S/390 Securing Data Transmissions

Mary Sweat
Advanced Technical Support
sweatm@us.ibm.com

Disclaimer

Advanced
Technical Support

The information contained in this document is distributed on as "as is" basis, without any warranty either express or implied. The customer is responsible for use of this information and/or implementation of any techniques mentioned. IBM has reviewed the information for accuracy, but there is no guarantee that a customer using the information or techniques will obtain the same or similar results in its own operational environment.

In this document, any references made to an IBM licensed program are not intended to state or imply that only IBM's licensed program may be used. Functionally equivalent programs that do not infringe IBM's intellectual property rights may be used instead. Any performance data contained in this document was determined in a controlled environment and therefore, the results which may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

It is possible that this material may contain references to, or information about, IBM products (machines and programs), programming, or services that are not announced in your country. Such references or information must not be construed to mean that IBM intends to announce such IBM Products, programming or services in your country.

IBM retains the title to the copyright in this paper as well as title to the copyright in all underlying works. IBM retains the right to make derivative works and to republish and distribute this paper to whomever it chooses.

Trademarks

- 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

Why Bother to Secure Data?

WSC
Advanced
Technical Support

- Privacy
- Trust
  - detect modification of data
- Access control
- Electronic payments
- Corporate security
  - legal & corporate responsibilities
- Provide digital signatures and support electronic verification of identity via digital signatures
- Countless other fields

- There are some very good reasons to use cryptography today. The best reason is to protect sensitive data as it flows across "unprotected and/or unknown space" electronically.

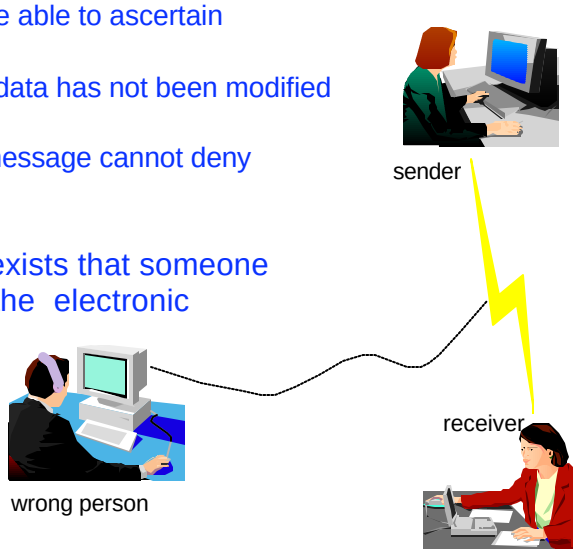
  Do we really know exactly where the data goes as it leaves and travels to some destination we indicate? What about all the various "paths and routings" that we may not know about?

- The next best reason for cryptography is to meet legal and corporate responsibilities, such as;
  1. data privacy
  2. data security
  3. authentication

Objective

WSC Advanced Technical Support

- Bottom Line
  - You want to send some data securely to a receiver and you want to ensure the data's;
    - confidentiality - only the intended party can read it
    - authentication - receiver should be able to ascertain origin of data
    - integrity - receiver can verify that data has not been modified in transit
    - nonrepudiation - the author of a message cannot deny authorship of that message
  - Without security the possibility exists that someone else opens the letter or hears the electronic communication

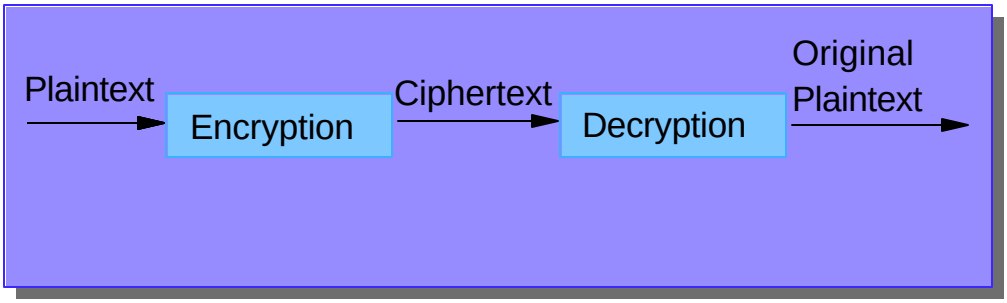


- With confidentiality an intruder should not be able to read or understand the data
- With authentication an intruder should not be able to masquerade as someone else
- With integrity an intruder should not be able to substitute a false message for a legitimate one
- With nonrepudiation you can verify a documents origin, similar to using witnesses and notaries to attest to the fact that people signing a paper contract did in fact sign it.

How Can We Meet the Objective?

WSC Advanced Technical Support

Use Cryptography - the art or science of keeping messages secret



- Cleartext or Plaintext is the message sent from the sender to receiver in readable form
- Encryption is scrambling or manipulating the contents of the message in such a way that it hides its contentsfrom outsiders
- Ciphertext is the encrypted message
- Decryption is the process of retrieving the Plaintext from the Ciphertext
- Algorithm is the procedure the defines the encryption/decryption process
- Key is the coding method that encryption and decryption usually use. The coding method is such that decryption can be performed only by knowing the proper key ; two types of keys, symmetric (secret key) and asymmetric (public key)
- Symmetric algorithms use the same key for encryption/ decryption. Asymmetric algorithms use a different key for encryption and decryption, they are also known as "public-key" algorithms.
- Asymmetric permits the encryption key to be public and the decryption key is held only by the proper receipt. The decryption key is called the private key or secret key.
- Generally, symmetric algorithms are much faster to execute on a computer than asymmetric ones because asymmetric or public algorithms are more mathematical intensive then symmetric algorithms.

S/390 Options

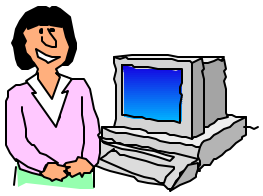
- Integrated Cryptographic Service Facility (ICSF) APIs
- Open Cryptographic Services Facility (OCSF) APIs
- System SSL (set of APIs for the SSL protocol)
- Other IBM application products that make cryptographic function requests as part of its normal task and use the S/390 crypto hardware with or without your explicit request.
  - WebSphere (HTTP, WAS servers)
  - eCommunications Server
    - TN3270e with Host-On-Demand
    - Virtual Private Networks (VPN)

- APIs - Application Program Interface
- OCSF is not a supported acryonim becuase the acronym belongs to someone else.

How to Choose???

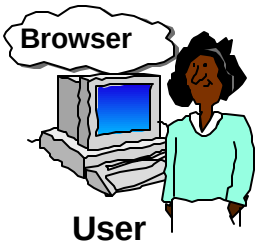
WSC Advanced Technical Support

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

- In order to determine which option to use we need to know how we want to initiate the data flow. If the system at the other end is one that we will communicate with a lot, we might want to have a semi-static means to flow data with privacy and/or content validation.

- Standards such as, Secure Sockets Layer (SSL) and IPSec, are great in this situations because with a minimum of system setup and configuration we have a way to have;
  1. a choice of algorithms and thus, can support a wider audience of participants
  2. a software crypto engine support without external requirements to limit in-house development requirements

- Standards give us flexibility to be able to communicate with anyone without the overhead of support requirements. Sometimes, though, the purpose of our business requires that we have a more specific type of control over the data and or data flow. When this occurs we might need to code our own application to meet our requirements and that might mean providing a client at the other end.

File Encryption

- 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.

- How cryptography works;
  both sender and receiver must use SAME ALGORITHM, SAME ALGORITHM OPTIONS, MATCHED KEYS, and sender and receiver must understand how that data is packaged.

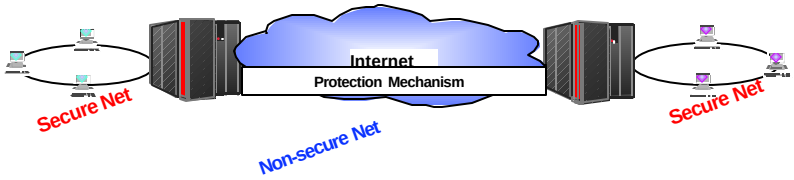
- A good example of this is where you just want to send an encrypted file to someone. Most people would want to use FTP (File Transfer Protocol). FTP is a standard means of transferring data and most people have the ability to use it.

- Unfortunately, there is no standard secure FTP protocol. You could write your own code to provide encryption and data content verification but you need the end result to be understood and viewable at the other end. So, you write a client, send it to each location to communicate with, maintain it, keep track of it, etc.

S/390 Virtual Private Network (VPN)

WSC Advanced Technical Support

- VPN is a mechanism that provides security when two or more secure networks communicate across a nonsecure network
  - S/390 VPN is part of the eNetwork Communication Server
  - configuration commands are part of OS/390 Firewall Technologies
- VPN provides secure communications based on IPSec protocol
  - IPSec is a set of protection mechanisms (protocols) used to implement VPNs over IP based networks
  - used in the IP layer of TCP/IP
  - provides security services to ensure;
    - encapsulation
    - integrity
    - confidentiality



- IETF - Internet Engineering Task Force
- PSec is defined by the IETF's IP Security Protocol Working Group (IPSec)
- IP Security Working Group is a group of people whose purpose in life is to define standards pertaining to how to protect traffic in an IP based network
- IPSec working group's home page; www.ietf.org/html.charters/ipsec-charter.html
- IPSec is essentially an encapsulation protocol, namely, one that allows you to place one packet inside another.
- OS/390 offers two types of VPNs;
  > manual, the attributes and encryption keys must be managed by administrative processes
  > dynamic, the attributes and encryption keys are managed by the Internet Key Exchange (IKE) protocoll

S/390 VPN Security

WSC Advanced Technical Support

- IPSec options for protecting data
  - IPSec's Authentication Header Protocol (AH) - RFC 2402
    - obsolete RFC 1826
  - IPSec's Encapsulation Security Protocol (ESP) - RFC 2406
    - obsolete RFC 1827

- Which do I use?

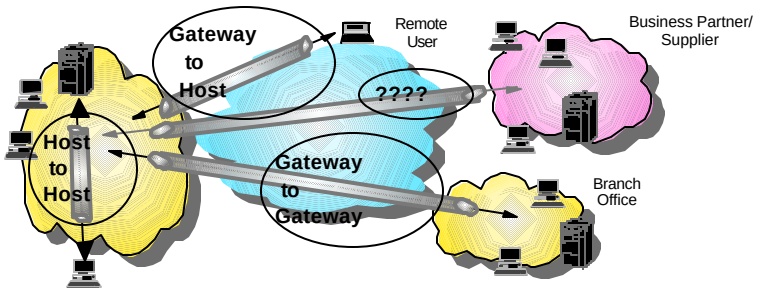
• I only need data privacy	Use the ESP protocol
• I only need data integrity and/or data origin authentication	Use either the AH or the ESP protocol with NULL encryption
• I want privacy, integrity and authentication	Use the ESP protocol

- When you want privacy, integrity and authentication you could use both the ESP protocol for data privacy and the AH protocol for data integrity/authentication
- The latest version of the ESP protocol will provide data integrity and authentication which the earlier version did not. IPSec still supports both versions.

S/390 VPN Communications

WSC
Advanced
Technical Support

- Communicate Host-to-Host and Host-to-Client
  - S/390 Host-to-S/390 Host easily
  - S/390 Host to other non-S/390 host (as long as they support IPSec)
  - Host-to-Client requires a non-S/390 client that supports IPSec



S/390 VPN Setup

WSC Advanced Technical Support

- Requires setup performed on systems that have established a tunnel between them
  - define a tunnel between you and the system you wish to communicate with
  - define the Security Association desired via command options (policy, encrypthow, algorithm)
  - authentication algorithm options
    - KEYED_MD5, HMAC-MD5, HMAC-SHA
  - encryption algorithms
    - CDMF, DES, triple DES
- ICSF and Crypto hardware (if enabled) are used transparently by the VPN to perform some of the IPSec cryptographic functions

- A hash function creates a fixed length string from a block of data. If the function is one way, it is also called a message digest function. These (fast) functions analyze a message and produce a fixed length digest which is practically unique i.e. finding a message with an identical hash is very unlikely with very fast computers . There is no known feasible way of producing another message with the same digest. Such algorithms are normally used to create a signature for a message which can be used to verify it's integrity.

- MD5 produce 128-bit digests. The MD5 algorithm is the de-facto hashing standard for digests. Public domain versions are available for most platforms on the Internet and it is widely used in integrity checking systems.
  .
- An interesting variation of hashes are Message Authentication Codes (MAC), which are hash functions with a key. To create or verify the MAC, one must have the key. This is useful for verifying that hashes have not been tampered with during transmission. HMAC-MD5 computes the authentication checksum by combining a 128 bit key, the Hash-based Message Authentication Code (HMAC) authentication algorithm and the MD5 hash algorithm.

- HMAC is a secret key authentication algorithm. Data integrity and data origin authentication as provided by HMAC are dependent upon the scope of the distribution of the secret key. If only the source and destination know the HMAC key, this provides both data origin authentication and data integrity for packets sent between the two parties; if the HMAC is correct, this proves that it must have been added by the source.

- SHA-1 (Secure hashing algorithm) is a NIST sponsored hashing function that has been adopted by the U.S. government as a standard. It produces a 160-bit hash (i.e. larger than MDx) and is roughly 25% slower than MD5. SHA-1 is recommended

S/390 Telnet (TN3270e)

WSC Advanced Technical Support

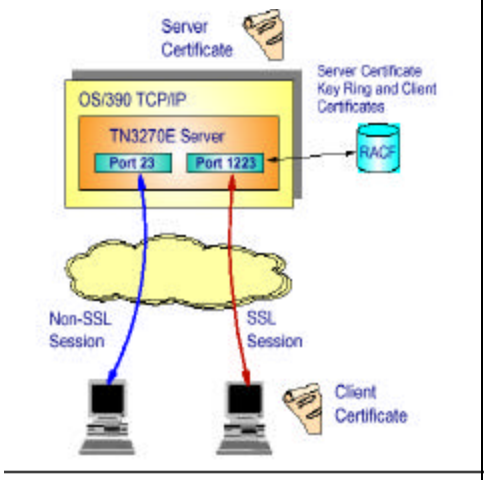
- Telnet Server (TN3270e) allows you to access remote systems
  - is part of the eNetwork Communication Server
- Network session is secured by using the Secure Socket Layer (SSL), a protocol developed by Netscape
  - SSL is a communication protocol for secure socket communications
  - provides privacy and reliability between two communicating applications
    - data encryption
    - message integrity
    - server and client authentication based on public key certificates
- TN3270e Support
  - server authentication with SSL in 2.6
  - client authentication with SSL in 2.8
  - RACF certificate support in 2.8

- Netscape has offered SSL as a proposed standard protocol to the World Wide Web Consortium and the Internet Engineering Task Force as a standard security approach for Web browsers and servers.
- SSL protocol is intended to be used on top of a reliable transport, such as Transmission Control Protocol (TCP/IP).
- Advantage of SSL with TN3270e is that the session now has encrypted data, (i.e. user ID and password no longer flow in clear text over the network link). Additionally, the client application can verify that the connection has been set up with a legitimate server, since the server's certificate must be authenticated by the client, or the session will be terminated.
- Host on Demand and PComm are the two applications that exploit server side authentication using SSL.

TN3270e Setup

WSC
Advanced Technical Support

- Ports
  - define SSL and non-SSL ports
    - SSL and non-SSL is not allowed over the same port at the same time
    - maximum of 255 ports supported for TN3270e
    - one port must be configured as SSL-only
    - data on a secure port is encrypted before it is sent to the client
- Define SSL environment parameters in TCP/IP profile
- Algorithms
  - MD5
  - SHA
  - RC2,
  - RC4
  - DES
  - triple DES
- Requires setup performed on all systems

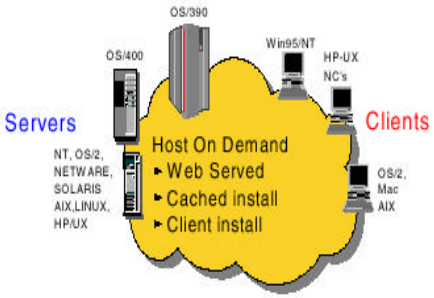


- The MD5 and SHA settings indicate the integrity checking method and are only valid for SSL V3 clients. Other algorithms are encryption methods and support V3 and V2 of SSL.

TN3270e Communication

WSC Advanced Technical Support

- Communicate Server-to-Client
  - Server-to-Client requires a Host-On-Demand (HOD) or PCOMM on client
  - Server-to-Client requires TN3270e server on the host
- HOD
  - client using a browser which has signed-applet support, can connect directly to a Telnet server
  - gives client access to TN3270e sessions
- ICSF and Crypto hardware can be used transparently by the S/390 TN3270e code to perform some of the SSL cryptographic functions



- HOD is a Java-based applet for browser access to 3270, 5250, VT100 and VT220 host. HOD server must run on the same system as a web server and the web server must be configured for the HOD server. When a client contacts the web site, the web server will pass the request to the HOD server and the HOD server, in turn, will pass the TN3270e emulator information back to the client.

- HOD provides SSL client authentication and RACF certificate support. The client certificate is supplied to the server for authentication and it can be used as input to RACF to verify that the certificate maps to a user ID. This provides some access control at the Telnet server which normally operates as pass-through and traditionally does no access control. This support ensures that the end user cannot get past the TN3270 server and attempt access to the SNA subsystem without a valid user ID on the Telnet system. This support was added in 2.8
  PCOMM also support the certificates.

- HOD allows installations to define TN3270 session information in a centralized site. This eliminates the need to update the Personal Communications definition in each individual workstation.

S/390 Web (HTTP) Server

WSC
Advanced
Technical Support

- HTTP Server allows you to access Web sites via the Internet and access files
  - part of Domino Go WebServer
- HTTP Server can secure the network session by using SSL
  - indicate SSL usage within the HTTP server
  - define the SSL environment parameters in the HTTP server config file
  - use IKEYMAN to create certificates and key database
  - data on a secure port is encrypted before it is sent to the client
- Requires setup performed on host system

HTTP Server Security

- Algorithms
  - Triple-DES
  - RC4
  - RC2
  - DES
  - SHA
  - MD5
- Communicate Server-to-Client
  - easily with most browsers supporting SSL
- ICSF and Crypto hardware can be used transparently by the S/390 HTTP Server code to perform some of the SSL cryptographic functions.

- Port 443 is the SSL default port for HTTP server.
- Encryption support is provided by RC2, RC4, DES or triple DES
- Hardware encryption is used if DES or triple DES is specified and the hardware crypto and ICSF are available.
- The earliest version of Domino Go Web Server to exploit hardware encryption for SSL is DGW 4.6.1

System Secure Socket Layer (SSL)

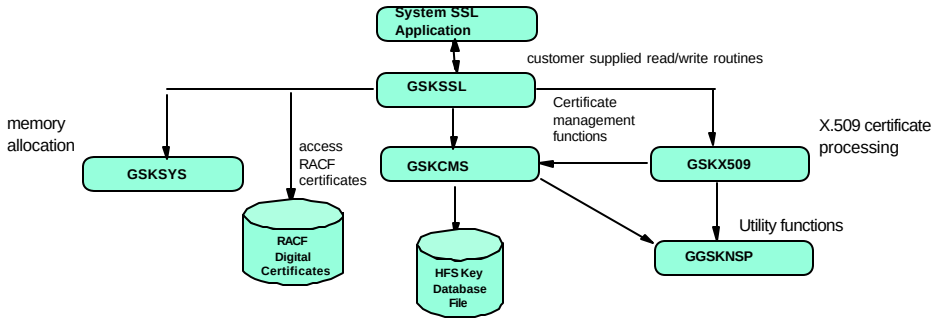
WSC Advanced Technical Support

- Included in OS/390 Base, as a toolkit consisting of DLL libraries
  - part of Cryptographic Services Base element of OS/390 (FMID HCPT290)
  - set of SSL C/C++ callable application programming interfaces
    - used with OS/390 Sockets APIs
    - provides the functions required for applications to establish secure sockets communication
  - uses Cryptographic Services Security Level 2 (FMID JCPT293)
    OR
  - Cryptographic Service Security Level 3 (FMID JCPT291)

- Cryptographic Services Base element is part of the base element of OS/390.
- The APIs are shipped in DLL libraries which reside in PDSs so they can be called from HFS-based as well as PDS-based program
- Triple DES is shipped these days to foreign countries with several IBM products however, it is not applicable to all IBM products. Therefore, the export regulations should still be reviewed and the import regulations of other countries.

System SSL Function

WSC Advanced Technical Support



- Provide functions required to establish secure socket communications when used with the OS/390 Sockets APIs
  - both client and server can use x.509 certificates when securing communications
  - provide programming interfaces for both server and client applications
  - provide utility function to support certificate storage and maintenance
    - gskkyman - create/manages HFS file that contains private keys & certificate info
    - RACDCERT - installs/maintains private keys & certificates in RACF

- Using System SSL, customers can;
  1. leverage secure socket communications in their applications
  2. create/manage their own digital certificates
  3. use RACF digital certificates for secure communications in their applications

- Advantages;
  1. saves the customer from providing their own SSL code
  2. allows for consolidation of digital certificates in RACF
  3. makes use of cryptographic hardware if enabled

System SSL

WSC
Advanced
Technical Support

- Communicate Server-to-Client
  - Server-to-Client easily with most browsers supporting SSL
- Requires setup performed on host system
- ICSF and Crypto hardware can be used transparently by System SSL to perform some of the SSL cryptographic functions

- OS/390 components using System SSL include; LDAP, OS/390 Firewall Configuration Server.

- BSAFE is used as a software crypto engine to perform cryptographic functions when ICSF and the crypto hardware are not available. However, applications cannot directly access the BSAFE interfaces. So a program must be written which makes used of System SSLs libraries.

Integrated Cryptographic Service Facility (ICSF)

WSC Advanced Technical Support

- Provides APIs which applications use for cryptographic services
  - part of base OS/390
  - used with hardware OS/390 cryptographic feature
    - secure, high-speed, and performs actual cryptographic functions
- S/390 application written with OS/390 ICSF
  - Communication can be secured by using ICSF APIs to request desired functions
- Requires setup performed on all systems
  - requires S/390 crypto hardware to be enabled
- 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

- Using ICSF APIs allows you to make the decisions and have complete control over your application. However, you must write the application and decide on every function you want this set of APIs to perform. Each function must be agreed upon by the receiving system and they must have hardware or software in place to handle the same functions.

- Using ICSF API's requires you to have the hardware Crypto Coprocessor Facility enabled. You must have a 9672 or 3000 cryptographic hardware that supports DES and/or Triple-DES and ICSF software (CSF address space) active. The 9672 G3, G4, and G5 and the 2003 processors support the DES algorithm in cryptographic hardware. The 9672 G4, G5, and G6 processors and the Multiprise 3000 support both DES and Triple-DES in hardware.

- To enable ICSF you must;
  1. enable and setup hardware cryptographic coprocessors
  2. define the ICSF environment and system data sets
  3. activate the hardware with hardware master keys (protects application keys)

- If you have large amounts of encrypted data then using hardware encryption can be faster and reduce the CPU utilization. If encrypting smaller amounts software may be faster or the same.

ICSF Support

WSC Advanced Technical Support

- Algorithms
  - DES or triple DES (for encryption/decryption)
  - Commercial Data Masking Facility (CDMF)
  - Rivest-Shamir-Adelman (RSA) (transport of data keys)
  - Digitial Signature Standard (DSS) (generation/verification of digital signatures)
- Callable services
  - PKA Encrypt and PKA Decrypt
  - enhances the security and performance of SSL applications

- The generation and verification of digital signatures uses both the RSA and DSS algorithms.
- PKA - Public Key Algorithm
- PKI - Public Key Infastructure, architecture for usage of PKA

ICSF Major Functions

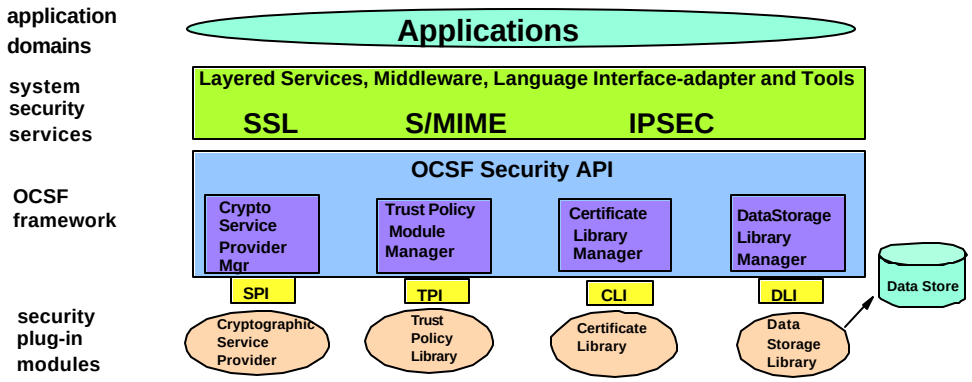
WSC
Advanced
Technical Support

- Ensures data privacy by encrypting/decrypting data
- Manages personal identification numbers
- Ensures integrity of data through use of;
  - message authentication codes (MACs)
  - modification detection codes (MDCs)
  - hash functions
  - digital signatures
- Ensures privacy of cryptographic keys by encrypting them under a master key

Open Cryptographic Service Facility (OCSF)

WSC Advanced Technical Support

- Included in OS/390 Base, as APIs
  - integrates and manages all the security services
    - supports the development of secure applications and system services
  - service provider interface (SPI)
    - supports service provider modules that implement building blocks for secure operations



- OCSF requires that you use BSAFE and have a license for it before putting any applications into production

- The Crypto Service Providers are optional features and one should be installed on the same system as OCSF;
  1. Security Level 1 (RC2, RC4, RC5)
  2. Security Level 2 (DES, RC2, RC4, RC5)
  3. Security Level 3 (TDES, RC2, RC4, RC5)

- The OCSF Framework has a rich application programming interface (API) to support the development of secure applications and system services. It also has a service provider interface (SPI) to support service provider modules that implement building blocks for secure operations.

- Provides interfaces for;
  1. cryptographic services
  2. trust policy services
  3. certificate services
  4. data store services

- Provides Service Providers for;
  1. cryptographic services
  2. certificates services

- OCSF is intended for use by UNIX System Services Applications.

OCSF Function

WSC Advanced Technical Support

- Provides a common security API used to access services of service provider modules
  - IBM's implementation of CDSA
- Mediates all interactions between applications and the service provider modules
- Implements and enforces the applicable cryptographic policy
- Allows integration of other security functions provided by independent service provider modules
  - common API for accessing these services
  - redirects application API calls to the selected module that will perform the request
- Focuses on security peer-to-peer, store-an-forward and archival applications

- CDSA - Common Data Security Architecture
- Applications request security services through the OCSF security API or through system security services implemented over the OCSF API.
- The service provider modules actually perform the requested security services.

Unknown User; No control over system

- 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 systems
- ICSF and Crypto hardware can be used transparently by the S/390 applications

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 is part of OS/390 VSAM, if using it in encryption mode, then ICSF must be running in COMPAT mode.

  If using IDCAMS the receiving end must have also have IDCAMS setup in their hardware or the receiving application must be written so it handles the IDCAMS header. IDCAMS will add two header records.

References

WSC Advanced Technical Support

- TechDoc Web Site
  - www-1.ibm.com/support/techdocs/atsmastr.nsf
    - find info, on OS/390 products from technical experts, which covers introductory and overview information to indepth usage and detailed setup including hints and tips
- SecureWay Communications Server for OS/390 V2R8 TCP/IP: Guide to Enhancements (SG24-5631)
  - TN3270e and HOD
- OS/390 Integrated Cryptographic Service Facility Overview (GC23-3972)
  - Introduction to ICSF
- OS/390 Firewall Technologies Guide and Reference (SC24-5835)
  - Setup of OS/390 VPN endpoint