Attachmate

CryptoConnect

Encryption System

For

Extra! Office

Security Policy

March 16, 2001

Introduction

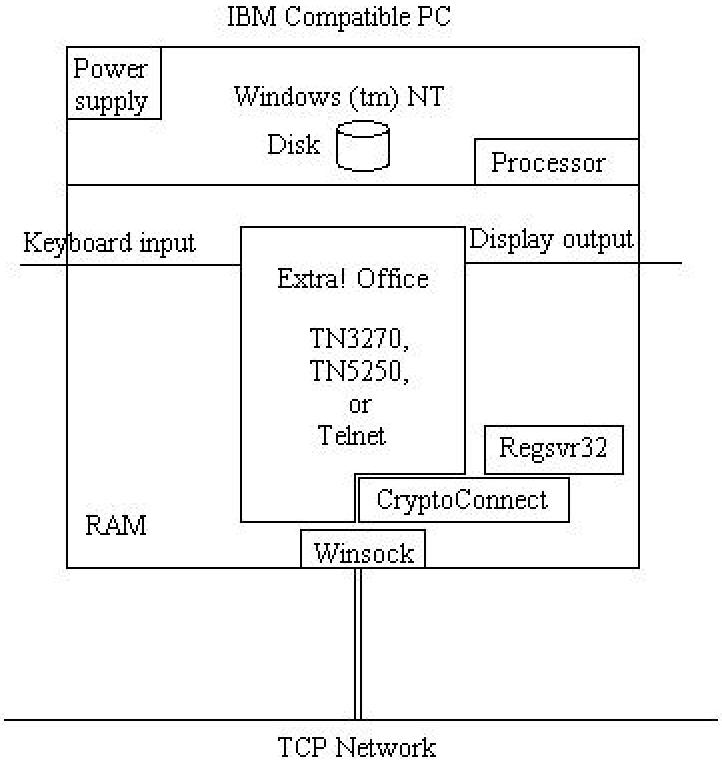
The CryptoConnect Encryption System (CryptoConnect ES) for Extra! Office (E!O) is a Dynamic Link Library, ATMCRYPT.DLL. It serves as security module which is called to perform encryption on all of the data sent to and received from the host. When used in conjunction with E!O's Telnet, TN3270 or TN5250 transports, it connects via TCP to an encryption server (the CryptoConnect Gateway) which handles the encryption tasks at the host end. It supports the FIPS approved SHA-1, DSA, DES, and Triple DES CBC algorithms. It also uses the non-FIPS approved RSA algorithm.

When CryptoConnect ES is first loaded, it runs a test for structural integrity. If the test fails, the module is not permitted to run.

Each time a session is opened, CryptoConnect ES runs a set of self-tests to verify the validity of the cryptographic suite. Known answer tests are run for RSA Public Key encryption and decryption, DES encryption and decryption, Triple DES encryption and decryption, SHA-1 hashing, and DSA signature verification. If these tests are not successful, the session open request is rejected.

Every random number that is generated is checked against the preceding value to protect against repeating values.

CryptoConnect ES uses a slightly modified version of the Secure Sockets Layer protocol to negotiate encryption parameters and keys with the encryption server. It uses the negotiated keys and parameters to encrypt all data being sent to the host, and to decrypt all data received from the host.



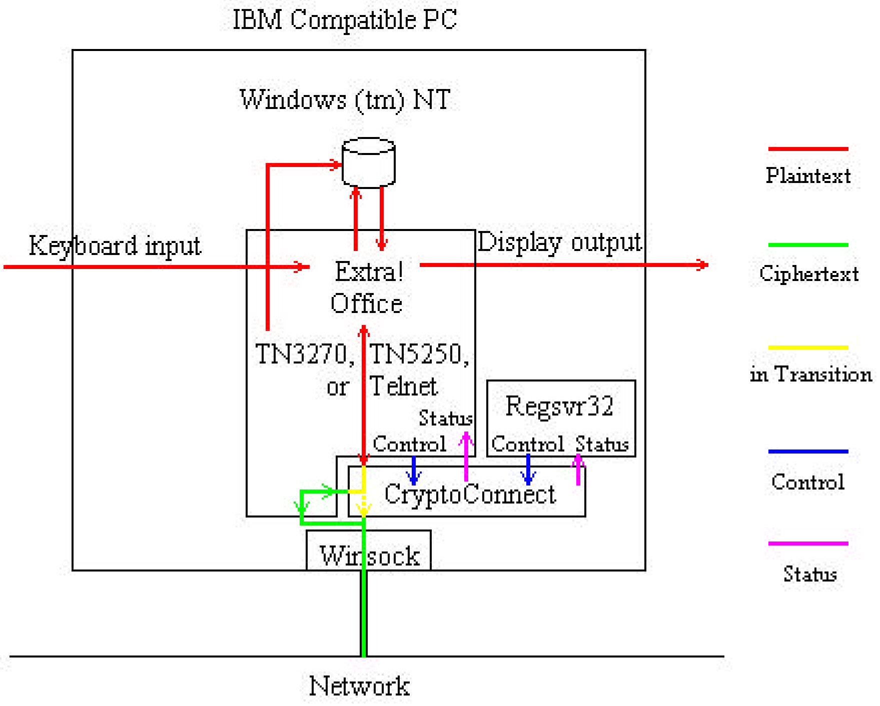
Overview

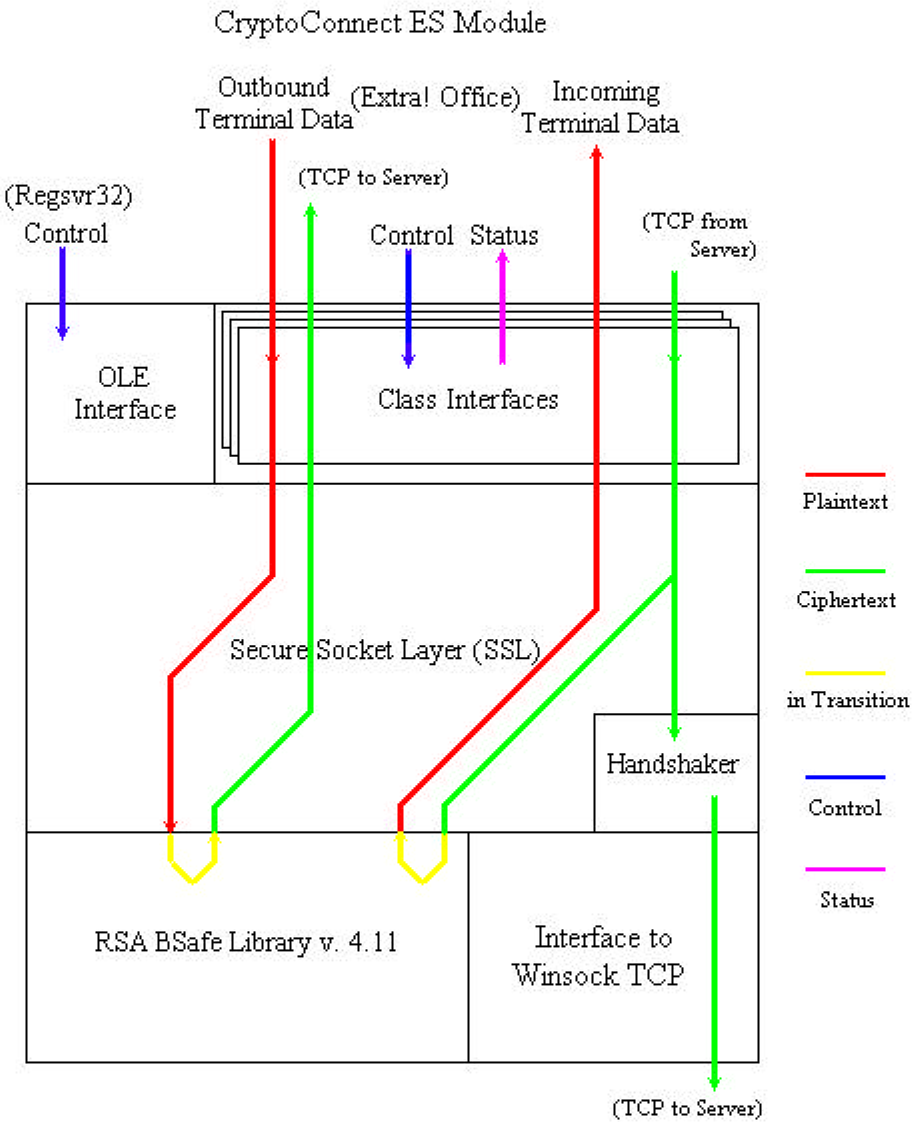
For FIPS 140-1 classification purposes, the Attachmate CryptoConnect Encryption System is intended to meet FIPS 140-1 level one requirements and is considered to be a "multi-chip standalone module." The "Secure Cryptographic Boundary" encompasses an IBM compatible PC with one or more network connections (supplied by the user), running the Attachmate Extra! Office software package with the CryptoConnect ES module, under either Windows^{tm} 95, Windows^{tm} 98, or Windows NT^{tm} (version 4.0, service pack 3 or higher) running in single user mode. All cryptographic functions are performed within the CryptoConnect ES module by the FIPS 140-1 certified version 4.11 of the RSA^{tm} BSAFE® Crypto-C toolkit.

A DSA public key, used for a DSA signature verification integrity test, is stored with the module. No other keys are permanently stored within the cryptographic boundary. The SSL (Secure Sockets Layer) protocol is used to agree on keys with the server at the other end of the communications link. The server encrypts random key generation material with its private key and sends it to CryptoConnect ES along with its certificate, containing the public key needed to decrypt it. CryptoConnect ES generates further random key material, encrypts it with the public key, and returns it to the server. Both sides then derive matching DES or Triple DES keys and vectors from the shared key material and use them to encrypt and decrypt their communications. These keys and vectors are never output, and the memory containing them is zeroized after use.

CryptoConnect ES uses a FIPS-approved random number generation algorithm to generate its key material. The methods employed in deriving the keys and vectors deviates from the SSL standard, which specifies the use of MD5 and SHA-1 to produce a “pre-master secret” and then to derive a “master secret,” and from that, the keys and vectors. Since the use of MD-5 in key generation is not FIPS approved, CryptoConnect ES uses only SHA-1 in generating the pre-master and master secrets and the keys and vectors.

Plaintext data, whether entered by the user or by other software, is presented by E!O to CryptoConnect ES through the input data API. The data is encrypted and the encrypted data is passed out via TCP through the network output data interface. Cyphertext data is received via TCP through the network input data interface. The data is decrypted and passed to E!O via the output data API.





The **OLE Interface** contains control logic for access to the module.

The **Class Interfaces** handle plaintext and ciphertext routing.

The **Secure Sockets Layer** handles memory management and plaintext and ciphertext management. It also includes the various error alarms.

The **Handshaker** negotiates encryption parameters.

The **RSA BSafe Library** handles all cryptographic algorithms and processing, key handling and zeroization.

The **Winsock TCP Interface** routes negotiation parameters to TCP during the negotiation of encryption parameters.

Roles and Services

The CryptoConnect Encryption System supports a User role and a Crypto-Officer role. No Maintenance role is supported. The CryptoConnect ES module does not support user identification or authentication for these roles.

A User is any entity that can operate an E!O Terminal Tool that supports the use of the CryptoConnect ES module to encrypt its communications. When a User is active, the CryptoConnect ES module is in the User State. A User has no access to any of the cryptographic parameters or keys.

A User can request the CryptoConnect ES module to (1) open an encryption session, (2) supply the current encryption status of the connection, (3) encrypt data to be sent to the host, (4) decrypt data received from the host and (5) close the encryption session. The following functions are available to a program operating in the User role:

QueryInterface – returns a pointer to a particular interface (OLE).
CreateInstance – creates the OLE object that gives access to the remaining functions (OLE).
AddRef – increments the OLE object's reference count (OLE).
NewSSLSocket – creates an encryption session and begins SSL parameter negotiation with the host.
Receive/Recv – receives encrypted data from the host, decrypts it, and returns clear text to the caller.
Encrypt/Send – accepts clear text and encrypts it for transmission to the host.
getEncryptionBuildLevel – returns identification of the encryption module installed.
getCurrentSecurityType – returns an indication of the SSL-negotiated cipher strength (key size).
writeDataEmpty – marks write data buffer empty; optionally releases the memory (after zeroizing).
CloseSocket – closes the encryption session.
Release – decrements the OLE object's reference count and deletes the object when it reaches 0 (OLE).

In addition, the following functions are available to the operating system while the module is operating in the User role:

DllMain – called by the operating system when the DLL is loaded or unloaded
DllCanUnloadNow – determine whether the DLL can be unloaded or is still in use
DllGetClassObject – get an "Object Factory" object for OLE
LockServer – called by OLE to lock the DLL in memory (or unlock it)

E!O can be configured to create several communication sessions which access the CryptoConnect ES module simultaneously. The state information for each is maintained independently and one session cannot access any of the data or state information of any other.

A Crypto Officer is any entity that can install or uninstall the CryptoConnect ES module as an OLE component. When a Crypto Officer is performing installation, the CryptoConnect ES module is in the Crypto Officer State. A Crypto Officer has no access to any of the cryptographic parameters or keys.

The following functions may be called by a program operating in the Crypto Officer role:

DllRegisterServer – registers the DLL as an OLE object.
DllUnregisterServer - unregisters the DLL as an OLE object.

In addition, the following functions are available to the operating system while the module is operating in the Crypto Officer role:

DllMain – called by the operating system when the DLL is loaded or unloaded
DllCanUnloadNow – determine whether the DLL can be unloaded or is still in use