IBM Americas ATS, Washington Systems Center

ICSF Update
Session #7997

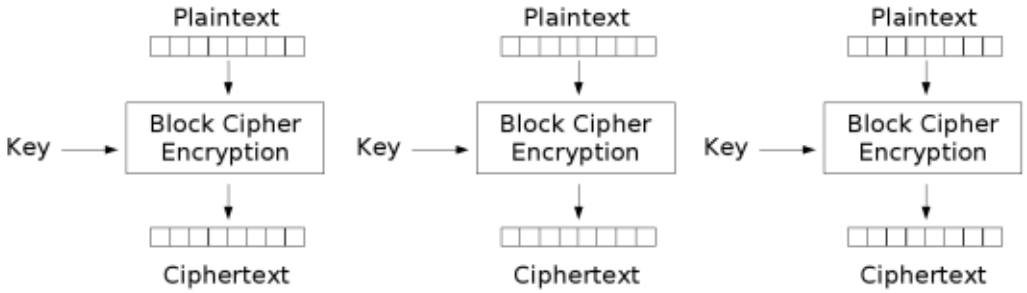
Greg Boyd
boydg@us.ibm.com

Agenda

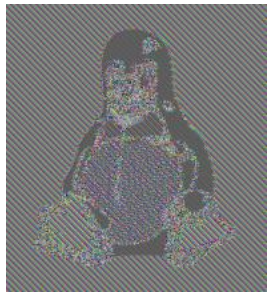
- **zEnterprise 196 Hardware**
  - CPACF
  - CEX3
- **ICSF**
  - HCR7780
  - FIPS SPE
  - Toleration and Migration
- **VM and Linux**
- **TKE 7.0**

z196 Hardware - CPACF

- **MSA-4 (Message Security Assist 4)**
  - New instructions for additional chaining options (CFB, OFB, Counter Modes)
  - New option for existing instructions (XTS-AES)



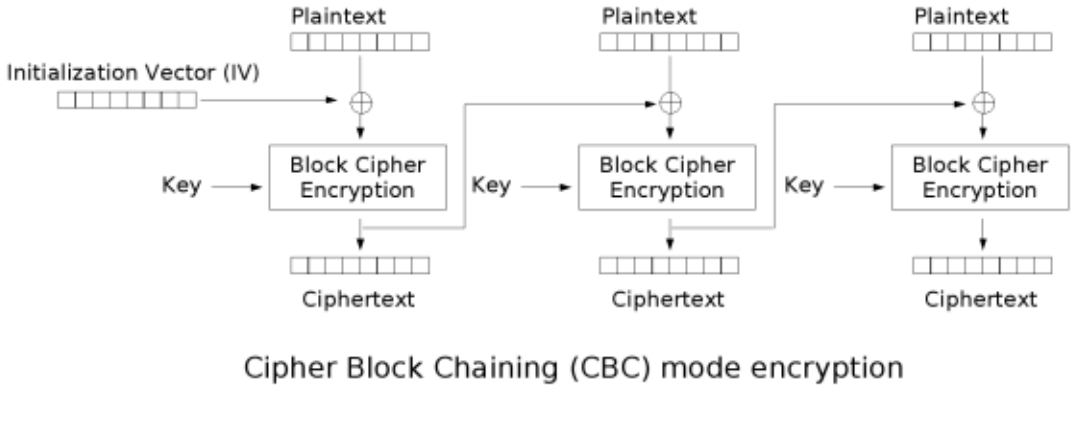
Electronic Codebook (ECB) mode encryption



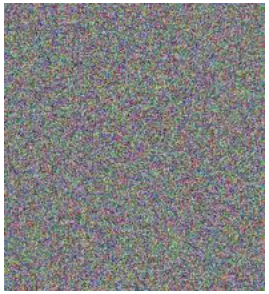
Images from Wikipedia

z196 Hardware - CPACF

- **MSA-4 (Message Security Assist 4)**
  - New instructions for additional chaining options (CFB, OFB, Counter Modes)
  - New option for existing instructions (XTS-AES)



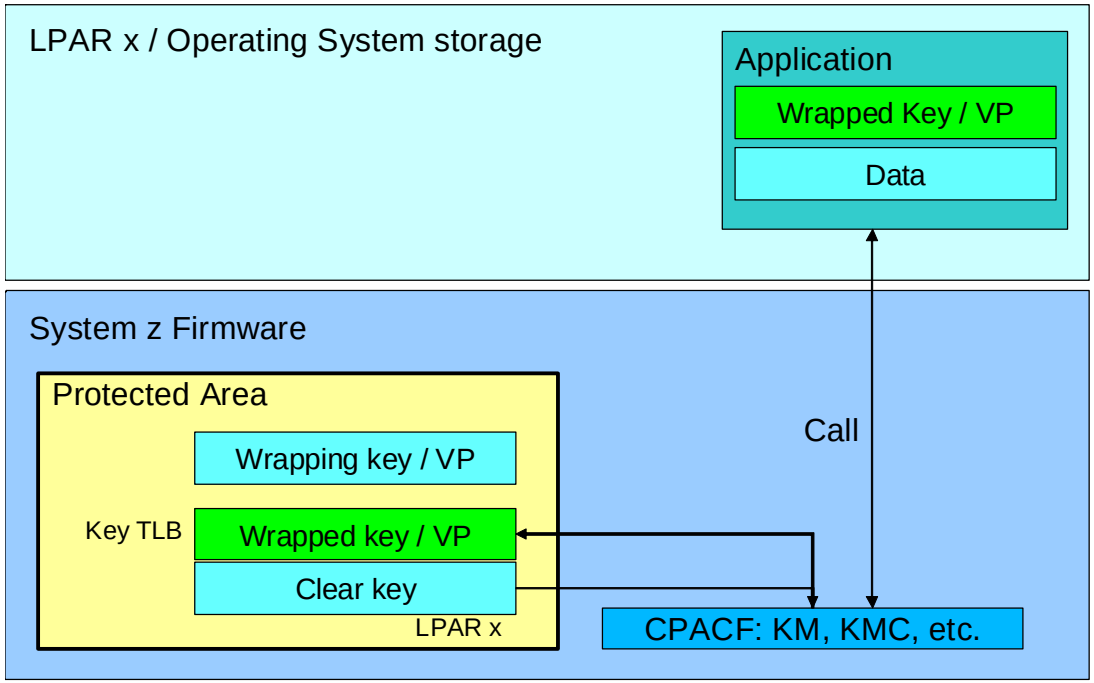
Cipher Block Chaining (CBC) mode encryption



Images from Wikipedia

z196 Hardware - CPACF

- **MSA-3 (Message Security Assist 3)**
  - Became available on the GA3 of the z10 EC/GA2 of the z10 BC
    - Protected Key Support



Protected Key – How it works

- **Create a key, with the value 'ABCD' and store it as a secure key in the CKDS (i.e. encrypted under the Master Key, MK)**
  - E_{MK}(x'ABCD') => x'4A!2' written to the CKDS and stored with a label of MYKEY
- **Execute CSNBSYE (the clear key API to encrypt data), but pass it the key label of a secure key, MYKEY; and text to be encrypted of 'MY MSG '**
  - CALL CSNBSYE(….,

    MYKEY,

    'MY MSG ' ….)

Protected Key – How it works (cont …)

- **ICSF will read MYKEY from the CKDS and pass the key value x'4A!2' to the CEX3**
- **Inside the CEX3, recover the original key value and then wrap it using the wrapping key**
  - D_{MK}(x' 4A!2') => x' ABCD'
  - E_{WK}(x'ABCD') => x'*94E'
- **ICSF will pass the wrapped key value of x'*94E' to the CPACF, along with the message to be encrypted**
- **In the CPACF, we'll retrieve the wrapping key, WK**
  - D_{wk}(x' *94E') => x' ABCD'
  - $E_{x'ABCD'}$('MY MSG ') => ciphertext of x' 81FF18019717D183'

Suite B

- **Symmetric Encryption**
  - AES w/key sizes of 128 and 256
- **Digital Signatures**
  - ECDSA
- **Key Agreement**
  - ECDH
- **Message Digest**
  - SHA-2 (SHA-256 and SHA-384)

http://www.nsa.gov/ia/programs/suiteb_cryptography/

z196 Hardware – CEX3

- **Elliptic Curve Support**
  - ECDSA
  - New ECC Master Key
- **Effective Key Size Security**

RSA Key Size	ECC Key Size
1024	163
3072	256
7680	384
15360	512

- Point multiplication $Q=kP$
- Repeated point addition and doubling: $9P = 2(2(2P)) + P$
- Public key operation: $Q(x,y) = kP(x,y)$
  Q = public key
  P = base point (curve parameter)
  k = private key
  n = order of P
- Elliptic curve discrete logarithm
  Given public key kP, find private key k
- Best known attack: Pollard's rho method with running time: $\frac{(\pi n)^{1/2}}{2}$

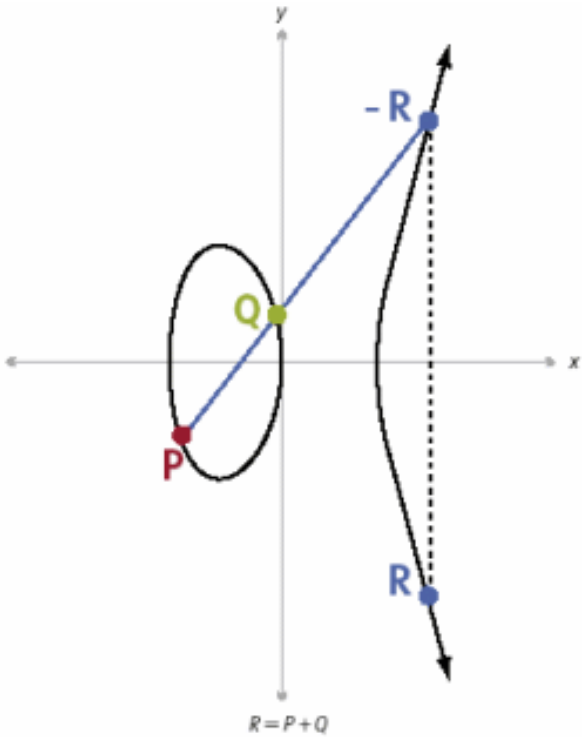


Image from DeviceForge and other sites

z196 Hardware – CEX3

- **CEX3**
  - ANSI X9.8 (PIN Processing)
  - ANSI X9.24 (CBC Key Wrapping)
  - HMAC (w/APAR OA33260 in 1Q2011)
  - Concurrent Patch Apply (CPA) / Concurrent Driver Upgrade (CDU)

ICSF – HCR7780

- **MSA-4, MSA-3**
- **Elliptic Curve Support**
  - New 256-bit ECC Master Key
- **ANSI X9.8 PIN**
- **ANSI X9.24 (CBC Key Wrapping)**
  - Original vs Enhanced
- **HMAC**
- **TKE 7.0**

ICSF – HCR7780

- **FIPS Mode SPE for PKCS #11 – Public Key Cryptographic Token Interface**
  - PKCS #11 provides APIs for talking to devices which hold crypto info or perform crypto operations (think Smart Cards)
  - FIPSMODE was an option in HCR7770
  - SPE provides additional support required for FIPS certification
- **CKDS Constraint Relief**
  - CKT, in-storage copy of CKDS, above the bar
  - Optimized for speeding up searches
  - Limit performance impact of bulk updates
    - Buffering Read-Aheads
    - Tighten allocate / open / IO / close / deallocate process

ICSF – HCR7780

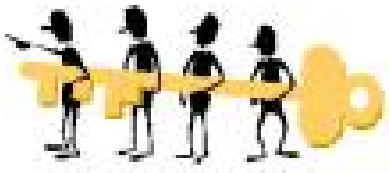
- **PCI Audit**
  - Several current subtypes will have additional info
    - RACF Userid
    - Connect Group
    - Certificate Issuer's Distinguished Name
    - Certificate Subject's Distinguished Name
    - Registry that authenticated the user
    - Jobname, Job Entry Date & Time
    - Terminal ID
    - Security Label
    - User-defined Identification Field
  - New SMF Type 82 Subtype 29 (TKE Offload)

- **AMODE 64 Support**

ICSF Versions supported on z196

- **ICSF FMIDs**
  - HCR7780 (www.ibm.com/systems/z/os/zos/downloads)
  - HCR7770 (z/OS V1.12)
  - HCR7751 (z/OS V1.11)
  - HCR7750 (z/OS V1.10)
  - HCR7740 (z/OS V1.9 with IBM Lifecycle Extension with PTFs)
  - HCR7731 (z/OS V1.8 with IBM Lifecycle Extension with PTFs)
  - HCR7731 (z/OS V1.7 with IBM Lifecycle Extension with PTFs)*

    (*note that z/OS V1.7 included HCR7720, but HCR7720 will not support the z196, you must have upgraded to HCR7731 or later on your z/OS 1.7 system)

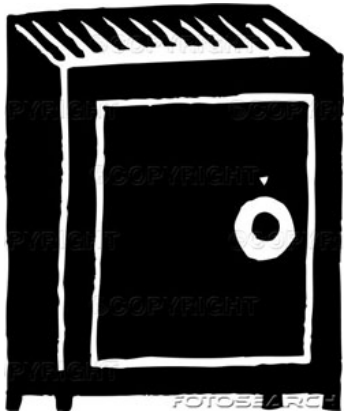


Crypto Express3 Support

- **Crypto Express3 Toleration APARs**
  - ICSF OA29839
  - RMF OA28670
  - SAF OA29194
  - RACF OA29193

ICSF Toleration

- **Toleration APAR OA33320**
  - CBC Key Wrapping – 'Enhanced' key wrapping
  - ECDSA Keys in the PKDS
- **HMAC Support OA33260 (1Q2011)**
  - No toleration support, but all versions of ICSF must be on HCR77780 before you can start using it



z/VM 5.4 and z/VM 6.1

- **Provides guest support, VM does not directly use the crypto hardware**
  - Crypto Express3 - VM64656
  - Protected Key Support - VM64793

Linux on System z

- **CPACF**
  - MSA-4 support in a future distribution
- **CEX3**
  - Drivers in SUSE SLES10 SP3 and SLES11 and Red Hat RHEL 5.4 provide toleration support (CEX3 acts like a CEX2)
  - CCA (secure key support) software download at the CryptoCards website (http://www.ibm.com/security/cryptocards/?S_TACT=107AG01W&S_CMP=campaign)
- **Thin interrupts**
  - already supported in Novell SUSE SLES 10 SPE and Red Hat RHEL 5.4

TKE 7.0 – New hardware platform

- **TKE 7.0 will run on a new hardware platform**
  - 4765 (CEX3) Crypto Card
  - Add USB ports; Drop serial ports
    - Old Kobil Smart Card readers used a serial port
    - New Omnikey Smart Card readers use the USB
    - Support USB Flash Memory Drive (as an alternative to the DVDRAM media)
  - New Smart Cards
    - JCOP41 NXP Smart Cards replacing the older Data Key Smart Cards
    - Six digit PINs

TKE 7.0 - New Key Support

- **Support ECC Master Keys**
  - 32-byte AES Key to protect ECC Keys
  - Generation and loading of ECC keys not supported on TKE 7.0
- **CBC Key Wrapping**
  - KW-ENH Key Wrapping Enhanced
  - KW-ORIG Key Wrapping Original

TKE 7.0 - Migration Wizard

- **TKE 6.0 introduced a configuration migration utility to automate the process of replacing a host crypto adapter**
  - Captured public configuration data
    - Roles
    - Authorities
    - Domain Control Settings
    - Only ‘public’ (non-secret data), no key material
- **TKE 7.0 adds support for migration of key material**
  - Master Keys only
  - New Smart Card Types
    - Migration CA (MCA)
    - Injection Authority (IA)
    - Key Part Holder (KPH)

TKE 7.0 - Audit Offload

- **Payment Card Industry Data Security Standards (PCI-DSS) driving new requirements**
  - With TKE 5.3 we provided additional logging for security-relevant events on the TKE
  - These records can be written to the DVD for post processing
- **With TKE 7.0 we'll support sending those records to a specified z/OS Host, using SMF**

Summary

- **z196 continues the implementation and support of new crypto technology, techniques and standards to support the evolving world of data security**