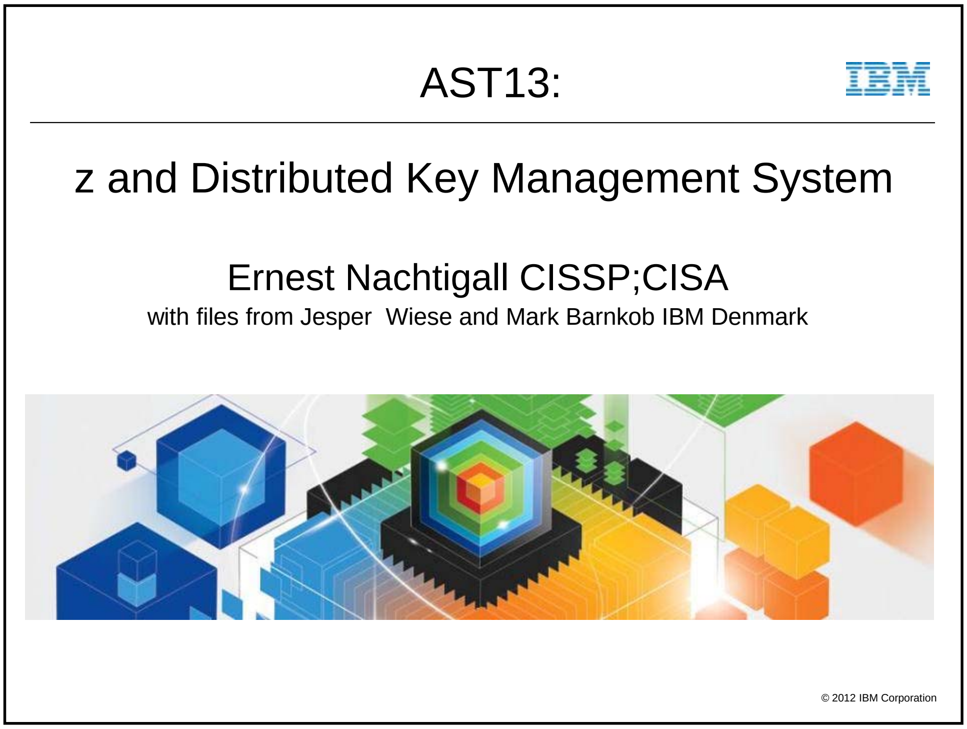
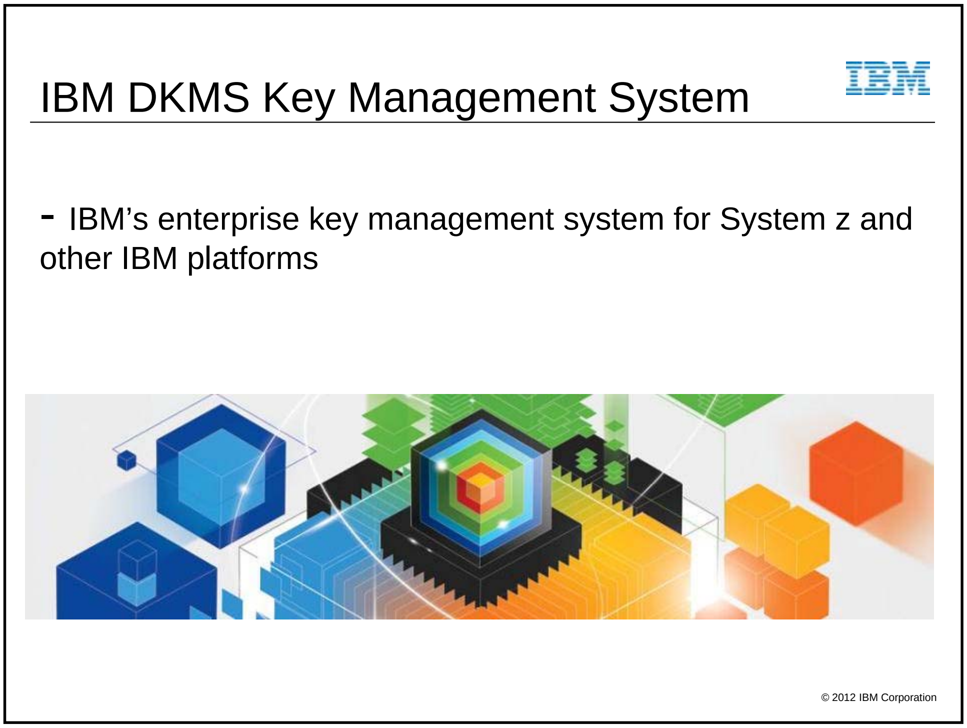


AST13:

z and Distributed Key Management System

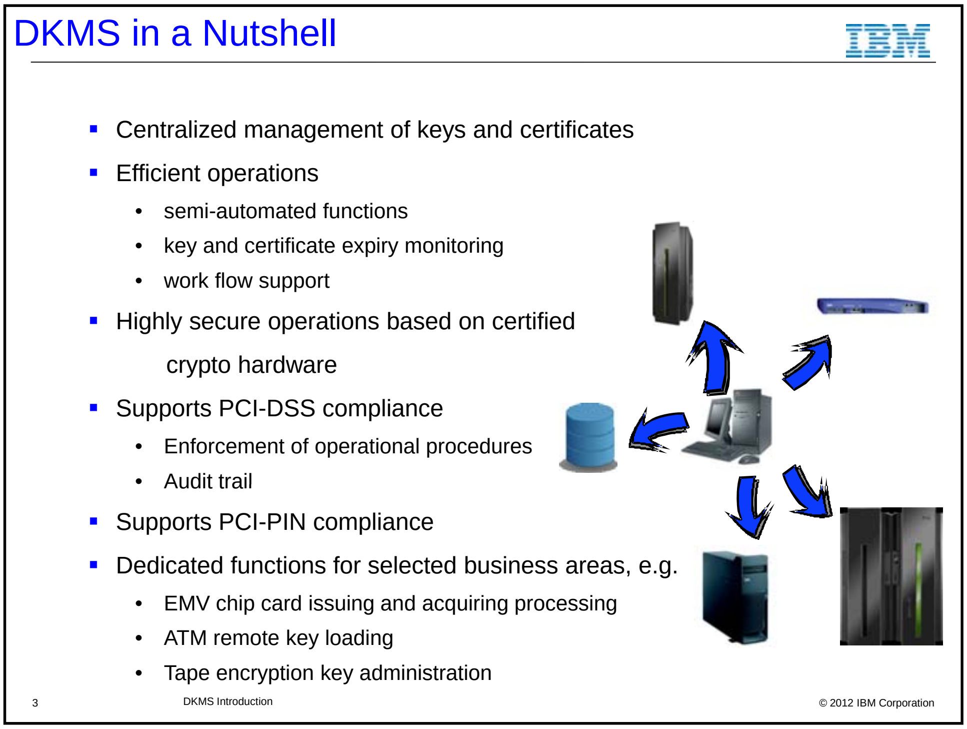
Ernest Nachtigall CISSP;CISA

with files from Jesper Wiese and Mark Barnkob IBM Denmark



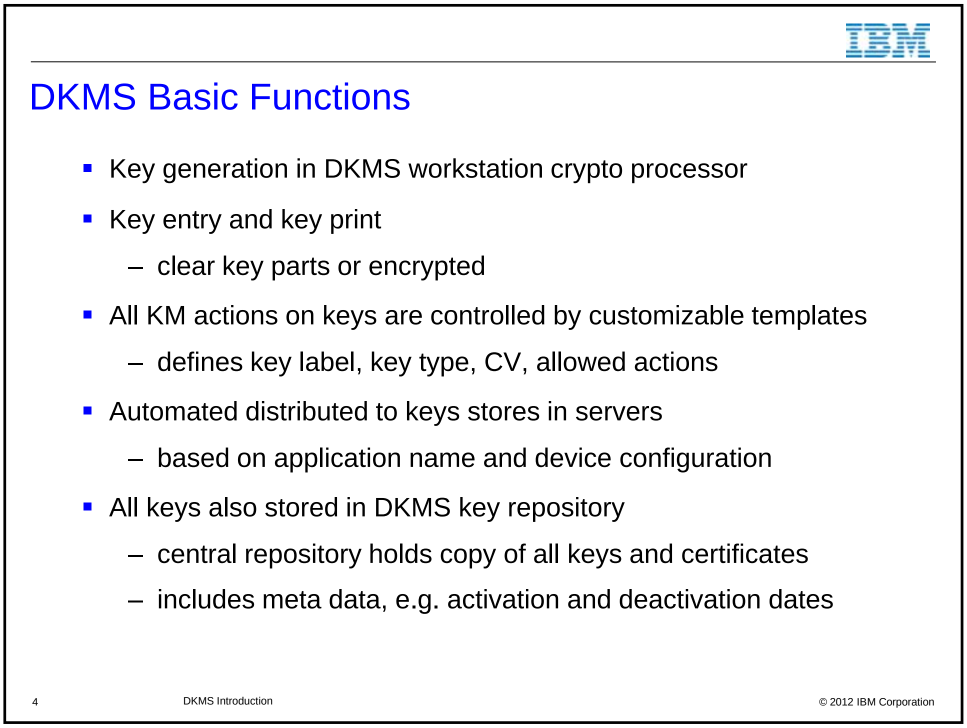
IBM DKMS Key Management System

- IBM's enterprise key management system for System z and other IBM platforms



DKMS in a Nutshell

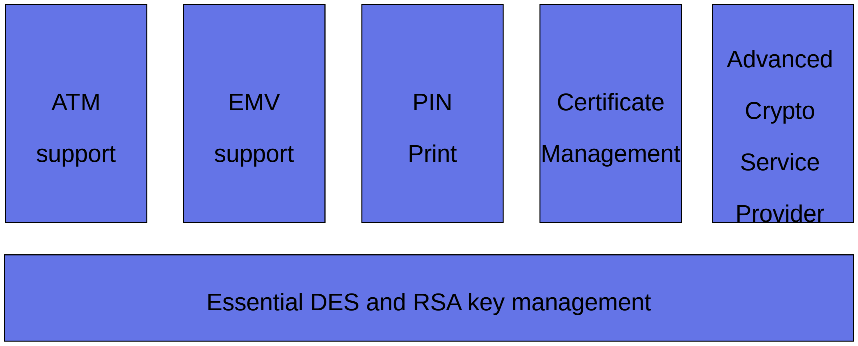
- Centralized management of keys and certificates
- Efficient operations
  - semi-automated functions
  - key and certificate expiry monitoring
  - work flow support
- Highly secure operations based on certified crypto hardware
- Supports PCI-DSS compliance
  - Enforcement of operational procedures
  - Audit trail
- Supports PCI-PIN compliance
- Dedicated functions for selected business areas, e.g.
  - EMV chip card issuing and acquiring processing
  - ATM remote key loading
  - Tape encryption key administration



DKMS Basic Functions

- Key generation in DKMS workstation crypto processor
- Key entry and key print
  - – clear key parts or encrypted
- All KM actions on keys are controlled by customizable templates
  - – defines key label, key type, CV, allowed actions
- Automated distributed to keys stores in servers
  - – based on application name and device configuration
- All keys also stored in DKMS key repository
  - – central repository holds copy of all keys and certificates
  - – includes meta data, e.g. activation and deactivation dates

DKMS – The Foundation and the Pillars



ATM support

EMV support

PIN Print

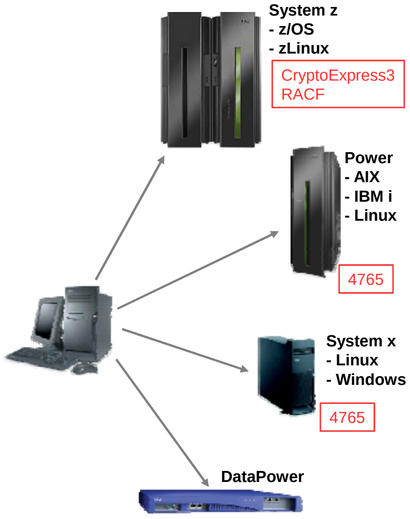
Certificate Management

Advanced Crypto Service Provider

Essential DES and RSA key management

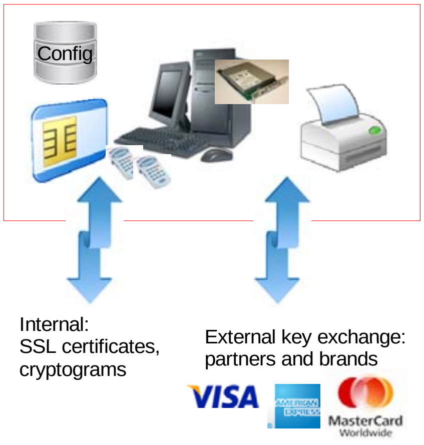
DKMS Key Features

- On-line management of large, heterogeneous environments – from mainframes to distributed servers
- Symmetric and asymmetric keys as well as certificates
- Continuous operation ensured by secure backup and restore of keys
- Automated monitoring of expiry of keys and certificates
- Semi-automated operations enable easy rotation of keys and renewal of certificates



DKMS Workstation

- Focal point for all key management
  - generation and entry of persistent keys
  - monitoring and administration of key life cycle
  - printing of key parts
- IBM4765 crypto co-processor FIPS 140-2 level 4
- Strong 2-factor authentication (Smart cards)
- Dual control, Group logon m-of –n
- Split knowledge enforced



Internal:
SSL certificates,
cryptograms

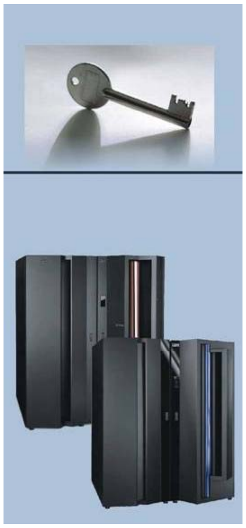
External key exchange:
partners and brands

DKMS Key Values

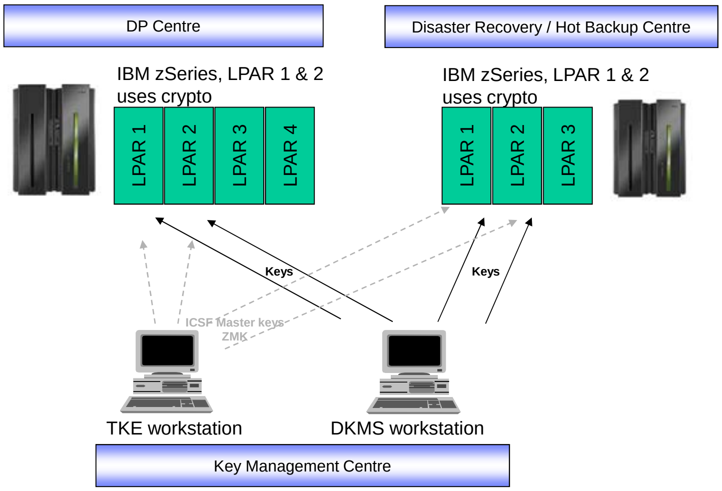
- System audit perspective:
  - Log of all essential operations
  - Support for System Monitoring Facility on z/OS
  - Supports PCI-DSS compliance
    - Enforcement of operational procedures
    - Audit trail
- Development perspective:
  - Removes key management burden from customer’s applications
  - High level API offered for several business areas thus freeing application programmers from dealing directly with crypto

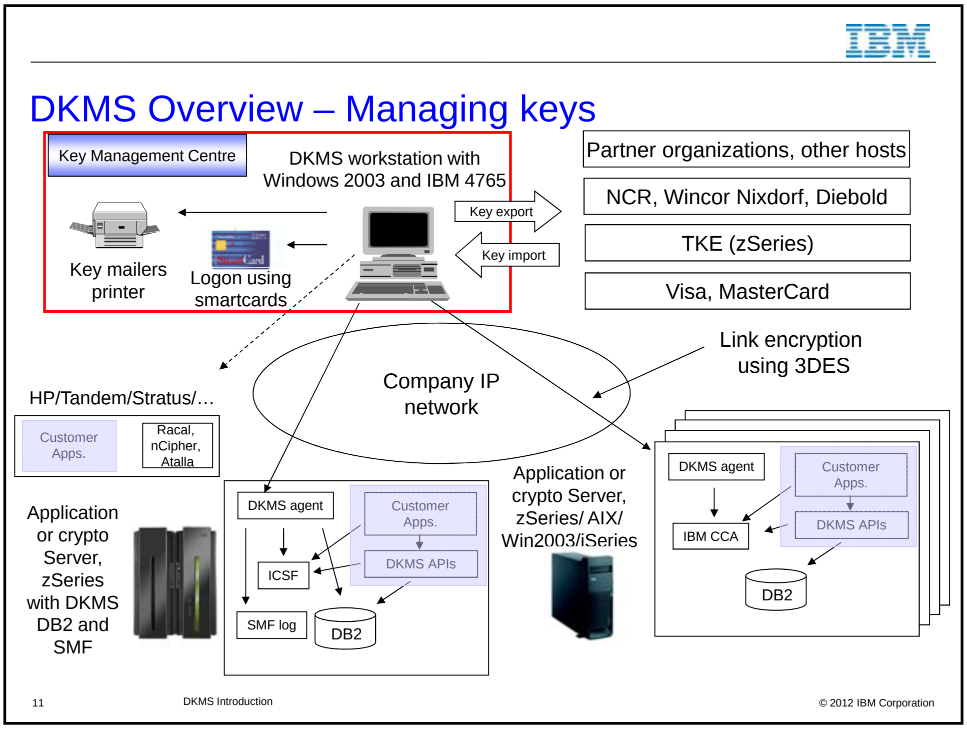
DKMS Supported Systems and Keys

- Supported key types and lengths
  - DES, TDES
  - RSA (up to 4096 bit keys)
  - AES (128, 192, 256 bit keys)
- IBM
  - CryptoExpress2 and 3 on z/OS and zLinux
  - RACF on z/OS (private keys in ICSF)
  - DataPower (private keys and certificates)
  - IBM 4764/5 on AIX, OS/400, IBM i, Windows
- Thales
- Off-line
  - PKCS#11
  - SSL server key stores
    - PKCS#12, JKS, KDB



Typical DKMS Configuration with IBM zSeries





DKMS Overview – Managing keys

Key Management Centre

DKMS workstation with Windows 2003 and IBM 4765

Key mailers printer

Logon using smartcards

Key export

Key import

Partner organizations, other hosts

NCR, Wincor Nixdorf, Diebold

TKE (zSeries)

Visa, MasterCard

Link encryption using 3DES

Company IP network

HP/Tandem/Stratus/…

Customer Apps.

Racal, nCipher, Atalla

Application or crypto Server, zSeries with DKMS DB2 and SMF

DKMS agent

Customer Apps.

DKMS APIs

ICSF

SMF log

DB2

Application or crypto Server, zSeries/ AIX/ Win2003/iSeries

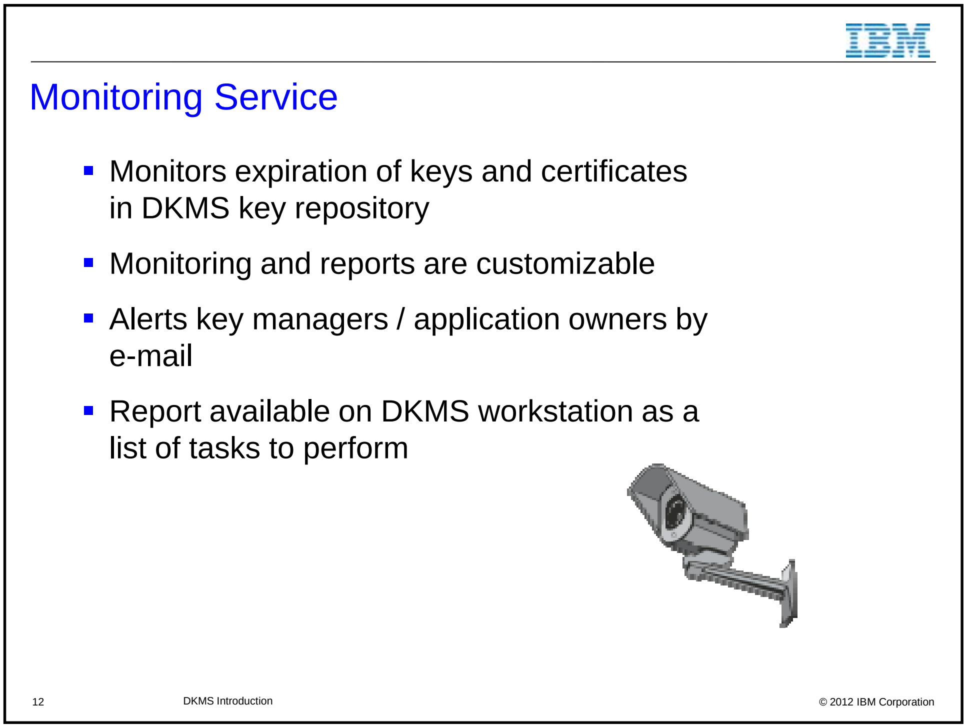
DKMS agent

Customer Apps.

DKMS APIs

IBM CCA

DB2

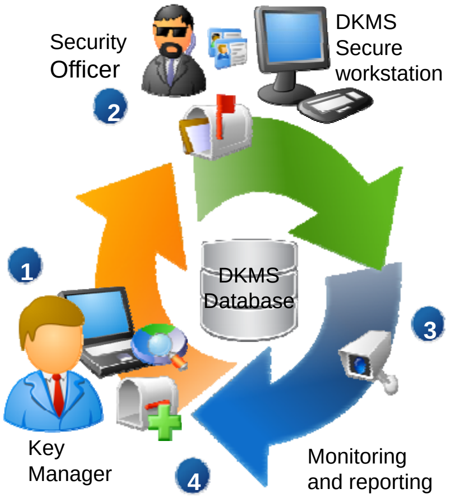


Monitoring Service

- Monitors expiration of keys and certificates in DKMS key repository
- Monitoring and reports are customizable
- Alerts key managers / application owners by e-mail
- Report available on DKMS workstation as a list of tasks to perform

Key Management Workflow

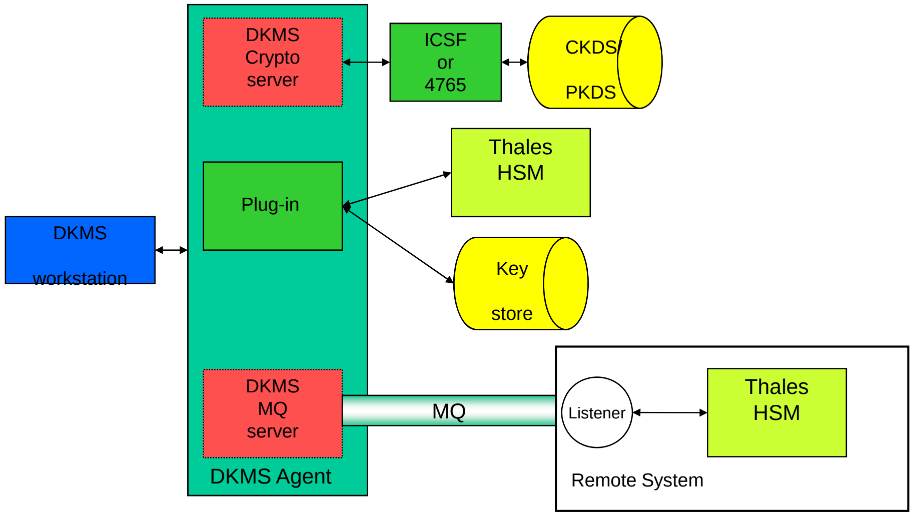
- Efficient key management through automated monitoring
  1. Request new key, key renewal, or key revocation
  2. Requested tasks performed by security officer
  3. Monitoring and Reporting – e-mail & DKMS Browser
  4. Analysis leads to new key management requests



Support for Thales/Racal HSMs

- Key transport from DKMS WS to server:
  - DKMS generates keys under a ZMK and the keys are imported at the Thales side under an LMK
  - DKMS generates keys directly under an LMK
- Some customer/application dependent code is needed
  - Access to application-owned key store
  - Access to Thales HSM

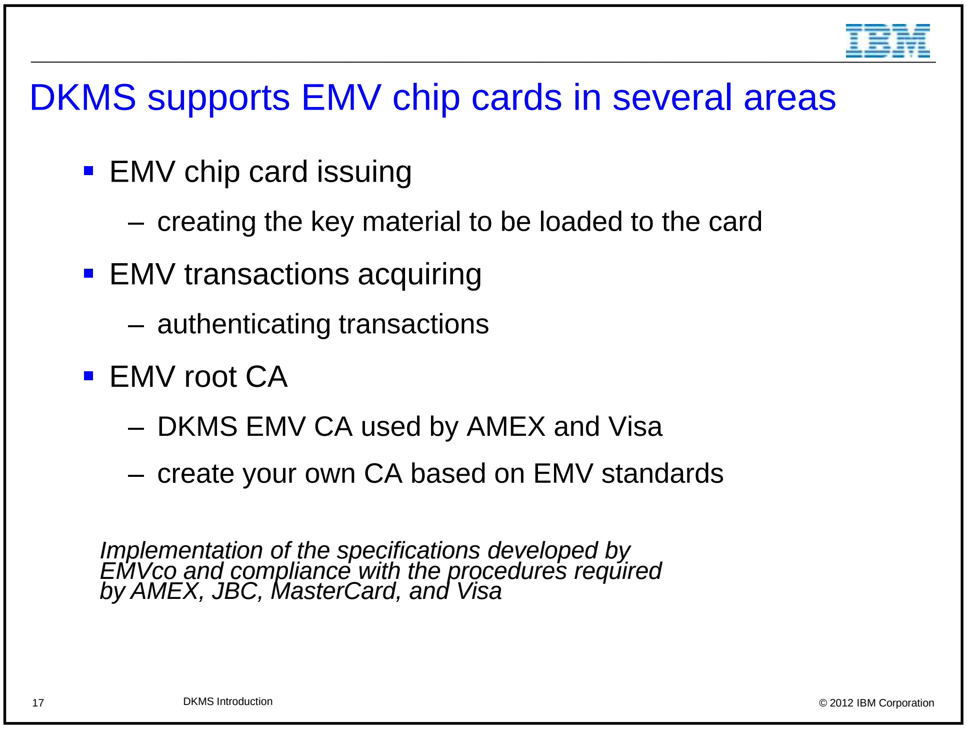
Support for Thales/Racal HSMs



EMV

Chip card issuing and transaction acquiring

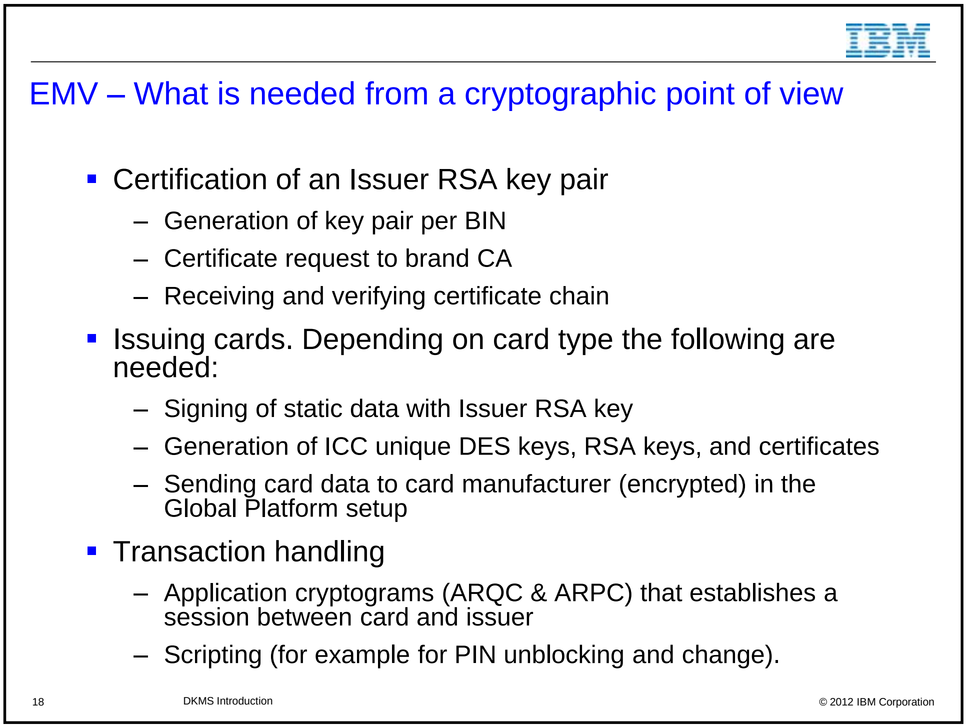
EMVco specifications



DKMS supports EMV chip cards in several areas

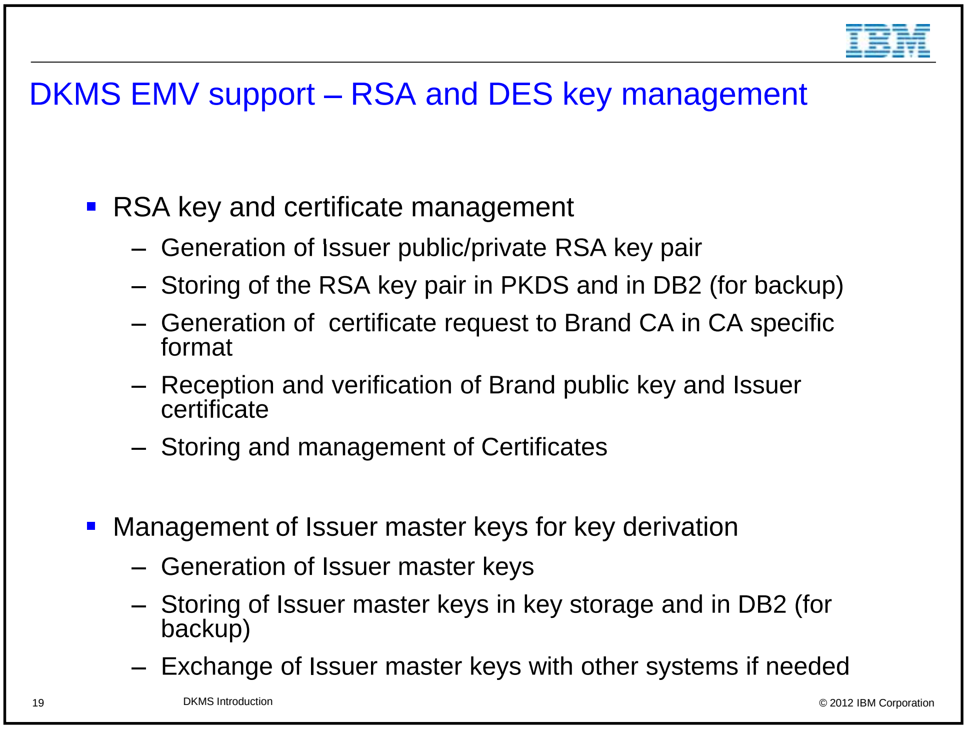
- EMV chip card issuing
  - creating the key material to be loaded to the card
- EMV transactions acquiring
  - authenticating transactions
- EMV root CA
  - DKMS EMV CA used by AMEX and Visa
  - create your own CA based on EMV standards

Implementation of the specifications developed by EMVco and compliance with the procedures required by AMEX, JBC, MasterCard, and Visa



EMV – What is needed from a cryptographic point of view

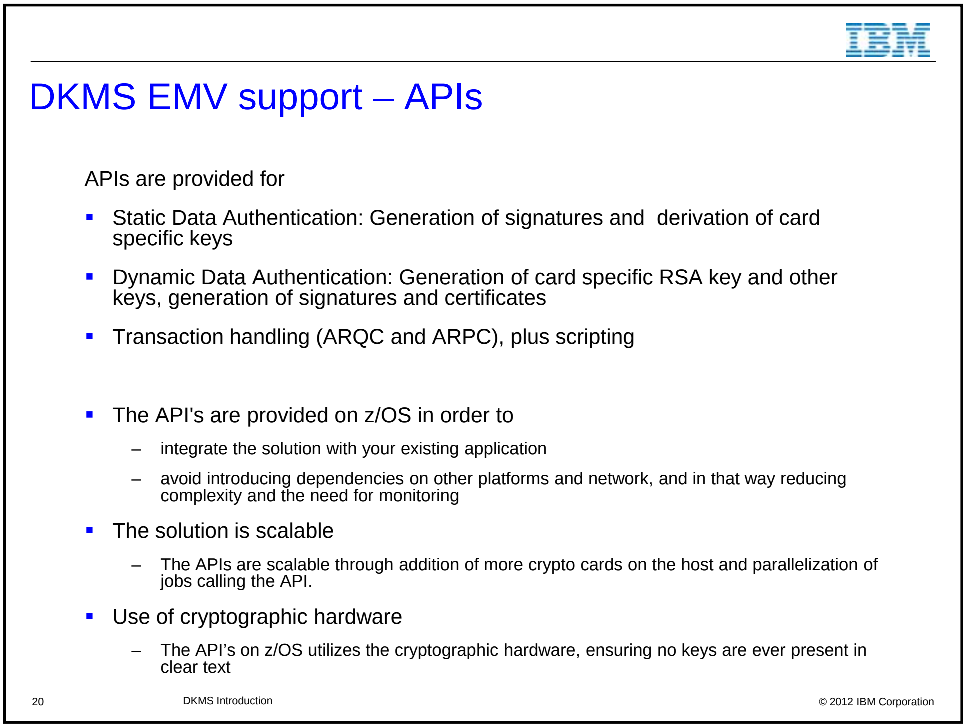
- Certification of an Issuer RSA key pair
  - Generation of key pair per BIN
  - Certificate request to brand CA
  - Receiving and verifying certificate chain
- Issuing cards. Depending on card type the following are needed:
  - Signing of static data with Issuer RSA key
  - Generation of ICC unique DES keys, RSA keys, and certificates
  - Sending card data to card manufacturer (encrypted) in the Global Platform setup
- Transaction handling
  - Application cryptograms (ARQC & ARPC) that establishes a session between card and issuer
  - Scripting (for example for PIN unblocking and change).



DKMS EMV support – RSA and DES key management

- RSA key and certificate management
  - Generation of Issuer public/private RSA key pair
  - Storing of the RSA key pair in PKDS and in DB2 (for backup)
  - Generation of certificate request to Brand CA in CA specific format
  - Reception and verification of Brand public key and Issuer certificate
  - Storing and management of Certificates

- Management of Issuer master keys for key derivation
  - Generation of Issuer master keys
  - Storing of Issuer master keys in key storage and in DB2 (for backup)
  - Exchange of Issuer master keys with other systems if needed

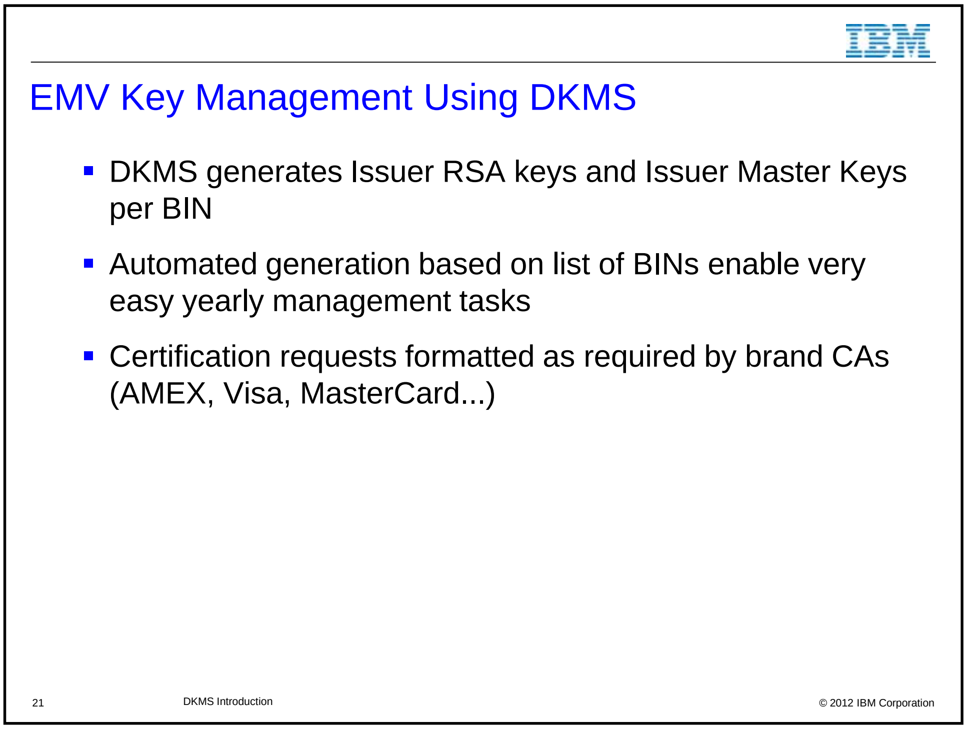


DKMS EMV support – APIs

APIs are provided for

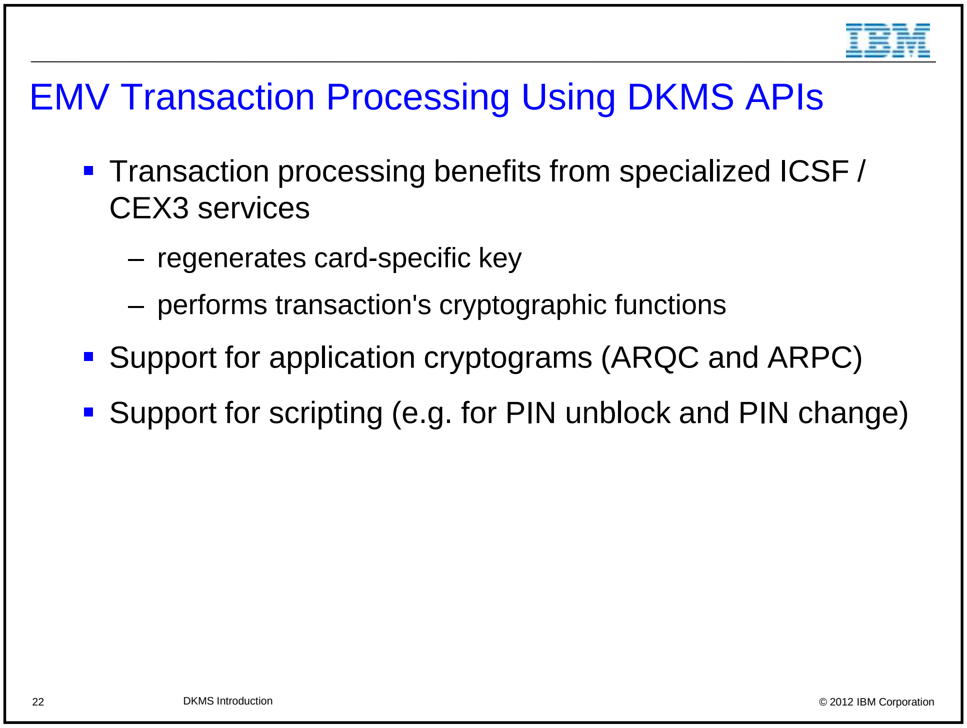
- Static Data Authentication: Generation of signatures and derivation of card specific keys
- Dynamic Data Authentication: Generation of card specific RSA key and other keys, generation of signatures and certificates
- Transaction handling (ARQC and ARPC), plus scripting

- The API's are provided on z/OS in order to
  - integrate the solution with your existing application
  - avoid introducing dependencies on other platforms and network, and in that way reducing complexity and the need for monitoring
- The solution is scalable
  - The APIs are scalable through addition of more crypto cards on the host and parallelization of jobs calling the API.
- Use of cryptographic hardware
  - The API’s on z/OS utilizes the cryptographic hardware, ensuring no keys are ever present in clear text



EMV Key Management Using DKMS

- DKMS generates Issuer RSA keys and Issuer Master Keys per BIN
- Automated generation based on list of BINs enable very easy yearly management tasks
- Certification requests formatted as required by brand CAs (AMEX, Visa, MasterCard...)

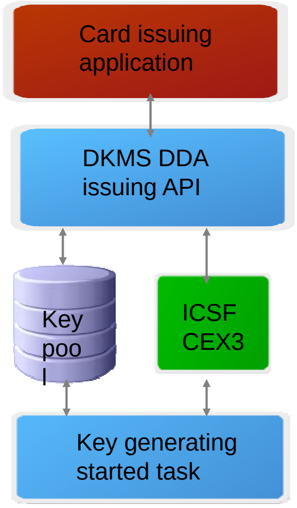


EMV Transaction Processing Using DKMS APIs

- Transaction processing benefits from specialized ICSF / CEX3 services
  - regenerates card-specific key
  - performs transaction's cryptographic functions
- Support for application cryptograms (ARQC and ARPC)
- Support for scripting (e.g. for PIN unblock and PIN change)

EMV Chip Card Data Generation

- Support for SDA, DDA, and CDA chip cards
- Generates card unique DES keys, RSA keys, and certificates
  - RSA key generation is time consuming
  - RSA keys generated in advance to a pool
  - utilizing spare capacity during off-peak hours
- Signs data with Issuer RSA key
- Prepare data to card manufacturer (Global Platform support)

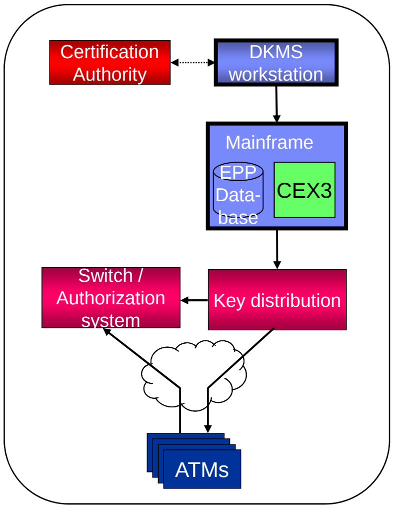


DKMS Extension

.

ATM Remote Key Loading

- Electronic distribution of Terminal Master Keys (TMKs)
- Replaces manual handling of TMK key parts
  - saves cost
  - eliminates errors
- Described in ANSI X9.24 part 2
- Is supported by the major ATM vendors Diebold, NCR, and Wincor Nixdorf



PIN Distribution

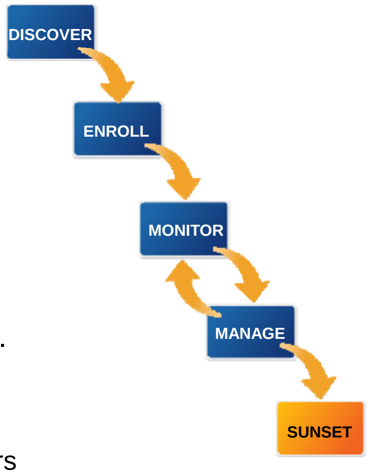
- PIN Print
  - Flexible and secure print of PIN mailers
  - Low cost alternative for small and medium capacity
- Web based PIN Management
  - allows internet-banking clients to view and optionally change PIN
  - designed to seamless integration with existing web-banks
  - end-to-end encryption from crypto HW to client browser
  - very cost efficient compared to PIN mailers

X.509 Certificate Management – the Problem

- Risk of service unavailability due to expired certificates
- For every certificate dependant application:
  - Generate keys and request certificates
  - Receive and install certificates
  - Keep track of certificate expiration date !!
- Tools and procedures depend on application and server type
- Sparse access control and audit functions
- Cost of certificates

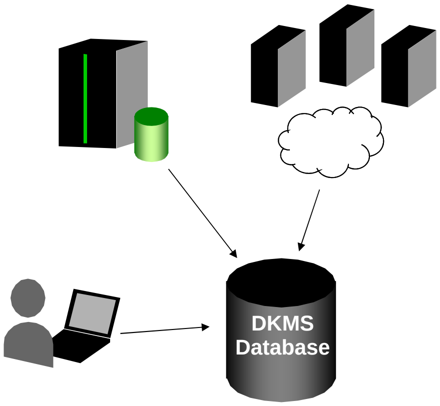
Certificate Management

- Centralized management of certificates
- Expiry monitored and managed:
  Avoid certificate expiry
- Efficient work flow
  - certificate managers takes care of monitoring reports and request certs
  - System administrators install certs and restart service
- Focused on z/OS cert management
  - RACF Key Ring, MQ, System SSL...
  - Tight interaction with z/OS PKI Services
  - off-line support for distributed servers



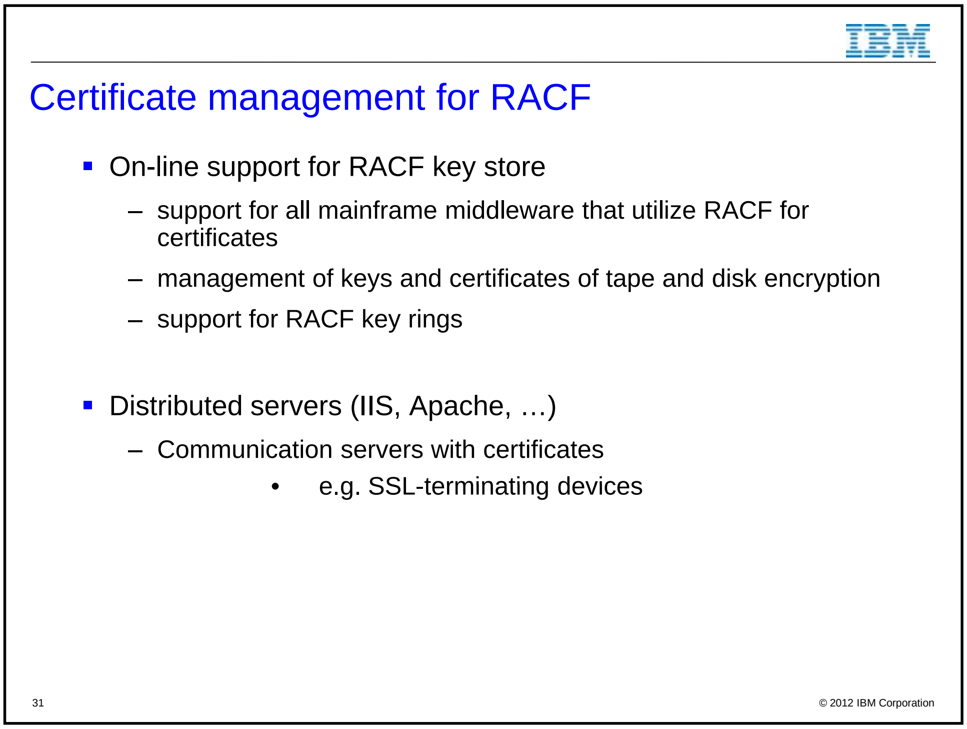
Discover and Enroll Existing Certificates

- **Scan network and discover certificates in the distributed environment and enroll**
- **Discover certificates in RACF key stores and enroll**
- **Manual enrollment of existing certificates**
- **Customized tools importing existing databases**



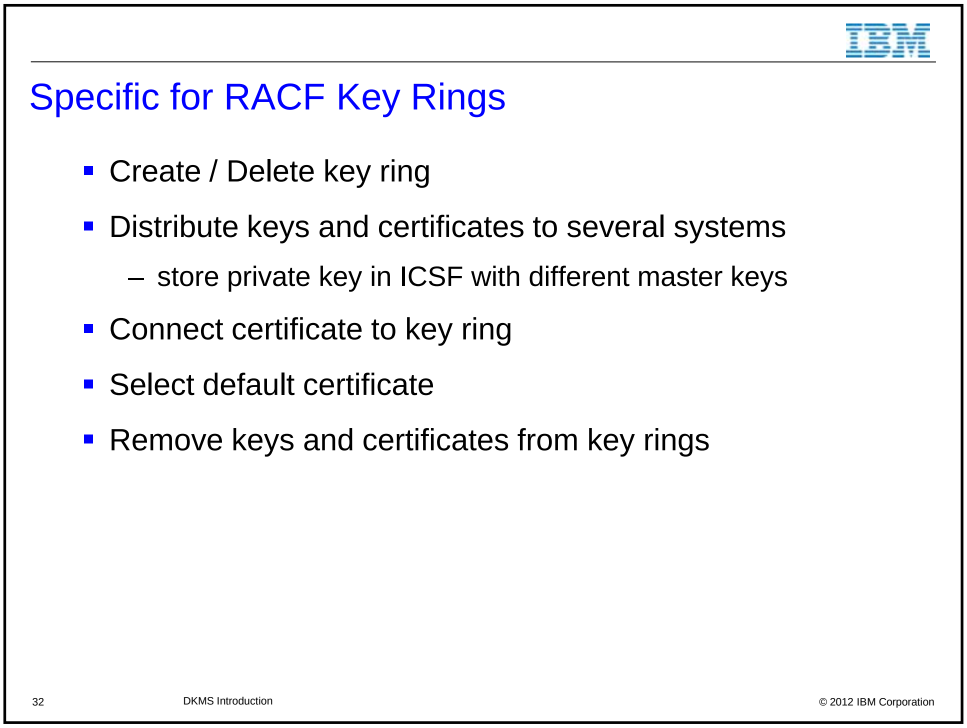
Certificate Management – DKMS Functions

- Search for expiring certificates
- Generate RSA key pair
- Create certificate request and send request to CA
- Receive certificate (chain) from CA
- Verify and store certificates in DKMS database
- Certificate installation depends on server type



Certificate management for RACF

- On-line support for RACF key store
  - support for all mainframe middleware that utilize RACF for certificates
  - management of keys and certificates of tape and disk encryption
  - support for RACF key rings
- Distributed servers (IIS, Apache, …)
  - Communication servers with certificates
    - e.g. SSL-terminating devices



Specific for RACF Key Rings

- Create / Delete key ring
- Distribute keys and certificates to several systems
  - store private key in ICSF with different master keys
- Connect certificate to key ring
- Select default certificate
- Remove keys and certificates from key rings

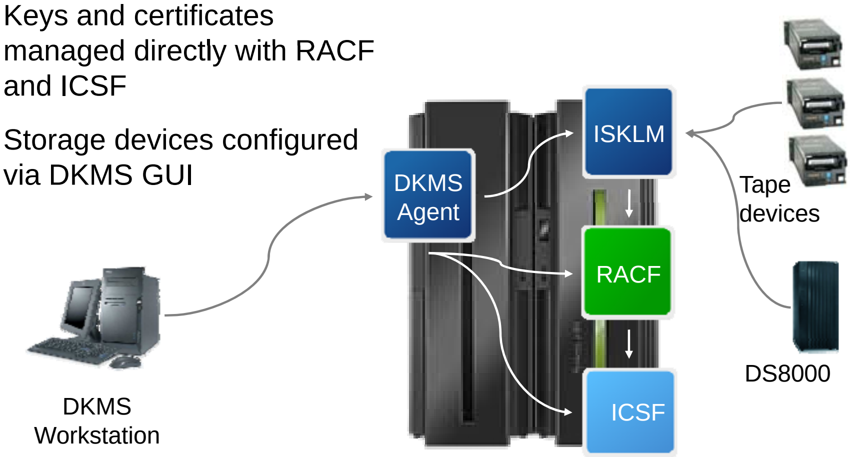
SSL Key Management – Benefits

- Key and certificate management for servers including timely renewals
- Centralized operations at DKMS reduces the amount of required resources
- High security based on DKMS access-control
- All operations logged in DKMS audit log

- Cost reduction and more reliable management of keys and certificates.

DKMS Support for Tape and Disk Encryption

- Integration with ISKLM
- Keys and certificates managed directly with RACF and ICSF
- Storage devices configured via DKMS GUI



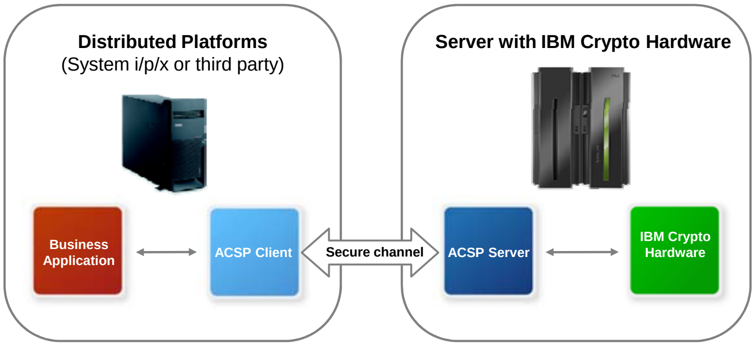
Advanced Cryptographic Service Provider

Remote encryption for System z and other platforms

The ACSP Concept

- Replace HSMs installed in distributed servers with a Net HSM
  - Utilize mainframe crypto capacity as the Net HSM
    Other servers like Linux servers supported as well
  - Expose crypto functions to client applications
- Benefits
  - Cost effective use of available crypto capacity
  - Reduced administration and simpler key management
  - Crypto support for platforms with no native IBM crypto HW support
  - High scalability, reliability, and availability

ACSP Components



- ACSP client platforms
  - AIX, Linux, Windows
  - (in reality any platform that supports Java)
- ACSP client APIs
  - CCA in Java and C
  - PKCS#11 basic set

- Transport network
  - TCP
  - MQ
  - SSL/TLS protected

    server/client authentication

- ACSP server platform
  - z/OS, zLinux, Linux
  - CEX2, CEX3, 4764, 4765

End of presentation

enachtig@ca.ibm.com