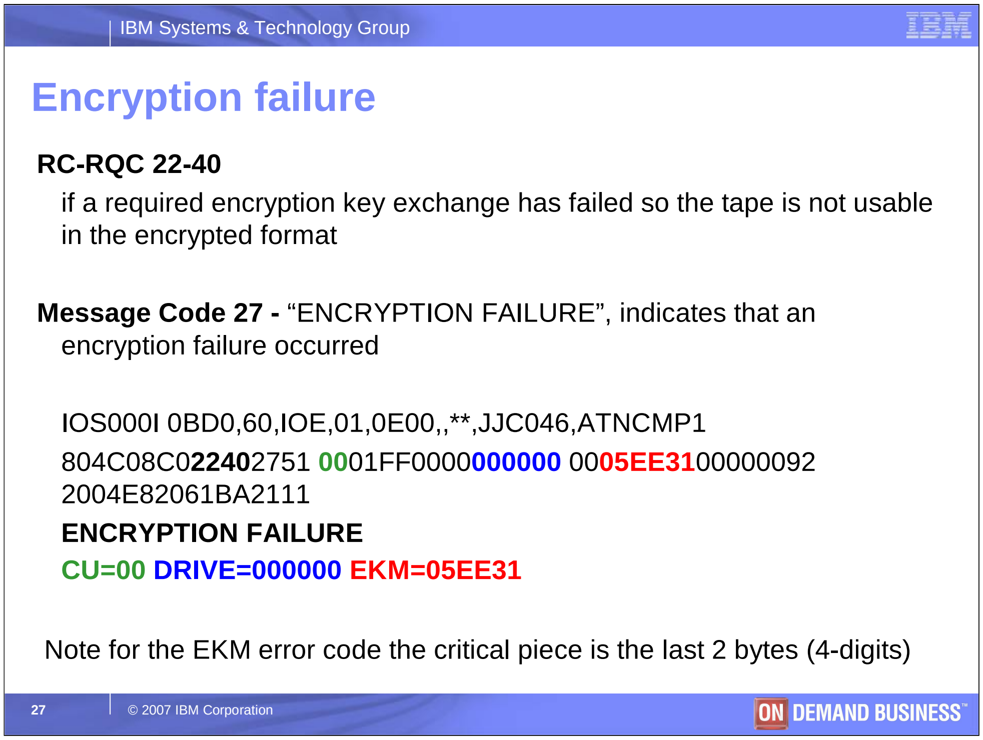
- **Volume label structure (for the first file sequence; VOL1, HDR1, HDR2 ...) is encrypted with a known drive key**
  - Enables the volume label structure (for the first file sequence) to be decrypted without going to the key manager
  - Down the road, knowing the volume label information may help determine what key labels were used to encrypt the data key

26 © 2007 IBM Corporation ON DEMAND BUSINESS™

If the data on a tape cartridge is encrypted, all data on that tape cartridge is encrypted using a data key provided by the encryption key manager (EKM). However, the volume label structure for the first file sequence is encrypted using a known drive key.

This enables the volume label structure to be read without going to the EKM. Why is this important?

Being able to read the volume label structure for the first file sequence may help you determine what key labels may have been used to encrypt the data key. This will help you to determine what data is on the tape cartridge. Unless you have the private key of the asymmetric key pair that was used to encrypt the data key, you won’t be able to decrypt the data key.



IBM Systems & Technology Group

Encryption failure

RC-RQC 22-40

if a required encryption key exchange has failed so the tape is not usable in the encrypted format

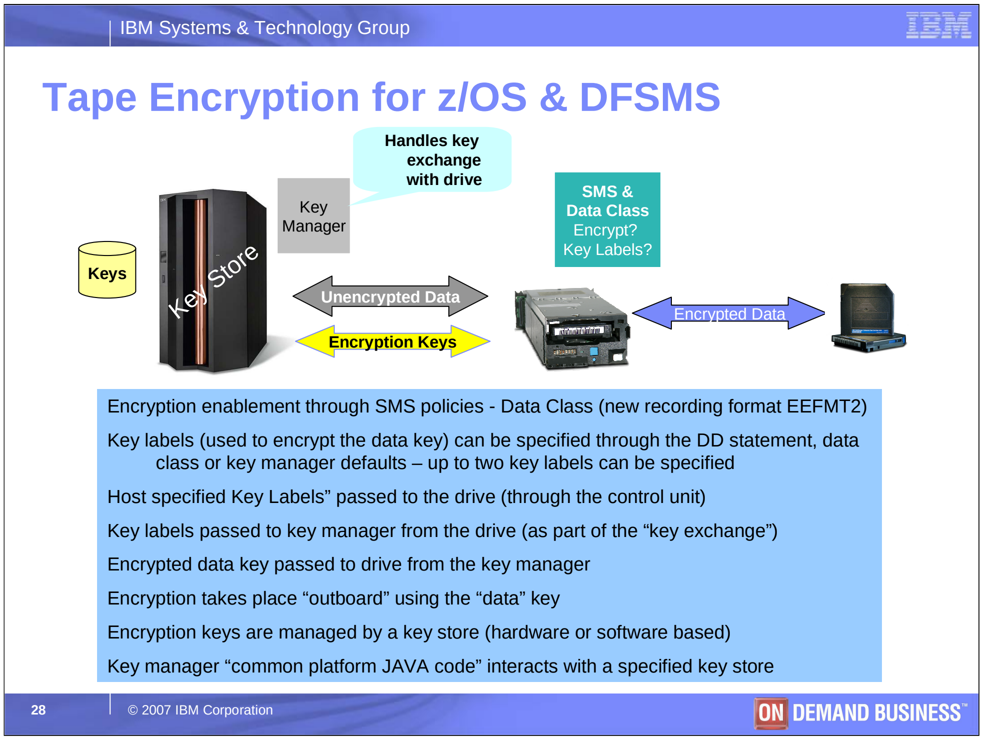
Message Code 27 - “ENCRYPTION FAILURE”, indicates that an encryption failure occurred

```
IOS000I 0BD0,60,IOE,01,0E00,,**,JJC046,ATNCMP1
804C08C022402751 0001FF0000000000 0005EE3100000092
2004E82061BA2111
ENCRYPTION FAILURE
CU=00 DRIVE=000000 EKM=05EE31
```

Note for the EKM error code the critical piece is the last 2 bytes (4-digits)

27 ON DEMAND BUSINESS™

So what happens if you request the encryption format EEFMT2, but a problem occurs? The example above shows a sample IOS000I message indicating whether the failure was a control unit (CU), drive or encryption key manager (EKM) failure. In the example above, the failure was an encryption key manager (EKM) failure X’EE31” with invalid key labels being passed that were not in the key store. Refer to *z/OS DFSMS Software Support for IBM System Storage TS1120 Tape Drive (3592) - SC26-7514-03* for additional information on the error codes that may be reported.



IBM Systems & Technology Group

Tape Encryption for z/OS & DFSMS

Handles key exchange with drive

Key Manager

SMS & Data Class Encrypt? Key Labels?

Keys

Key Store

Unencrypted Data

Encryption Keys

Encrypted Data

Encryption enablement through SMS policies - Data Class (new recording format EEFMT2)

Key labels (used to encrypt the data key) can be specified through the DD statement, data class or key manager defaults – up to two key labels can be specified

Host specified Key Labels" passed to the drive (through the control unit)

Key labels passed to key manager from the drive (as part of the "key exchange")

Encrypted data key passed to drive from the key manager

Encryption takes place "outboard" using the "data" key

Encryption keys are managed by a key store (hardware or software based)

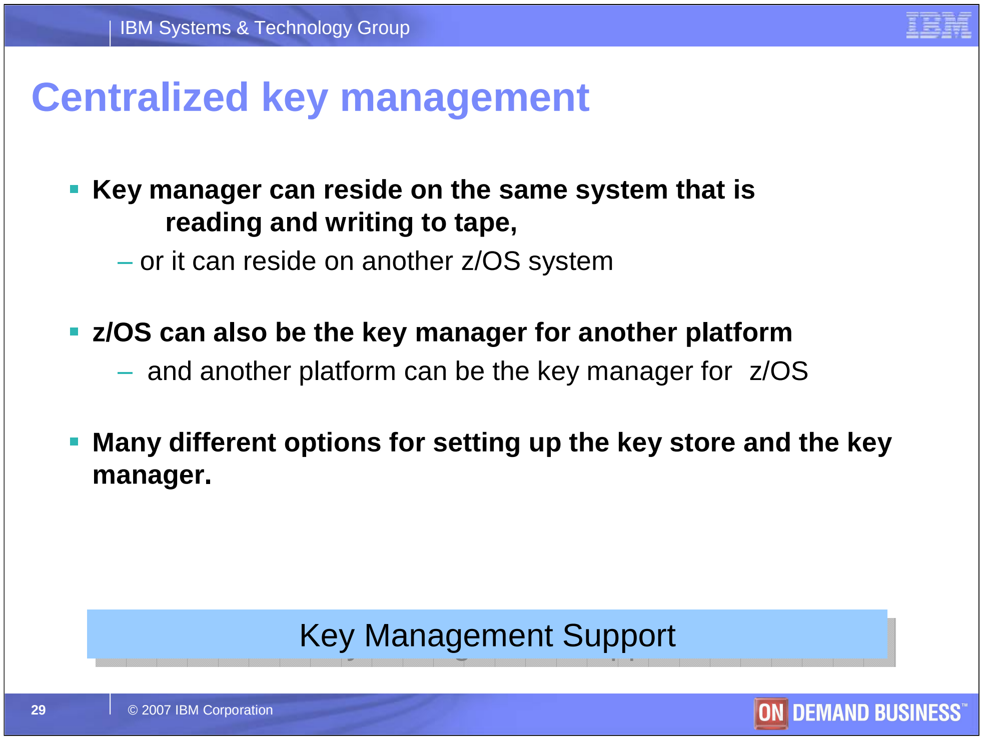
Key manager "common platform JAVA code" interacts with a specified key store

28

ON DEMAND BUSINESS™

The picture above summarizes the encryption flow. First, encryption starts with data class enablement through specification of the encryption format EEFMT2 (EE2). Also, as part of an encryption request, the key labels that are used to encrypt the data key can come from a couple of different sources (DD statement, data class or the encryption key manager).

Any host specified key labels are then passed to the drive along with the request to encrypt the tape cartridge. The key labels are then passed to the encryption key manager from the drive as part of the key exchange. The encrypted data key is then passed from the key manger to the drive with encryption of the data taking place outboard in the drive using the data key that was passed by the key manager. The encryption keys (key encrypting keys) that are used to encrypt "wrap'" the data key are maintained by a key store and accessed by the key manager. The key manager is a common platform JAVA application that can run on any IBM server: i, x, p or z and can also run on Windows, HP-UX or Sun Solaris.

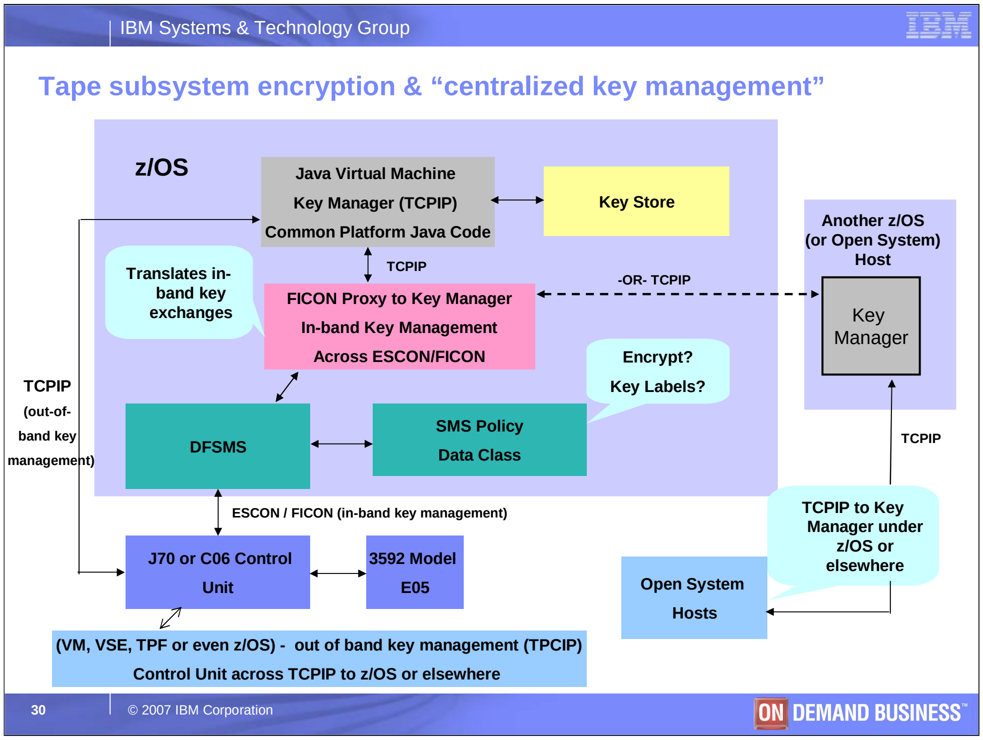


Centralized key management

- **Key manager can reside on the same system that is reading and writing to tape,**
  - or it can reside on another z/OS system
- **z/OS can also be the key manager for another platform**
  - and another platform can be the key manager for z/OS
- **Many different options for setting up the key store and the key manager.**

Key Management Support

The encryption key manager (EKM) can reside on the same system that is reading and writing to tape, or it can reside on another z/OS system. z/OS can also be the key manager for another platform's encryption keys and another platform can also manage z/OS's encryption keys. Since the EKM can be installed on just about any JAVA Virtual Machine, there are many different options for setting up the key store and the key manager.



IBM Systems & Technology Group

Tape subsystem encryption & “centralized key management”

z/OS

Java Virtual Machine
Key Manager (TCPIP)
Common Platform Java Code

Key Store

TCPIP

Translates in-band key exchanges

FICON Proxy to Key Manager
In-band Key Management
Across ESCON/FICON

-OR- TCPIP

Another z/OS (or Open System) Host

Key Manager

Encrypt?
Key Labels?

TCPIP (out-of-band key management)

DFSMS

SMS Policy
Data Class

TCPIP

ESCON / FICON (in-band key management)

TCPIP to Key Manager under z/OS or elsewhere

J70 or C06 Control Unit

3592 Model E05

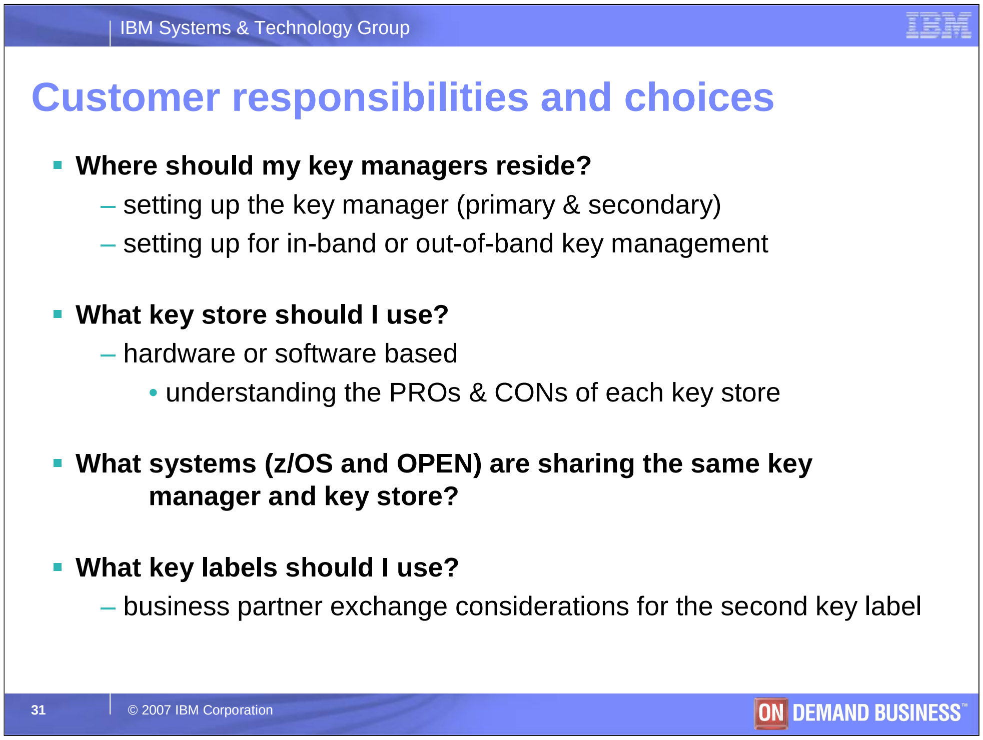
Open System Hosts

(VM, VSE, TPF or even z/OS) - out of band key management (TPCIP)
Control Unit across TCPIP to z/OS or elsewhere

30 © 2007 IBM Corporation

ON DEMAND BUSINESS™

The picture above ties together all of the pieces involved in the solution and what the different options are for centralized key management. The picture really starts with the “SMS Policy” box making a decision whether to encrypt a particular application or job’s data. If the data is to be encrypted, it then shows the flow between the drive and the encryption key manager going either in-band or out-of-band for key management. The picture also shows that even with in-band key management there are many different options for where the key manager can reside. The key manager does not have to reside on the same system that is reading and writing to tape.



IBM Systems & Technology Group

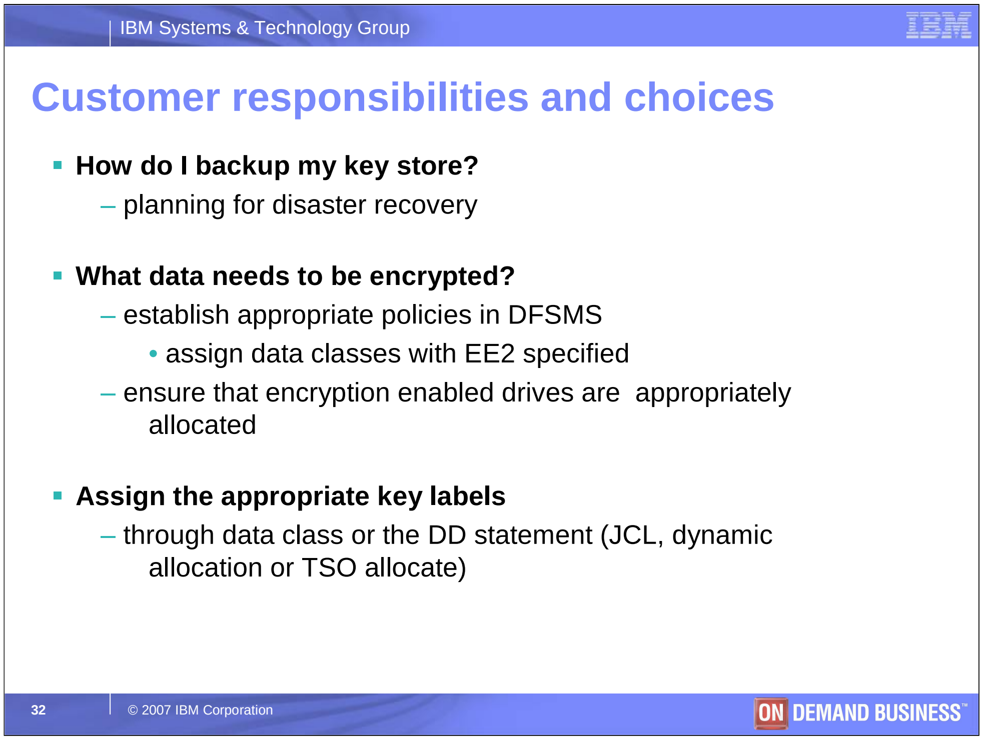
Customer responsibilities and choices

- **Where should my key managers reside?**
  - – setting up the key manager (primary & secondary)
  - – setting up for in-band or out-of-band key management
- **What key store should I use?**
  - – hardware or software based
    - understanding the PROs & CONs of each key store
- **What systems (z/OS and OPEN) are sharing the same key manager and key store?**
- **What key labels should I use?**
  - – business partner exchange considerations for the second key label

31

ON DEMAND BUSINESS™

We've provided an overview of the encryption support and the role that z/OS & DFSMS play in the tape encryption solution. What must you take into consideration when thinking about the encryption solution? First you must consider where your key manager (primary and secondary) should reside and whether you will be using in-band or out-of-band key management. Then you must also consider what key store you should use to house the key encrypting keys and the PROs and CONs of each key store. You must also decide what systems (z/OS and Open) will be sharing the same key store and key manager. Then for business partner key exchanges, what key labels should be used for the second key label and consideration of the "hash" encoding mechanism.



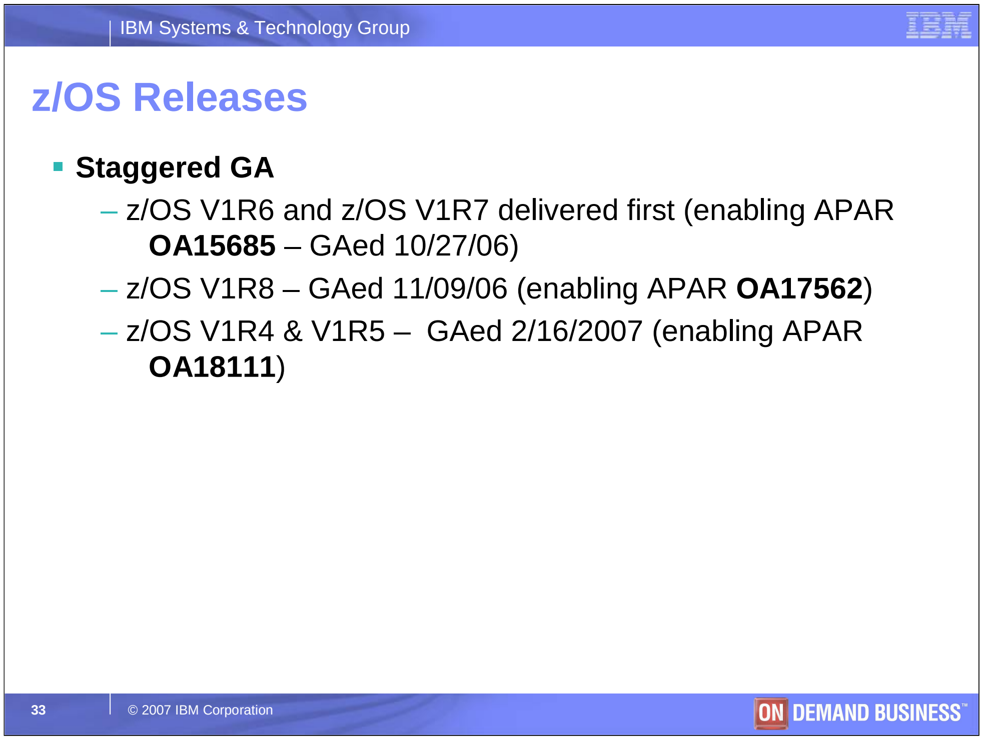
IBM Systems & Technology Group

Customer responsibilities and choices

- **How do I backup my key store?**
  - – planning for disaster recovery
- **What data needs to be encrypted?**
  - – establish appropriate policies in DFSMS
    - assign data classes with EE2 specified
  - – ensure that encryption enabled drives are appropriately allocated
- **Assign the appropriate key labels**
  - – through data class or the DD statement (JCL, dynamic allocation or TSO allocate)

32 © 2007 IBM Corporation

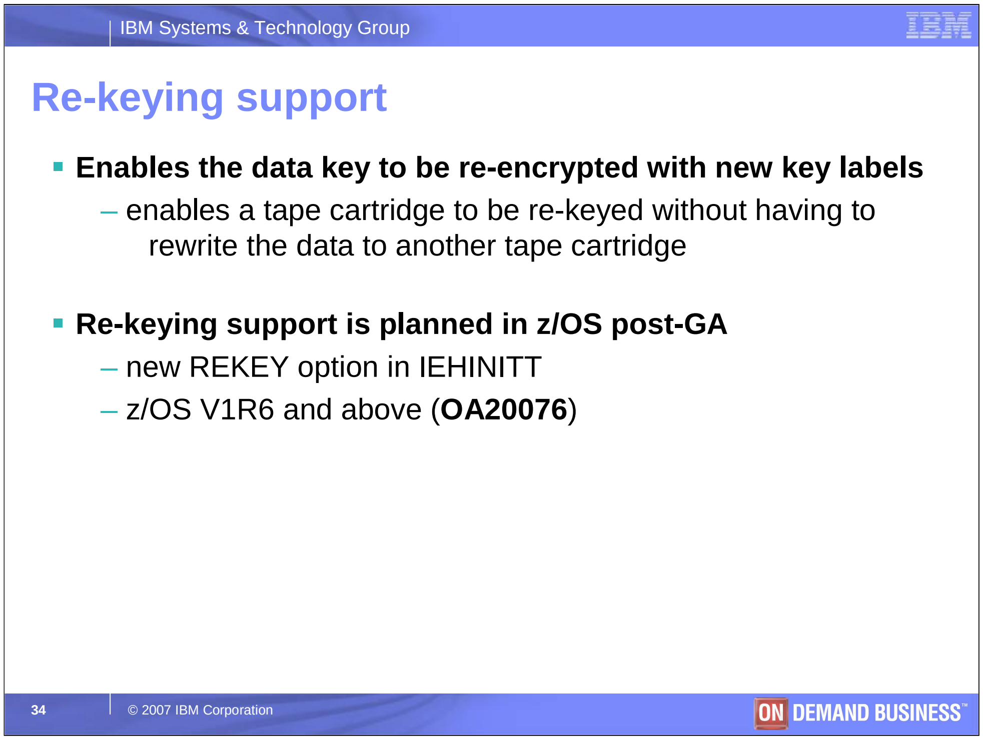
You must also plan to backup your key store and determine what is needed for disaster recovery so that your key store and key encrypting keys can be restored. Then determine what data needs to be encrypted. Is it all tape data or certain applications? You need to establish the appropriate data class policies and assign the appropriate key labels.



z/OS Releases

- **Staggered GA**
  - z/OS V1R6 and z/OS V1R7 delivered first (enabling APAR **OA15685** – GAed 10/27/06)
  - z/OS V1R8 – GAed 11/09/06 (enabling APAR **OA17562**)
  - z/OS V1R4 & V1R5 – GAed 2/16/2007 (enabling APAR **OA18111**)

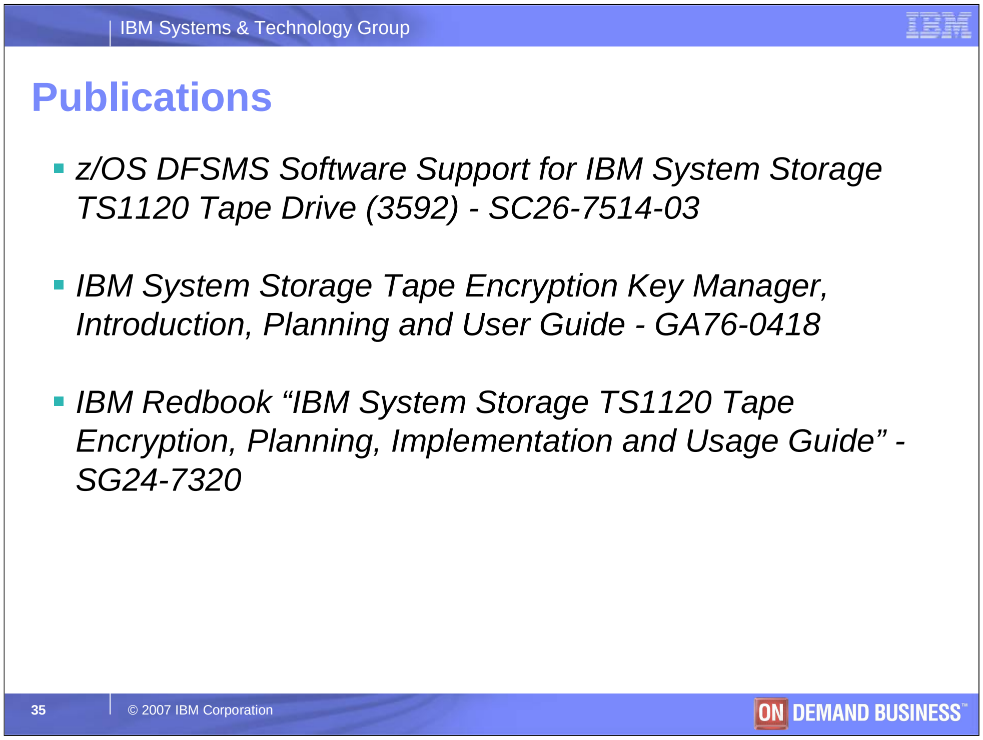
The encryption support on z/OS was delivered with a staggered GA. The slide above shows the different GA dates for the releases and the main enabling APAR for each release. The enabling APAR (and underlying PTF) will then pull in all of the necessary support.



Re-keying support

- **Enables the data key to be re-encrypted with new key labels**
  - enables a tape cartridge to be re-keyed without having to rewrite the data to another tape cartridge
- **Re-keying support is planned in z/OS post-GA**
  - new REKEY option in IEHINITT
  - z/OS V1R6 and above (**OA20076**)

As a follow-on to the encryption support discussed in this presentation, there are plans to supporting tape encryption re-keying on z/OS. Re-keying enables a data key to be re-encrypted “re-wrapped” using new key labels without having to rewrite the data to another tape cartridge. This is important if the tape now needs to go to another business partner. As part of the re-key operation, the tape cartridge, will be mounted so that new EEDK structures can be written on the tape cartridge. This support will be provided through enhancements to the existing tape initialization utility (IEHINITT) available on z/OS. There will be a new REKEY option available that will enable new key labels to be specified. Refer to APAR OA20076 for additional information. This support will be provided on z/OS V1R6 and above with a targeted GA date 3Q2007. However, current plans are always subject to change.



Publications

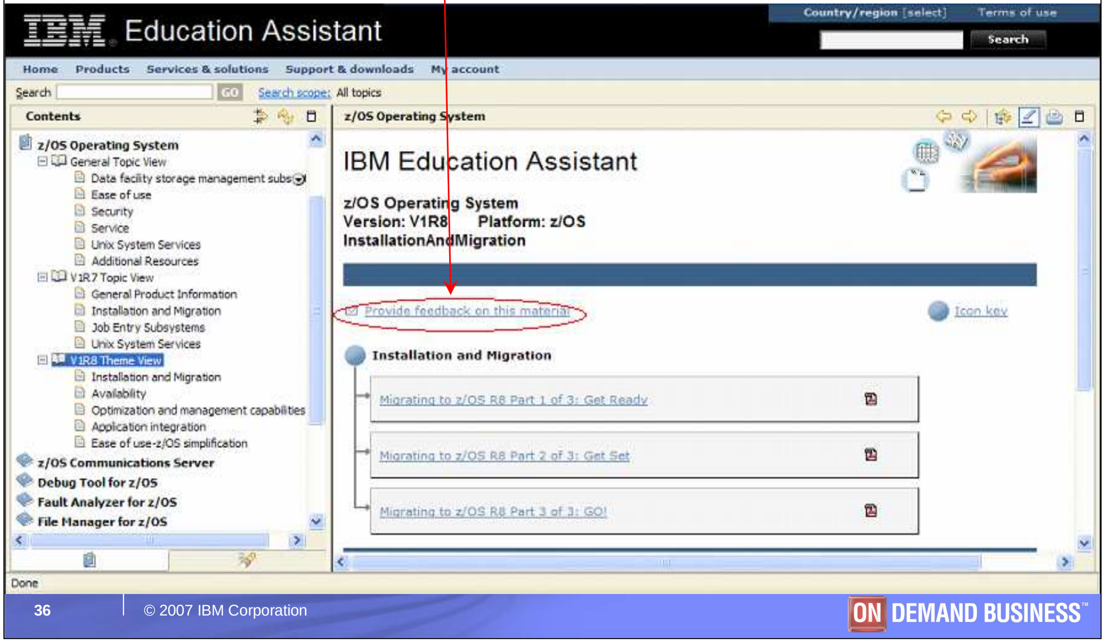
- *z/OS DFSMS Software Support for IBM System Storage TS1120 Tape Drive (3592) - SC26-7514-03*
- *IBM System Storage Tape Encryption Key Manager, Introduction, Planning and User Guide - GA76-0418*
- *IBM Redbook “IBM System Storage TS1120 Tape Encryption, Planning, Implementation and Usage Guide” - SG24-7320*

ON DEMAND BUSINESS™

Refer to the publications above for additional detail on the support. Also, refer to a planned separate presentation for additional detail on the encryption flow and terminology.

Did you find this IBM Education Assistant module useful?

Provide feedback and specify the module title on your response.



z/OS Operating System

IBM Education Assistant

z/OS Operating System
Version: V1R8 Platform: z/OS
InstallationAndMigration

Provide feedback on this material

Icon key

Installation and Migration

Migrating to z/OS R8 Part 1 of 3: Get Ready

Migrating to z/OS R8 Part 2 of 3: Get Set

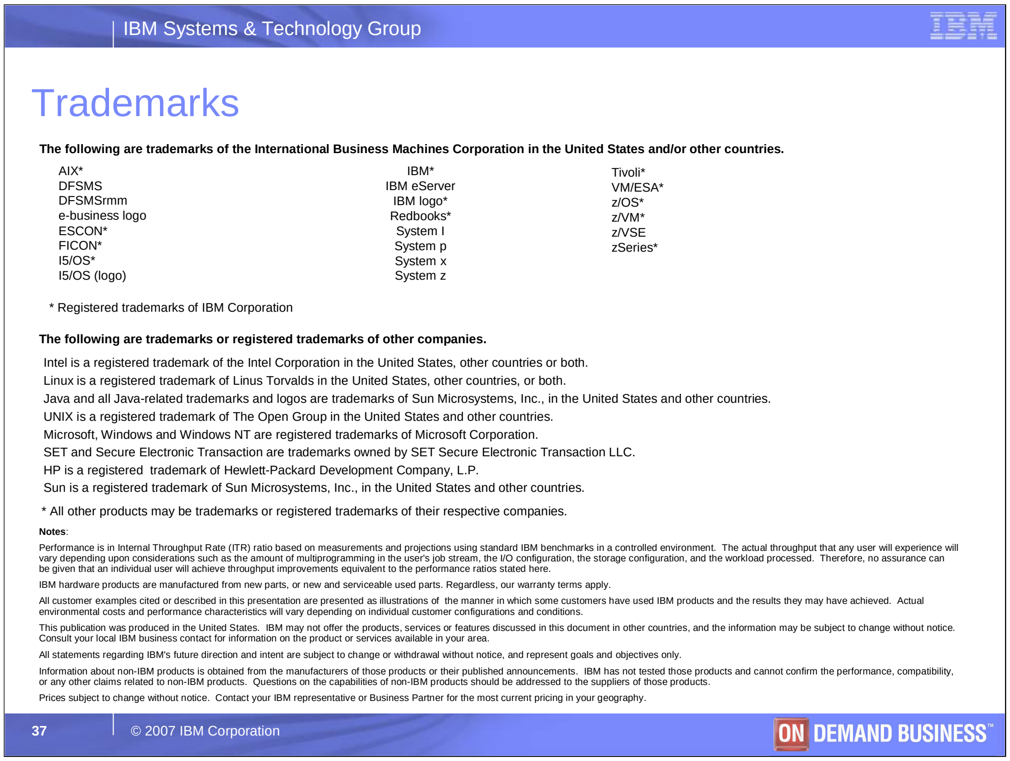
Migrating to z/OS R8 Part 3 of 3: GO!

In your feedback to IBM, answer these three questions:

How helpful was this IEA presentation? Give it a rating from 1 to 5 where 1 = very helpful and 5 = not at all helpful

Did this presentation save you a service call to IBM? Yes or No

What other topics would you like to see covered in z/OS IEA?



Trademarks

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

AIX*
DFSMS
DFSMSrmm
e-business logo
ESCON*
FICON*
I5/OS*
I5/OS (logo)

IBM*
IBM eServer
IBM logo*
Redbooks*
System I
System p
System x
System z

Tivoli*
VM/ESA*
z/OS*
z/VM*
z/VSE
zSeries*

* Registered trademarks of IBM Corporation

The following are trademarks or registered trademarks of other companies.

Intel is a registered trademark of the Intel Corporation in the United States, other countries or both.
Linux is a registered trademark of Linus Torvalds in the United States, other countries, or both.
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.
HP is a registered trademark of Hewlett-Packard Development Company, L.P.
Sun is a registered trademark of Sun Microsystems, Inc., in the United States and other countries.

* 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.