Practical Uses of S/390 Hardware Cryptographic Support

Session #: 149

Marilyn Allmond
Advanced Technical Support
allmond@us.ibm.com

Disclaimer

WSC
Advanced Technical Support

Trademarks

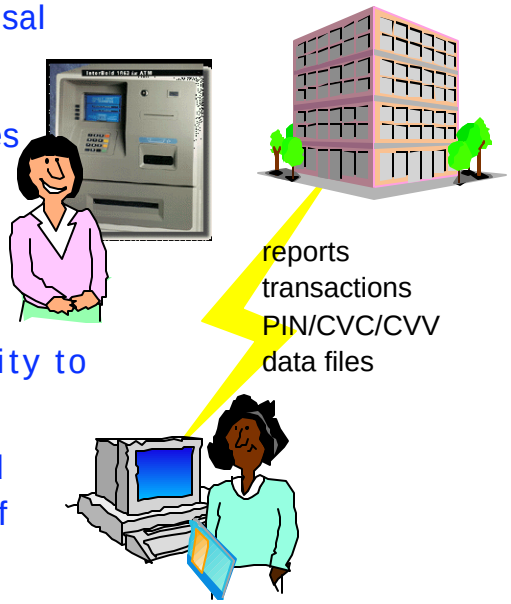
WSC Advanced Technical Support

- **The following are trademarks or registered trademarks of the International Business Machines Corporation:**
  - CICS
  - DB2
  - OS/390
  - RACF
  - S390
- **The following are trademarks or registered trademarks of RSA Data Security, Incorporated:**
  - BSAFE
  - RSA
- **VISA is a registered Trademarks of Visa in the United States and other countries.**
- **MASTERCARD is a registered Trademarks of MasterCard International Incorporated in the United States and other countries.**

Why use cryptography?

WSC Advanced Technical Support

- Protects data in the open, universal space of "electronic flow"
- Legal & Corporate responsibilities
  - To protect data privacy
  - To detect modification of data
  - To provide extra levels of security to common functions
  - To provide digital signatures and support electronic verification of identity via digital signatures
  - Combinations of any and all of the above



Practical Uses: Categories

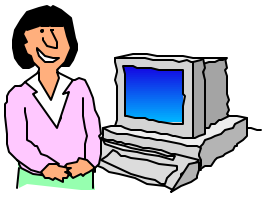
WSC Advanced Technical Support

- Your own application, making cryptographic function requests via
  - ICSF APIs
  - Open Cryptographic Services Facility APIs
- Some other IBM application product making cryptographic function requests as part of its normal task and using the S/390 crypto hardware with or without your explicit request. Such as,
  - Web Server (HTTP, WAS)
  - eCommunications Server
    - TN3270e with Host-On-Demand
    - IPSec in VPN
  - IDCAMS Repro

Who are you communicating with?

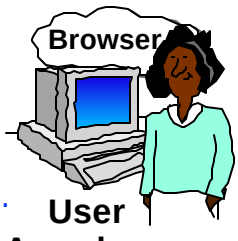
WSC Advanced Technical Support

- The system is known to yours or you can make requests for system changes, if necessary.
  - S/390 Firewall
  - S/390 VPN
  - S/390 TN3270e
  - S/390 Web Server or some other SSL-based application
  - ICSF APIs-based application



User Known

- The system is not known to yours and you do not want any special requirements for the user beyond having browser capability.
  - S/390 Web Server or some other SSL-based application



User Anywhere

To Write Code, Have System Pre-reqs or Not?

WSC Advanced Technical Support

- Just want to send a file encrypted %&!!!
- Sorry, but it is not as simple as that
- No standard nor 'de facto' standard for secure FTP
- OK, encrypt on your end and then FTP the encrypted file
- Good, but how does the other end know what algorithm you used to perform the encryption? How do you know the other end has a way to perform that algorithm? etc.

Known User; Some control over system

WSC Advanced Technical Support

- **S/390 IPSec within the Firewall Technologies**
  - Communication can be secured by using IPSec (fwtunnl command)
    - Define a tunnel between you and the system you wish to communicate with
    - Define the Security Association desired via command options (policy, encrypthow, algorithm)
- Communicate Host-to-Host and Host-to-Client
  - S/390 Host-to-S/390 Host easily
  - Host-to-Client requires a non-S/390 client that supports IPSec
- Requires setup performed on both systems
- ICSF and Crypto hardware are used transparently by the S/390 Firewall technology code to perform some of the IPSec cryptographic functions during Firewall or VPN usage.

Known User; . . .

WSC Advanced Technical Support

- **S/390 TN3270e** emulation flow with secure data flow between host and workstation client using Host-On-Demand
  - Communication can be secured by using SSL (fwtunnl command)
    - Configure the ports
    - Define SSL and non-SSL ports
    - Define the SSL environment parameters
- Communicate Host-to-Host and Host-to-Client
  - Host-to-Client requires a Host-On-Demand (HOD) client (available for both S/390 and workstations)
  - S/390 TN3270e is server and S/390 HOD is client code
- Requires setup performed on both systems
- ICSF and Crypto hardware can be used transparently by the S/390 TN3270e code to perform some of the SSL cryptographic functions.

Known User; . . .

WSC Advanced Technical Support

- **S/390 application written with OS/390 ICSF**
  - Communication can be secured by using ICSF APIs to request desired functions
    - Enable and setup hardware cryptographic coprocessors
    - Define the ICSF environment and system data sets
    - Activate the hardware with hardware master keys
- Communicate Host-to-Host and Host-to-Client fairly easily assuming
  - communicating parties support either DES or Triple-DES for encryption/decryption
  - packaging arrangement of data is agreed upon by all parties
- Requires setup performed on all systems
- This is true for applications written using OS/390 Open Cryptographic Services Facility (IBM's implementation of CDSA)

Known User; . . .

WSC Advanced Technical Support

- **S/390 application written with BSAFE**
  - Can use BSAFE™ type calls to request ICSF and crypto hardware usage!!
  - BSAFE CCA code is shipped with ICSF in the SCSFOBJ data set
- **To use ICSF in this code environment do the following**
  - Set up the BSAFE Choosers; adding a chooser for CCA hardware
  - Set up other normal BSAFE variables
  - Check for crypto hardware existence via QueryCrypto
  - Select Session Chooser based on results
  - CCA AI, AM, and KI types are listed in ICSF Application Programmer's Guide with the exception of 2 new AMs as of R8
    - AM_TOKEN_RSA_ENCRYPT
    - AM_TOKEN_CRT_DECRYPT
  - Link the application with both normal BSAFE OBJ libraries and the ICSF ACSFOBJ library by specifying in the pre-linker step

Unknown User; No control over system

WSC Advanced Technical Support

- **Applications that use**
  - standards that have common clients available in most browsers
- OS/390 Web Server
- eCommunications Manager, TN3270e and HOD
- Requires minimal setup performed on both systems
- ICSF and Crypto hardware can be used transparently by the S/390 applications.

Unknown and Known User

WSC Advanced Technical Support

- **S/390 HTTP Server** (SSL)
  - Communication can be secured by using SSL (fwtunnl command)
    - Define SSL Port and Indicate SSL usage
    - Define the SSL environment parameters
- Communicate Host-to-Host and Host-to-Client
  - S/390 Host-to-S/390 Host easily
  - Host-to-Client easily with most browsers supporting SSL
- Requires setup performed on host system
- ICSF and Crypto hardware can be used transparently by the S/390 HTTP Server code to perform some of the SSL cryptographic functions.

Unknown and Known User . . .

WSC Advanced Technical Support

- **S/390 application written with OS/390 System SSL**
  - Communication can be secured by using SSL
    - UNIX environment
    - Code application using System SSL DLLs
    - Define the SSL environment parameters
- Communicate Host-to-Host and Host-to-Client
  - Host-to-Host easily assuming communicating host application supports SSL
  - Host-to-Client easily with most browsers supporting SSL
- Requires setup performed on host system
- ICSF and Crypto hardware can be used transparently by the S/390 System SSL code to perform some of the SSL cryptographic functions.

Just a File - To Be Sent Securely

WSC Advanced Technical Support

- **Encrypt before FTPing**
  - Partner receives normal FTP and receiver must decrypt
  - Decryption implies
    - Shared key communicated between sender and receiver
    - Receiver has a crypto engine capable of performing encryption using an agreed upon algorithm
- **Use IDCAMS REPRO for Host-to-Host**
  - Requires running ICSF in COMPAT(YES) mode
- Write a small client-server application using a common API
  - Code could include packaging for key, data length, other parameters

IDCAMS REPRO; Using Clear Key in Command

WSC Advanced Technical Support

- **Encipher using a Clear Key in REPRO command**

```
//ENC    EXEC PGM=IDCAMS,REGION=4M
//SYSPRINT DD  SYSOUT=*
//SYSIN DD *
  REPRO                                    -
  INDATASET('CRYPTOB.TEST.CLEAR')          -
  OUTDATASET('CRYPTOB.TEST.CIPHER')        -
  ENCIPHER(PRIVATEKEY DATAKEYVALUE('01020102'))
//
```

- **Decipher using a Clear Key in REPRO command**

```
INDATASET('CRYPTOB.TEST.CIPHER')          -
OUTDATASET('CRYPTOB.TEST.CLEAR.AGAIN')    -
DECIPHER(DATAKEYVALUE('01020102'))
```

IDCAMS REPRO; Using Clear Key . . .

WSC Advanced Technical Support

- **==> Report from encipher JCL**

```
IDCAMS  SYSTEM SERVICES

  REPRO                                    -
  INDATASET('CRYPTOB.TEST.CLEAR')          -
  OUTDATASET('CRYPTOB.TEST.CIPHER')        -
  ENCIPHER(PRIVATEKEY
 DATAKEYVALUE('01020102'))
IDC0005I NUMBER OF RECORDS PROCESSED WAS 9
IDC0001I FUNCTION COMPLETED, HIGHEST
  CONDITION CODE WAS 0
IDC0002I IDCAMS PROCESSING COMPLETE. MAXIMUM
  CONDITION CODE WAS 0
```

IDCAMS REPRO; Using Enciphered Key

WSC Advanced Technical Support

- **Encipher using an encrypted key; Have IDCAMS generate a data key that will be protected by KEK (EXPORTER)** that is
  - shared with communication party
  - stored in CKDS with name KEKREPEX
  - printed in the IDCAMS report and data in its encrypted form (X'0969..)
- **Encipher JCL**

```
//ENC    EXEC PGM=IDCAMS,REGION=4M
//SYSPRINT DD  SYSOUT=*
//SYSIN DD *
  REPRO                                    -
  INDATASET('CRYPTOB.TEST.CLEAR')          -
  OUTDATASET('CRYPTOB.TEST.CIPHER.SYS')    -
  ENCIPHER                                 -
          (EXTERNALKEYNAME(KEKREPEX))
```

IDCAMS REPRO; Using Enciphered Key . . .

WSC Advanced Technical Support

- **Decipher using an encrypted key; KEK (EXPORTER)** must be stored in cryptographic key data set on receiving system with name KEKREPEX
- **Use encrypted key** as printed in the IDCAMS report (X'0969..)
- **Decipher JCL**

```
//DEC    EXEC PGM=IDCAMS,REGION=4M
//SYSPRINT DD  SYSOUT=*
//SYSIN DD *
  REPRO                                        -
  INDATASET('CRYPTOB.TEST.CIPHER.SYS')         -
  OUTDATASET('CRYPTOB.TEST.CLEAR.SYS')         -
  DECIPHER                                     -
         (SYSTEMKEY                            -
          SYSTEMDATAKEY(X'0969D3115EA7F2B4') -
          SYSTEMKEYNAME(KEKREPCK))
//
```

IDCAMS REPRO; Using Enciphered Key . . .

WSC Advanced Technical Support

- **Report from previous JCL**

```
IDCAMS  SYSTEM SERVICES

  REPRO                                        -
  INDATASET('CRYPTOB.TEST.CIPHER.SYS')         -
  OUTDATASET('CRYPTOB.TEST.CLEAR.SYS')         -
  DECIPHER                                     -
         (SYSTEMKEY                            -
          SYSTEMDATAKEY(X'0969D3115EA7F2B4') -
          SYSTEMKEYNAME(KEKREPCK))
IDC0005I NUMBER OF RECORDS PROCESSED WAS 7
IDC0001I FUNCTION COMPLETED, HIGHEST CONDITION
    CODE WAS 0
IDC0002I IDCAMS PROCESSING COMPLETE. MAXIMUM
    CONDITION CODE WAS 0
```

List of Practical APIs

WSC Advanced Technical Support

- **DES Encryption and Decryption**
  - DES or Triple-DES use CSNBENC and CSNBDEC
  - Specify key as token value; length within token determines DES or Triple-DES processing
- **RSA Encryption and Decryption**
  - For value no more than 128 bytes depending on modulus use CSNDPKE and CSNDPKD, format is PKCS #1.2
  - For a symmetric DES key 8, 16, or 24 bytes in length use CSNDSYG and CSNDSYI, format is PKCS#1.2 or zero-pad

List of Practical APIs . . .

WSC Advanced Technical Support

- **Hashing**
  - MAC, message authentication, CSNBMGN and CSNBMVR
    - APIs available for processing data within a data space; denoted by suffix of 1
  - SHA-1 and MD5, CSNBOWH
  - CVV or CVC, VISA™ and MasterCard™ functions for credit cards use CSNBCSG and CSNBCSV
- **Personal Identification Numbers for online access to accounts use**
  - CSNBPGN, CSNBCPA, CSNBPVR, and CSNBPTR
- **Digital Signatures, CSNDDSG and CSNDDSV**