Connect:Direct® Secure+ Option for i5/OS

Implementation Guide

Version 3.6

Sterling Commerce
An IBM Company

Connect:Direct Secure+ Option for i5/OS Implementation Guide
Version 3.6
First Edition

Sterling Commerce, Inc.

4600 Lakehurst Court Dublin, OH 43016-2000 * 614/793-7000

614/793-7000

Contents

Chapter 1 About Secure+ Option

Chapter 2 Installing and Setting Up Secure+ Option

Chapter 3 Maintaining Secure+ Option

Chapter 4 Accessing Secure+ Option Statistics

Chapter 1

About Secure+ Option

Secure+ Option provides enhanced security for Connect:Direct and is available as a separate component. It uses cryptography to secure data during transmission. You select the security protocol to use with the Secure+ Option product.

This chapter describes:

- Security concepts
- Secure Sockets Layer Protocol (SSL) and Transport Layer Security Protocol (TLS)
- Station-to-Station Protocol (STS)
- Secure+ Option tools
- Planning the Secure+ Option configuration
- Summary of processing using Secure+ Option

Security Concepts

Cryptography is the science of keeping messages private. A cryptographic system uses encryption keys between two trusted communication partners. These keys encrypt and decrypt information so that the information is known only to those who have the keys.

There are two kinds of cryptographic systems: symmetric-key and asymmetric-key. Symmetric-key (or secret-key) systems use the same secret key to encrypt and decrypt a message. Asymmetric-key (or public-key) systems use one key (public) to encrypt a message and a different key (private) to decrypt it. Symmetric-key systems are simpler and faster, but two parties must somehow exchange the key in a secure way because if the secret key is discovered by outside parties, security is compromised. Asymmetric-key systems, commonly known as public-key systems, avoid this problem because the public key may be freely exchanged, but the private key is never transmitted.

Cryptography provides information security as follows:

- Authentication verifies that the entity on the other end of a communications link is the intended recipient of a transmission.
- Non-repudiation provides undeniable proof of origin of transmitted data.
- Data integrity ensures that information is not altered during transmission.
- Data confidentiality ensures that data remains private during transmission.

Secure+ Option enables you to select one of three security protocols to use to secure data during electronic transmission: Transport Layer Security (TLS), Secure Sockets Layer protocol (SSL), or Station-to-Station protocol (STS).

Secure Sockets Layer Protocol (SSL) and Transport Layer Security Protocol (TLS)

The SSL and the TLS protocols use certificates to exchange a session key between the node that initiates the data transfer (the primary node, or Pnode) and the node that receives the data (the secondary node, or the Snode). A certificate is an electronic document that associates a public key with an individual or other entity. It enables you to verify the claim that a given public key belongs to a given entity. A certificate authority (CA) is the entity responsible for issuing and revoking these certificates. The CA validates an applicant's identity, creates a certificate, and then signs the certificate, thus vouching for an entity's identity. Use the IBM Digital Certificate Manager (DCM) to create and manage certificates and trusted root certificate files.

Security Levels

The SSL and TLS protocols provide three levels of security:

- The first level of security is activated when a trading partner connects to a Connect:Direct server. After the initial handshake, the Connect:Direct server sends its digital certificate to the trading partner. The trading partner checks that it has not expired and that it has been issued by a Certification Authority the trading partner trusts. The trading partner must have a trusted root file that identifies the CA.

  If the security fails on any one of these checks, the trading partner is notified that the site is not secure and the connection fails.

- The second level of security, called client authentication, requires that the trading partner send its own certificate. If enabled, the Connect:Direct server requests certificate information from the trading partner, after it returns its certificate information. If the client certificate is signed by a trusted source, the connection is established.

  In order to perform client authentication, the trading partner must have a key certificate file available at its site and the Connect:Direct server must have a trusted root file that validates the identity of the CA who issued the key certificate.

- The third level of security is defined in client authentication and requires that a certificate common name be defined in the receiver certificate. The Connect:Direct server searches the certificate file it receives from the trading partner and looks for a certificate common name. If the server cannot find the certificate common name, communication fails.

Data Security

To communicate using the SSL or TLS protocol, you must have both an X.509 certificate and a private key. The SSL and TLS protocols provide data security in the following areas:

- Strong authentication—Because the CA went through an established procedure to validate an applicant's identity, users who trust the CA can be sure the key is held by the owner. The CA prevents impersonation, and provides a framework of trust in associating an entity with its public and private keys.
- Proof of data origin and data integrity validation—The certificate provides proof of origin of electronic transmission and encryption validates data integrity. Encrypting the private key ensures that the data is not altered.
- Data confidentiality—Cipher suites encrypt data and ensure that the data remains confidential. Sensitive information is converted to an unreadable format (encryption) by the Pnode before being sent to the Snode. The Snode then converts the information back into a readable format (decryption).

Additional TLS Security

Both the SSL protocol and the TLS protocol manage secure communication in a similar way. However, TLS provides a more secure method for managing authentication and exchanging messages, using the following features:

- While SSL provides keyed message authentication, TLS uses the more secure Key-Hashing for Message Authentication Code (HMAC) to ensure that a record cannot be altered during transmission over an open network such as the Internet.
- TLS defines the Enhanced Pseudorandom Function (PRF), which uses two hash algorithms to generate key data with the HMAC. Two algorithms increase security by preventing the data from being changed if only one algorithm is compromised. The data remains secure as long as the second algorithm is not compromised.
- While SSL and TLS both provide a message to each node to authenticate that the exchanged messages were not altered, TLS uses PRF and HMAC values in the message to provide a more secure authentication method.
- To provide more consistency, the TLS protocol specifies the type of certificate which must be exchanged between nodes.
- TLS provides more specific alerts about problems with a session and documents when certain alerts are sent.

FIPS Certified Communication

If you want to increase security and provide FIPS-certified security for the SSL or TLS protocol, the cryptographic hardware option is available from IBM for iSeries servers. It encrypts and stores private keys associated with SSL transactions in tamper-responding hardware called the IBM 4758-023 Cryptographic Coprocessor PCI card.

The 4758 Cryptographic Coprocessor can be used with the OS/400 Digital Certificate Manager (DCM) to provide secure SSL private key storage, as well as increase iSeries server performance by off-loading from the iSeries server the cryptographic operations which are completed during SSL-session establishment.

To support load balancing and performance scaling, you can use up to eight 4758 Cryptographic Coprocessors with SSL on the iSeries server. When you install the 4758 Cryptographic Coprocessor, private keys are generated by the coprocessor as well as stored on the 4758 Cryptographic Coprocessor. The 4758 Cryptographic Coprocessor resists both physical and electronic hacking attempts. Refer to the IBM documentation for the steps necessary to secure private keys with cryptographic hardware.

Station-to-Station Protocol (STS)

The STS protocol is a three-pass variation of the basic Diffie-Hellman protocol. It enables you to establish a shared secret key between two nodes with mutual entity authentication. Nodes are authenticated using digital signatures that sign and verify messages.

Each message is signed by the Pnode with its current authentication private key (and possibly its previous authentication private key) and verified by the Snode using the corresponding public key of the Pnode. Each node uses two session keys to process control blocks: one for sending and the other for receiving. The encryption algorithms for control blocks and data copying functions are also determined. When strong authentication finishes successfully, control blocks are exchanged in an encrypted format for the entire session.

Providing Data Security

The STS protocol provides data security in the following areas:

- Strong authentication—The STS protocol uses a digital signature for strong authentication. After you enable this feature, control blocks are signed and verified. A digital signature uniquely authenticates the node signing an electronic document much like a human signature uniquely identifies the person signing his or her name to a physical document.
- Proof of data origin and data integrity validation—The digital signature verifies the sender of the message. The digital signature feature also provides data integrity validation. If the digital signature is verified, then an uncorrupted message was transmitted.
- Data confidentiality—The data encryption feature ensures confidentiality of the data sent in a Connect:Direct transfer. Sensitive information is converted to an unreadable format (encryption) by the Pnode before it is sent to the Snode. The Snode then converts the information back into a readable format (decryption). In order for the encryption/decryption process to work, each of these communicating nodes must have the public key value of the other.

Encryption Options

In previous releases of Secure+ Option, two versions of Secure+ Option were available for the STS protocol, based on government regulations regarding export laws. The difference in the versions is the encryption algorithms available.

The Limited Export version of Secure+ Option supports the following encryption algorithms:

- 56-bit DES using Cipher Block Chaining Mode (DESCBC56)
- 112-bit Triple DES in Cipher Block Chaining Mode (TDESCBC112)
- 128-bit IDEA in Cipher Block Chaining Mode (IDEACBC128)

The Export version of Secure+ Option supports only the 56-bit DES using Cipher Block Chaining Mode (DESCBC56) encryption algorithm. You must specify if a trading partner uses the Export version of Secure+ Option in the parameters file.

Secure+ Option Tools

Secure+ Option consists of three components: the Administration Tool (Admin Tool), the parameters file, and the access file. The following sections describe these components and their function within Secure+ Option.

Administration Tool

The Secure+ Option Administration Tool is an easy-to-use interface for configuring and maintaining the Secure+ Option environment. The Admin Tool is the only interface for creating and maintaining the Secure+ Option parameters file; operating system utilities and editing tools will not work. The Secure+ Option Admin Tool:

- Initializes the global data area
- Adds, deletes, displays, and copies node record information
- Maintains the Secure+ Option node records

The program also controls all updates between the Secure+ Option parameters file (SPNTMP) and Secure+ Option access file (SPACC). The Admin Tool and the session manager initialize the global data area and call two routines to control the maintenance of specific remote node records. The Admin Tool also calls a routine to update the access file.

Parameters File

The Secure+ Option parameters file contains information that determines the protocol and encryption method used during security-enabled Connect:Direct operations. To configure Secure+ Option, each site must have a parameters file that contains one local node record and at least one remote node record. The local node record defines the most commonly used security and protocol settings for the node at the site. Each remote node record defines the specific security and protocol used by a trading partner. You create a remote node record in the Secure+ Option parameters file for each Connect:Direct node that you communicate with.

For additional security, the parameters file is stored in an encrypted format. The information used for encrypting and decrypting the parameters file (and private keys) is stored in the Secure+ Option access file.

Access File

The Secure+ Option access file is generated automatically when you create the Secure+ Option parameters file. The access file provides greater security within a Secure+ Option environment, because it contains two encryption keys; one for encrypting and decrypting each record in the Secure+ Option parameters file and the other for encrypting and decrypting the private keys in the Secure+ Option parameters file. Your Secure+ Option administrator must secure the access file using available file access restriction tools. Availability of the access file to unauthorized personnel could compromise the security of your data exchange.

Planning the Secure+ Option Configuration

Before you configure the Connect:Direct environment for secure operations, first plan how you will use Connect:Direct Secure+ Option for i5/OS. Configure the Secure+ Option environment based on company needs or preferences. Below is a summary of the procedures necessary to configure Secure+ Option:

- Restrict access to Secure+ Option files—allowing access to the Secure+ Option parameters file (SPNTMP) and access file (SPACC) to unauthorized personnel may compromise the security of your data exchange. When planning the Secure+ Option implementation, ensure that the access attributes of these files remain as *PUBLIC *EXCLUDE.
- Populate the parameters file at your site by importing the Connect:Direct network map to create remote node records from the records defined in the network map.
- Enable the most commonly used protocol in the local node record. Enabling a protocol in the local node record configures remote nodes to default to the settings in the local node record.
- For remote nodes that will use the protocol defined in the local node record, the remote nodes are now enabled for secure communication. If you want to use settings that are different from the settings defined in the local node record, open the remote node record and change the appropriate settings.
- If a trading partner uses a protocol that is different from the protocol defined in the local node record, define the protocol in the remote node record. The remote node record must identify the same protocol as that used by the trading partner. Otherwise, Connect:Direct Secure+ Option will fail.
- If a trading partner does not use Secure+ Option, disable Secure+ Option in that remote node record.
- For nodes that use the SSL or TLS protocol for secure communication, you must configure a certificate.
- For nodes that use the STS protocol, you must exchange keys.

Key Management for the STS Protocol

When you configure a remote node record to use the STS protocol, you generate unique authentication and signature public keys. In addition, your trading partner generates authentication and signature public keys for that node. In order to communicate with the trading partner, all four keys must be defined in the parameters file for both your configuration and the trading partner's configuration. Therefore, you and your trading partner must exchange keys.

For the initial configuration, you manually exchange keys. You export keys and send them to the trading partner. Then you import the keys you receive from the trading partner into the parameters file. After the initial exchange, you can automate the exchange of key information.

If a remote node uses the STS protocol, you must decide how often to update keys and how to manage key files received from trading partners.

Key Exchange Method

After you exchange keys with a trading partner, both partners should enable the automatic key update feature for easier key management. When automatic key update is enabled, the updated key is sent to the trading partner node during the authentication process and the remote node record is updated with the new key values. Both you and your trading partner must enable automatic key update in order to use this feature.

Key Update Frequency

Decide how frequently to update authentication and signature keys. The more frequently you update key values, the more secure your environment is. When you turn on automated key updates, you can update keys daily, because the updated keys are sent to the trading partners automatically and securely during authentication.

Import Key File Management

Before you begin exchanging key files with a trading partner, you must consider how to manage key files. Secure+ Option names exported key files based on the name of the target node; therefore, new key files that you receive from a trading partner have the same name as the old key file. To avoid overwriting an old key file with a new one, you manage key files in one of the following ways:

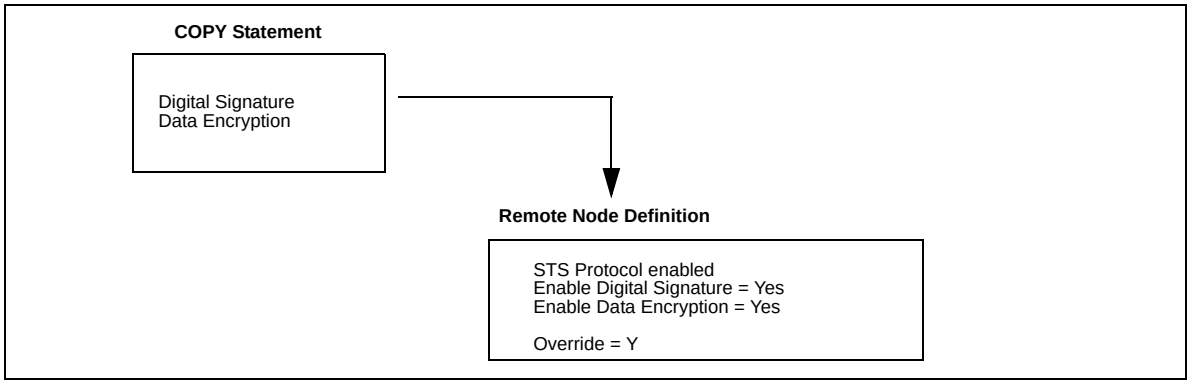
- Import the new key file immediately after receiving it from your trading partner and then delete the old key file.
- Rename the key file upon receipt or have your trading partner rename it before sending it.
- Create a file for each remote node and store each key file separately in the associated library for use if the configuration is lost or node records are accidently deleted.

It is not necessary to retain key files since the files are stored in the configuration file after you import them. However, saving the key files allows you to reconfigure the parameters file if a configuration is lost or if the node record is accidently deleted.

Overriding STS Functions from the CDSND or CDRCV Command

When you configure a node to use the STS protocol, you can use the CDSND or CDRCV command in Connect:Direct to override the settings in the parameters file, if override is enabled. It is not always possible to disable digital signatures and data encryption. If either node enables these options, the option are used.

The following illustration shows how the CDSND or CDRCV command overrides the security functions in a remote node:



Merging Secure+ Option Settings Using the STS Protocol

When two nodes use the STS protocol to exchange secure data, Secure+ Option settings are exchanged during authentication. These settings are then merged and the resulting value for each security function is used for the Connect:Direct session. The result is based upon the values defined on the primary node (Pnode) and the secondary node (Snode).

The following sections describe the results of these merged values based on the Pnode and Snode values.

Digital Signature

When Secure+ Option settings are merged, the most secure setting from either node is used for the digital signature feature. If either node enables the digital signature feature, digital signatures are used for the session. If both nodes disable digital signatures, digital signatures are not used. The following table shows the digital signature setting after the Pnode and Snode values are merged:

Pnode Value	Snode Value	Merged Results
Y	Y	Y
Y	N	Y
N	Y	Y
N	N	N

Algorithm for Encrypting Control Blocks

The algorithm that encrypts Connect:Direct control blocks used for strong authentication is the first algorithm ID in the Pnode list that is also in the Snode list. If the nodes do not share a common algorithm, authentication fails.

Autoupdate Public Keys

If both nodes enable the autoupdate function, the authentication and signature public key values are dynamically updated during authentication if the remote node supplies different values. Enabling autoupdate eliminates much of the work that has to be performed by the Secure+ Option administrator.

Data Encryption

The most secure setting from either node is used for data encryption. If the nodes do not share a common algorithm, the copy operation fails. The following table shows the setting after the Pnode and Snode values are merged.

Pnode Value	Snode Value	Merged Results
N	N	N
N	Y	The first algorithm ID in the Snode list that is in the Pnode list.
N	algorithm ID	The Snode algorithm ID if it is in the Pnode list.
Y	N | Y | algorithm ID	The first algorithm ID in the Pnode list that is in the Snode list.
algorithm ID	N | Y | algorithm ID	The Pnode algorithm ID if it is in the Snode list.

Summary of Processing Using Secure+ Option

After you configure Secure+ Option, you are ready to exchange data securely with other security-enabled Connect:Direct nodes. Your node must also be defined in the parameters file of your trading partner. Data is securely exchanged between two nodes using the protocol defined in the parameters file.

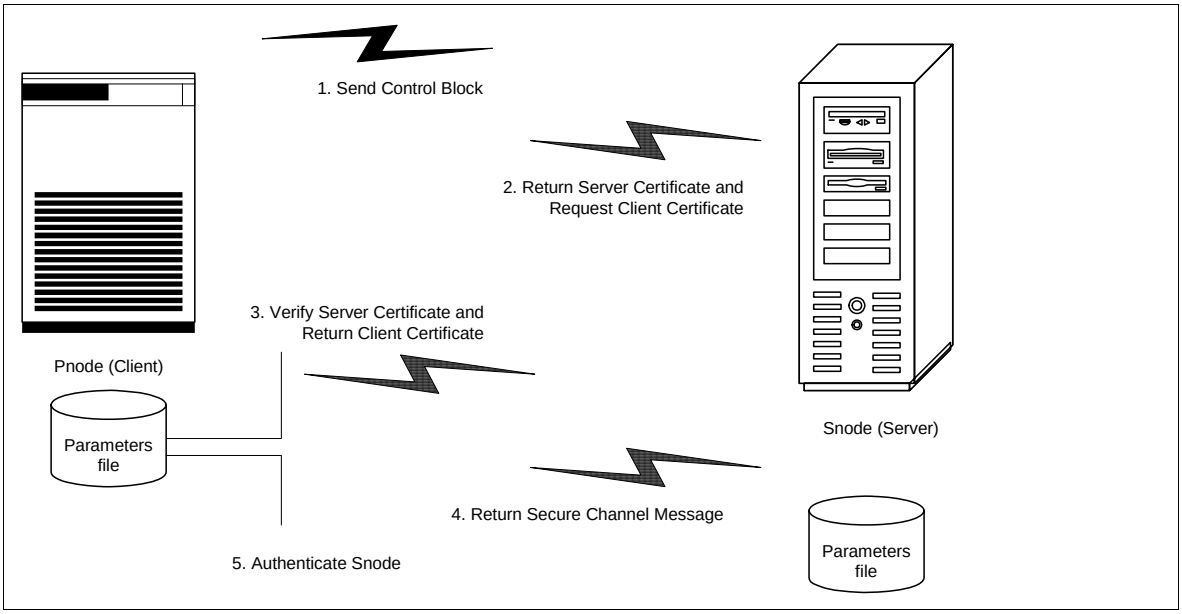
The following sections describe what happens during a data exchange between two Connect:Direct nodes using Secure+ Option with the TLS or SSL protocol and with the STS protocol.

TLS and SSL Secure+ Option Data Exchange

Data exchange consists of three steps: authentication, sending data, and receiving data. The TLS or SSL protocol data exchange process is described in the following sections. The primary node initiates the data transmission, and the secondary node receives the data.

Authentication

The following figure illustrates the authentication process using the TLS or SSL protocol:



The following steps occur during authentication:

1. The Pnode sends a control block to the Snode. The Snode confirms that it has a record defined in the Secure+ Option parameters file for the Pnode and determines the cipher suite to use for secure communication. If the Snode finds a record for the Pnode and a common cipher suite can be negotiated, the session continues.
2. The Snode sends its certificate back to the Pnode. Information for creating an encryption key is included. If client authentication is enabled, the Snode also requests a certificate from the Pnode.
3. The Pnode verifies that the certificate of the Snode is in its parameters file and generates a key session. If requested, it sends a client certificate to the Snode for verification.
4. The Snode confirms that a secure environment is established and returns a secure channel message.
5. The Pnode authenticates the Snode and establishes communications.

Sending Customer Data

After communication is authenticated, the Pnode begins transmitting data.

- Information for encrypting data is exchanged in the control blocks.
- If data compression is enabled, the Pnode compresses the data.
- The Pnode encrypts the data with a cipher suite recognized by both communications nodes.

Receiving Customer Data

The Snode receives the data.

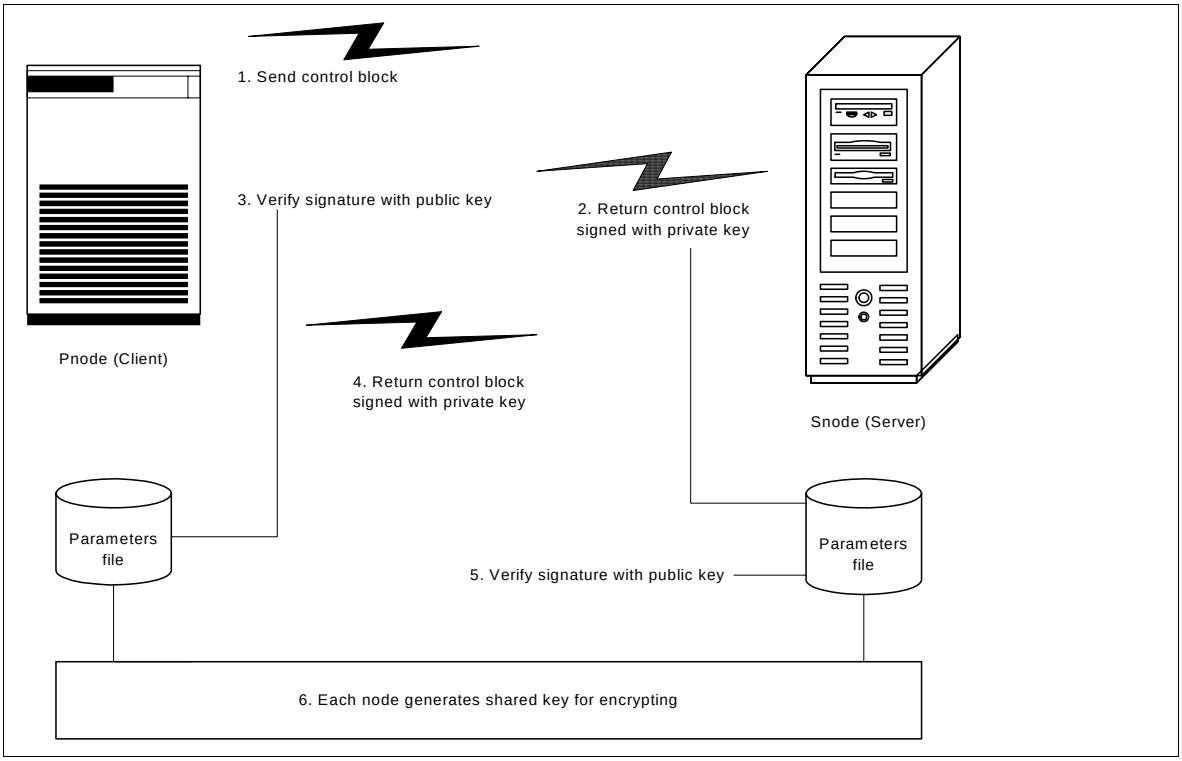
- The Snode decrypts the data using a cipher suite available for both the Pnode and the Snode.
- If the data is compressed, the receiving node decompresses it.

STS Secure+ Option Data Exchange

Data exchange consists of three steps: authentication, sending data, and receiving data. The STS protocol data exchange process is described in the following sections. The primary node initiates the data exchange and the secondary node receives the data.

Authentication

The following figure illustrates the authentication process using the STS protocol:



The following steps occur during authentication:

1. The Pnode sends a control block to the Snode. Information for creating an encryption key for the Pnode is included. The Snode confirms that it has a record defined in the Secure+ Option parameters file for the Pnode. If so, it retains the information for key encryption for processing later. If not, the session fails.
2. The Snode sends a control block signed with its private authentication key. Information for creating an encryption key is included.
3. The Pnode verifies the signature of the Snode using its public authentication key.
4. The Pnode returns a control block signed with its private authentication key.
5. The Snode verifies the signature using the public authentication key of the Pnode.
6. When authentication is successful, each node generates a shared session encryption key for encrypting control blocks.

Sending Customer Data

After communication is authenticated, the Pnode begins transmitting data.

- If data encryption is enabled, information for creating an encryption key is exchanged in the control blocks.

- If digital signature is enabled, the Snode applies the signature algorithm to the data using its private signature key to ensure that the data was sent by the Pnode and has not been altered.
- If data compression is enabled, the Pnode compresses the data.
- If data encryption is enabled, the Pnode encrypts the data with an encryption algorithm using a shared secret encryption key generated specifically for this transmission. The encryption algorithm is determined at authentication.

Receiving Customer Data

The Snode receives the data.

- If data is encrypted, the Snode decrypts the data using the encryption algorithm available for both the Pnode and the Snode.
- If the data is compressed, the Snode decompresses it.
- If digital signature is enabled, the Snode verifies the origin and integrity of the data by applying a verification algorithm using the public digital signature key of the Pnode.

Chapter 2

Installing and Setting Up Secure+ Option

This chapter provides instructions on preparing to use Secure+ Option if you purchased it separately from Connect:Direct for i5/OS. It also provides instructions for configuring nodes to run Secure+ Option.

Updating the License Management Key to Support Secure+ Option

In order to activate Secure+ Option, you need a license management key that activates Secure+ Option. If you purchased Secure+ Option at the same time that you purchased Connect:Direct for i5/OS, you installed Secure+ Option when you installed Connect:Direct for i5/OS and installed a key that activates Secure+ Option. If you purchase Secure+ Option separately from Connect:Direct for i5/OS, you receive a new license management key. Refer to the *Connect:Direct Secure+ Option for i5/OS Release Notes* for instructions on installing the key.

Configuration Considerations

You must configure Secure+ Option before you begin using it for secure communications. Perform the following tasks to configure the Secure+ Option environment:

- Create and save a parameters file that contains a single local node record and a remote node record for every trading partner that uses Secure+ Option.
- Configure the local node record. Configure the most commonly used protocol in the local node record because all remote node records default to the settings defined in the local node record.
- Configure those remote node records that use a protocol that is different from the one defined in the local node record.

Refer to the following table to help you decide how to configure the node records:

Scenario	Local Node Configuration	Remote Node Configuration	Other Setup Requirements
Most trading partners use the STS protocol.	Enable the STS protocol. Complete the procedure *Enabling the STS Protocol in the Local Node Record* on page 25.	❖ No configuration is required for trading partners who use the STS protocol. ❖ If a trading partner is using the SSL protocol, configure the SSL protocol in the remote node record. Complete procedure *Enabling the SSL or TLS Protocol in a Remote Node Record* on page 36. ❖ If a trading partner is using the TLS protocol, configure the TLS protocol in the remote node record. Complete the procedure *Enabling the SSL or TLS Protocol in a Remote Node Record* on page 36.	❖ Exchange authentication keys with trading partners. Complete the procedure *Preparing to Use the STS Protocol* on page 39. ❖ If some trading partners use the SSL or TLS protocol, obtain a certificate from a CA. Complete the procedure *Preparing to Use the SSL or TLS Protocol* on page 24.
Most trading partners use the SSL protocol.	Enable the SSL protocol. Complete the procedure *Configuring the Local Node Record for the SSL or TLS Protocol* on page 30.	❖ For trading partners who use the SSL protocol, no additional configuration is required. ❖ If a trading partner is using the STS protocol, configure the STS protocol in the remote node record. Complete the procedure *Enabling the STS Protocol in a Remote Node Record* on page 34. ❖ If a trading partner is using the TLS protocol, configure the TLS protocol in the remote node record. Complete the procedure *Enabling the SSL or TLS Protocol in a Remote Node Record* on page 36.	❖ Obtain a certificate from a CA. Complete the procedure *Preparing to Use the SSL or TLS Protocol* on page 24. ❖ If some trading partners use the STS protocol, exchange authentication keys with these trading partners. Complete the procedure *Preparing to Use the STS Protocol* on page 39.

Scenario	Local Node Configuration	Remote Node Configuration	Other Setup Requirements
Most trading partners use the TLS protocol.	Enable the TLS protocol. Complete the procedure *Configuring the Local Node Record for the SSL or TLS Protocol* on page 30.	❖ For trading partners who use the TLS protocol, no additional configuration is required. ❖ If a trading partner is using the SSL protocol, configure the SSL protocol in the remote node record. Complete the procedure *Enabling the SSL or TLS Protocol in a Remote Node Record* on page 36. ❖ If a trading partner is using the STS protocol, configure the STS protocol in the remote node record. Complete the procedure *Enabling the STS Protocol in a Remote Node Record* on page 34.	❖ Obtain a certificate from a CA. Complete the procedure *Preparing to Use the SSL or TLS Protocol* on page 24. ❖ If some trading partners use the STS protocol, exchange authentication keys with these trading partners. Complete the procedure *Preparing to Use the STS Protocol* on page 39.

Using the Secure+ Option Administration Tool

Use the Secure+ Option Administration Tool to initially set up and maintain Secure+ Option operations. To access the **SECURE+ ADMIN MAIN SCREEN**, at any command line, do one of the following:

- Type **SPADMIN** and press **Enter**.
- Type **GO CDADMIN** to open the **Connect:Direct Administration** menu and select option **6** from the menu.

The following screen displays all existing Secure+ Option parameter node entries:

```
05/13/04              SECURE+ ADMIN MAIN SCREENSSSSS

Position to node . . . _____________________________
Type option and press Enter.
  2=Change    4=Delete   5=Display    6=Add Alias Node

Opt   Node-Name      Typ   S+  STS   SSL  TLS   Ovr    Encryption   Sig  Lmt  Upd

__    *LCLNODE       L     Y   Y     N    N     Y      1DEACBC128    N          N
__    attilla        A
__    baseball       R     *   *     *    *     *      *
__    chili          R     Y   N     Y    N     Y
__    hotpepper      R     Y   N     N    Y     Y



F1=Help     F3=Exit      F5=Refresh       F6=Add NEW Entry   F7=Sync-Add
F8=Sync-Del    F12=Previous     F13=ReKey Secure+
```

About the Secure+ Option Admin Tool

When you start Secure+ Option Admin Tool, the screen displays all node records defined in a parameters file. The following table describes the fields that are displayed, including field descriptions and valid values for each field:

Field Name	Field Description	Valid Values
Opt	Type the appropriate number in this field next to the node to edit or display.	2=Change 4=Delete 5=Display 6=Add Alias Node
Node-Name	Displays the node record name.	Node name. An asterisk beside a node name indicates the local node record.
Typ	Displays the current record type.	L = local node record R = remote node record A = alias record
S+	Displays the status of Secure+ Option.	Y = Secure+ Option is enabled. N = Secure+ Option is disabled. * = default to local node.
STS	Identifies the status of STS security	Y = The STS protocol is enabled. N = The STS protocol is disabled. * = default to local node.
SSL	Identifies the status of SSL security	Y = The SSL protocol is enabled. N = The SSL protocol is disabled. * = default to local node.
TLS	Identifies the status of TLS security	Y = The TLS protocol is enabled. N = The TLS protocol is disabled. * = default to local node.
Ovr	Displays the status of override. If override is enabled in the local node record, the values defined in the remote node record override the values in the local node record. Activating override in a remote node record that use the STS protocol enables the values in the COPY statement to override the settings in the remote node record.	Y = Override is enabled. N = Override is disabled. * = default to local node.
Encryption	Indicates if encryption is enabled in the STS protocol and identifies the encryption algorithm.	The encryption algorithm configured. DESCBC56 TDESCBC112 IDEACBC128 NOALG indicates no algorithm has been configured.
Sig	Identifies if digital signature is enabled in the STS protocol.	Y= enabled. N = disabled. * = default to local node.
Lmt	Identifies if the trading partner is using the limited export version of the STS protocol.	Y = enabled. N = disabled. * = default to local node.
Upd	Indicates if the option to automatically update key values during communication is enabled.	Y = enabled. N = disabled. * = default to local node.

Navigating the Secure+ Option Admin Tool

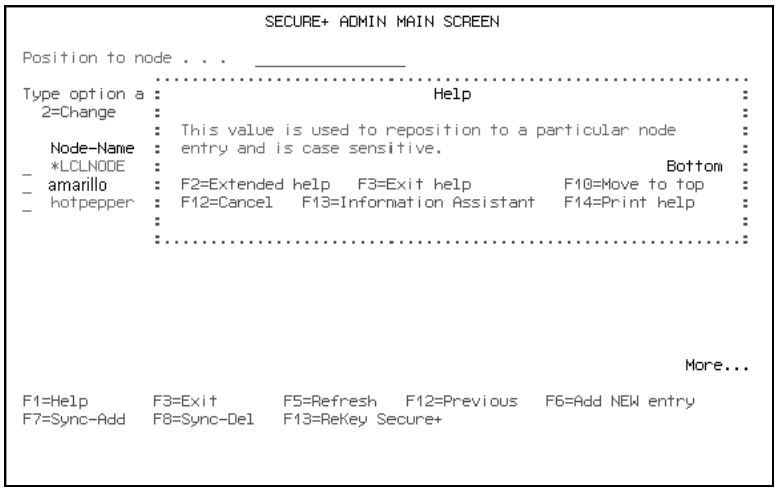
Use the following standard commands to navigate the **SECURE+ ADMIN MAIN SCREEN**.

Key	Function
F1	To access Help.
F3	To exit the menu.
F6	To display the **Add Secure+ Node** screen and add a new Secure+ Option node record.
F7	To synchronize the Secure+ Option parameters file and the Connect:Direct network map and automatically add network map records to the parameters file list if they do not currently exist.
F8	To synchronize the Secure+ Option parameters file and the Connect:Direct network map and automatically tags node records in the parameters file that do not exist in the network map for deletion. You can then selectively delete all records or individual records.
F12	To return to the previous menu.
F13	To decrypt and generate new keys for encrypting the parameters file, and re-encrypt the Secure+ Option parameters file. This is permitted only once per session.

Accessing Secure+ Option Admin Tool Help

You have the ability to access several types of Help information within the Secure+ Option Administration Tool. To access Help topics for the entry fields of the Secure+ Option Admin dialog boxes, fields, or windows:

1. Position the curser on the field and press **F1**. A dialog box displays a brief explanation of the field, as illustrated in the following sample screen:



2. Press **F14** to print the Help topic.

Creating the Secure+ Option Parameters File

To communicate with a node using Secure+ Option, that node must have a record in *both* your Connect:Direct network map and your Secure+ Option parameters file. For easier setup of your Secure+ Option environment, create your Secure+ Option parameters file from your existing network map.

To create Secure+ Option parameters files from an existing Connect:Direct network map:

❖ At the **SECURE+ADMIN MAIN SCREEN**, press **F7** (Sync-Add) to import all network map node records into the Secure+ Option parameters file.

A remote node record is automatically created in the Secure+ Option parameters file for each imported network map record. Each remote node record defaults to the local node record (*LCLNODE) settings.

Preparing to Use the SSL or TLS Protocol

If you plan to use the SSL or the TLS protocol to perform a secure connection, you must obtain a certificate and set up Connect:Direct to use certificates. Use the procedures in this section to obtain a certificate and set up Connect:Direct to use certificates.

Obtaining a Certificate

If you will use the SSL or TLS protocol for secure communications, you must obtain a certificate file from a trusted CA. Use one of three ways to obtain an X.509 version 3 certificate:

- Contract with a formal CA to obtain a server certificate. When you obtain the server certificate, import this certificate into the IBM System SSL toolkit key database using the IBM Digital Certificate Manager (DCM).
- Create a certificate using the IBM Digital Certificate Manager (DCM) and a certificate signing request. You then forward this certificate to a CA to be signed. When you receive the signed certificate, you import this certificate into the IBM System SSL key database. Refer to the IBM iSeries Information Center.
- You can become a CA by creating a self-signed certificate. If you create a self-signed certificate, you are responsible for distributing and administering the certificate roots file (*.arm file) that enables validating certificates. Refer to the IBM documentation in the iSeries Information Center.

Below is a summary of the steps necessary to obtain a certificate from a CA, using the DCM.

1. Use the IBM Digital Certificate Manager (DCM) to generate a Certificate Signing Request (CSR). Provide the following information:
   - Certificate type—identify if you are creating a server or client certificate.
   - Certificate store—identify the location of the certificate store.
   - Name of the CA who sign the certificate
   - Key size—the size of the key used to authenticate the certificate. The default of 1024 is sufficient for most organizations. Selecting a key size of 2048 increases security but requires more processing time.
   - Certificate label—if you are implementing client authentication, provide the certificate label. This information is provided in the common name field when you configure the node for client authentication.
   - Organization unit—information that identifies the division of the company requesting the certificate.
   - Organization name—information about the company requesting the certificate.

- Locality or city—the city where the company is located.
- State or province—the state or province where the company is located.
- Country or region—the country or region where the company is located.

When creating the certificate request in DSM, export the name of the local CA if it is not available in the DSM. Refer to the IBM documentation for instructions.

2. Send the CSR to the CA to obtain a certificate.
3. Once you obtain a certificate, store this file on the Integrated File System (IFS) on the i5/OS system. If the certificate is not in ASCII format, you must ftp the file in binary mode to the i5/OS IFS. Store the certificate in a protected area until it is loaded into the DCM.

Setting Up Connect:Direct to Use Certificates

You must select a Certificate Store, using DCM, in order to configure SSL/TLS. Below is a summary of the steps necessary to select a store. Refer to the IBM DCM documentation for detailed instructions.

- Select *SYSTEM Certificate Store.
- Enter the Certificate store path. The path of the Certificate store path should consist of the default path.
- If the default path listed is not displayed, type the Certificate store path as follows:

```
Certificate store path and filename: /QIBM/USERDATA/ICSS/CERT/SERVER/DEFAULT.KDB
```

- Your system may or may not allow you to change the default Certificate Store path. If the Certificate Store path is unavailable, enter the password to the *SYSTEM Certificate Store.

Configuring the Local Node Record

When you import the network map records into the Secure+ Option parameters file, Secure+ Option is disabled. You are now ready to configure the local node record. The local node record definitions are used as the default settings for all remote node records. Determine which protocol is used by most trading partners and configure this protocol in the local node record, which makes it easy to configure the remote node records.

Note: Only one protocol can be enabled for each node record. If more than one protocol is enabled, error messages are generated.

Enabling the STS Protocol in the Local Node Record

Perform the following procedure to enable the STS protocol in the local node record (*LCLNODE).

1. Type **SPADMIN** to open the **SECURE+ADMIN MAIN SCREEN**.
2. Type **2** by the local node record (*LCLNODE) and press **Enter** to open the node record.

```
05/01/04              Update Secure+ Node System: Comp 2

Node Name . . . : *LCLNODE          Node Type. . . . .  : R
Security Enabled:   Y               Encrypt Data . . .  : Y
Override Security: Y                Digital Signature . : Y
Auto Update . . :   Y               Enabled STS . . . . : Y
Auth. Time Out. :   120

Algorithm Names:  DESCBC56, TDESCB112, IDEACBC128

Alias Names . . .: *LCLNODE

AUT. Pub Key  TO Rmt: 0301.546A.7049.F149.OeFD.26AD.E83F.0C33
    AUT. Pub Key Expires:  Replace Prev. AUT. Key   Y
SIG.Pub Key  TO  Rmt: 0301.5D82.869.CF69.A9E1.88AC.95C7.8CE5
    SIG. Prev Key Expires: Replace Prev. SIG. Key   Y
AUT. Pub Key FROM Rm   0305.T9DD.CE75.2444.7AE9.5C74.E10A.B9E9.94B1.FC50.0B6C
SIG.Pub Key FROM Rm:   0206.D240.B84A.20E9.8D5D.84D2.5054.F4DD.5D52.D229.3DEB


F1=Help       F2=Enbl STS      F3=Exit      F4=Enbl SSL/TLS     F6=Reset AuKey
F7=Gen AuKey   F8-Reset SigKey   F9=Gen SigKey       F12=Return    F24=More Keys
```

Note: The **Update Secure+ Node** may display the SSL and TLS protocol fields and not the STS fields of information.

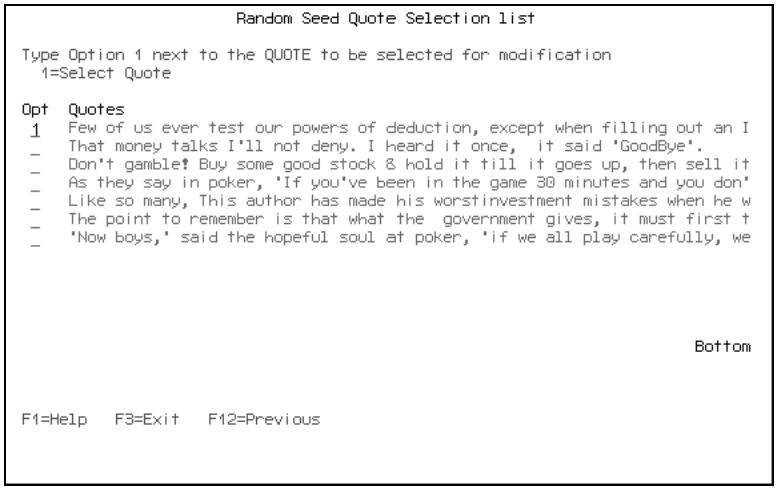
3. If necessary, press **F2** to display the STS protocol definition fields.
4. Set the following values:
   a. Security Enabled = N
   b. Override Security = Y
   c. Set Auth. Time Out the number of seconds to wait to receive control blocks before timing out.
   d. Type **Y** in the **Encrypt Data** field to enable data encryption during a file transfer.
   e. Type **Y** in the **Digital Signature** field to enable digital signatures.
   f. Type **Y** in the **Enabled STS** field to enable STS.

Note: You must set Security Enable to **N** until you have exchanged keys with your trading partner.

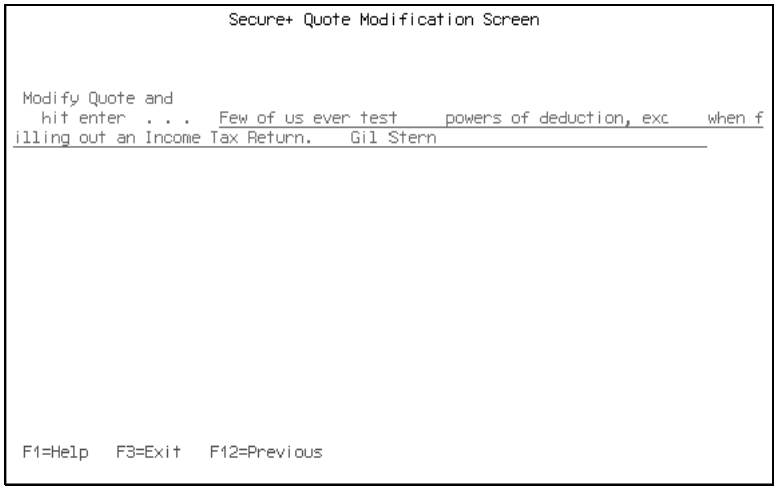
5. Identify the algorithm names to use for data encryption in the **Algorithm Names** field. If more than one algorithm is identified, identify them in the order in which to implement them.

6. To create the authentication keys in a local node record:

   a. Press **F7** to generate authentication keys. The **Random Seed Quote Selection List** screen is displayed.



   b. Choose a quote to change by typing the number **1** next to the quote and press **Enter**. The **Secure Quote Modification** screen is displayed.

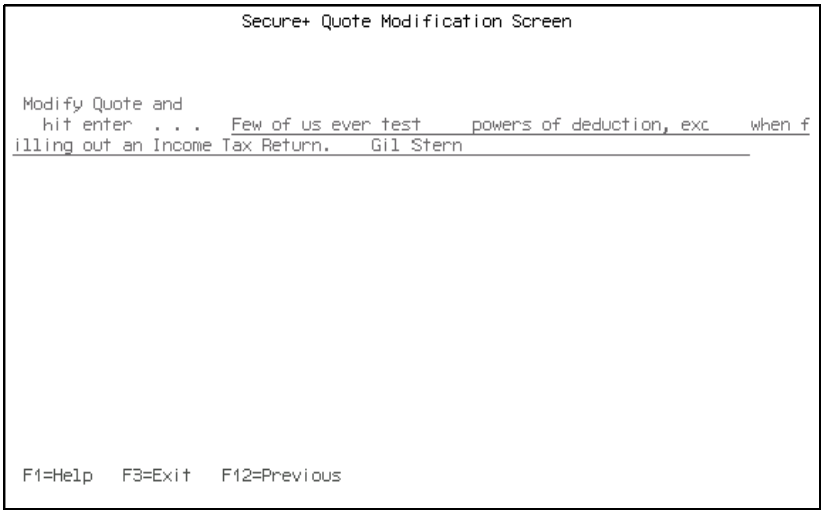


   c. Make at least *one change* to the quote and press **Enter** to save the changes. A message displays whether the key creation was successful. The **Update Secure+ Node** screen is displayed.

Note: If you have existing keys when you generate new authentication keys, the existing keys are saved as previous keys (if the **Replace Prev. AUT. Key** value is **Y**) with a default expiration date that is 30 days from the current date. Whether you choose manual or automated management of your keys, you must maintain a copy of the previous keys until your trading partner has the updated keys.

---

7. To create the signature keys in a local node record:
   a. From the **Update Secure+ Node** screen, press **F9**. The **Random Seed Quote Selection List** is displayed.
   b. Choose a quote to alter and type the number **1** next to it. The **Secure+ Quote Modification** screen is displayed.



   c. Make at least *one change* to the quote and press **Enter** to save the changes. A message displays whether the key creation was successful. The **Update Secure+ Node** screen is displayed.

> **Note:** If you have existing keys when you generate new signature keys, the existing keys are saved as previous keys (if the **Replace Prev.SIG. Key** value is **Y**) with a default expiration date that is 30 days from the current date. Whether you choose manual or automated management of your keys, you must maintain a copy of the previous keys until your trading partner has the updated keys.

8. If necessary, identify the date when the authentication keys expire in the **AUT Pub Key Expire Date** field.
9. Press **F3** to exit the **Update Secure+ Node** screen.

STS Protocol Field Definitions

The following table defines the STS protocol fields:

Field Name	Field Definition	Valid Values
Node Name	The name of the node being configured.	This field cannot be edited.
Security Enabled	Enables all Secure+ Option functionality including strong authentication. If Override=Yes in the local node record, the remote node record may determine if Secure+ Option is enabled.	Y | N

Field Name	Field Definition	Valid Values
Override Security	Specifies whether values defined in a remote node record can override the values defined in the local node record.	Y | N
Auto Update	Auto update of public keys during authentication. If Yes for both Pnode and Snode, after a one-time definition of parameters file entries, the public keys of the Pnode and Snode are exchanged during authentication. If a public key value is different from the one in the parameters file record, the record is updated with the new public key value for the remote node.	Y | N
Auth. Time Out	Maximum time, in seconds, that the system waits to receive Connect:Direct control blocks exchanged during the Secure+ Option authentication protocol.	A numeric value from 0-999. The default is 120 seconds. 0=No time-out. Connect:Direct waits indefinitely to receive the next message. Specify a time to prevent malicious entry from taking as much time as necessary to attack the authentication protocol.
Node Type	Identifies the type of node being modified.	L = local R = remote This field cannot be edited.
Encrypt Data	Enables data encryption during the Copy operation. If Yes, Algorithm must be populated and Security Enable field must also be set to Y. If the Snode enables encryption, the Pnode cannot override it, even if Override Security field value is Y.	Y | N
Digital Signature	Enables digital signatures to sign and verify messages when communicating with a trading partner. To enable digital signatures from the local node record, this value must be set to **Y**.	Y | N
Enable STS	Enables the STS protocol.	Y | N
Algorithm Names	Identifies the acceptable data encryption algorithms to use when Copy file encryption is requested. The algorithms are listed in order of preference, with the most-preferred algorithm listed first.	DESCBC56 | TDESCB112 | IDEACBC128 DESCBC56 (valid for both Export and Limited Export versions) TDESCBC112 (Limited Export version only) IDEACBC128 (Limited Export version only)
AUT. Pub Key TO Rmt	Public key used for strong authentication.	Created using Random Seed selection.
AUT.Prev. Key Expires	Expiration date for previous authentication public keys.	Format MM/DD/YY HH:MM:SS If time is not specified, 00:00:00 is used.
SIG. Pub Key TO Rmt	Public key used for digital signatures.	Created using Random Seed selection.

Field Name	Field Definition	Valid Values
Sig. Prev. Key Expires	Expiration date for previous digital signature public keys.	Format MM/DD/YY HH:MM:SS If time is not specified, 00:00:00 is used.
Replace Prev. AUT. Key	Specifies whether or not to save the current authentication key when creating a new one. An expiration date is required if this field is set to **Y**. This eliminates the need to update Secure+ Option parameters files across all nodes in the network simultaneously when changing public keys for the local node.	Y | N
Replace Prev.SIG Key	Specifies whether or not to save the current digital signature key when creating a new one. An expiration date is required if this field is set to **Y**. This eliminates the need to update Secure+ Option parameters files across all nodes in the network simultaneously when changing public keys for the local node.	Y | N

Configuring the Local Node Record for the SSL or TLS Protocol

Perform the following procedure to enable the SSL or STS protocol in the local node record (*LCLNODE).

1. Type **SPADMIN** to start Secure+ Option. The **SECURE+ADMIN MAIN SCREEN** is displayed.
2. Type **2** by the local node record (*LCLNODE) and press **Enter**. The **Update Secure+ Node** screen is displayed.
3. Type **F4** to display the SSL or TLS protocol fields as illustrated:

```
05/01/04              Update Secure+ NodeSystem: Comp 2

Node Name . . .: *LCLNODE
Security Enable:   Y
Node Type   . .:   L

Enable SSL    . . . : Y
Enable TLS    . . . : N
Enable Client Auth  : Y
Alias Names . . .: *

Application ID     :
Cipher Suites . .  :
Cert Common Name . :

F1=Help    F2=Enbl STS     F3=Exit   F4=Enbl SSL/TLS   F5=Refresh
F9=Cipher Suites        F12=Return
```

4. Type **N** in the **Security Enable** field to enable security.

Note: You must set Security Enable to **N** until you have exchanged keys with your trading partner.

5. Do one of the following:
   a. To enable the SSL protocol, type **Y** in the **Enable SSL** field.
   b. To enable the TLS protocol, type **Y** in the **Enable TLS** field.
6. To enable client authentication, type **Y** in the **Enable Client Auth** field.
7. Type the application ID you created in the DCM when you added the Secure+ Option application in the **Application ID** field.
8. To enable cipher suites:
   a. Press **F9** to open the **Secure+ TLS/SSL Cipher Suites Selection** screen.

```
05/01/04              Secure+ SSL/TLS Cipher Suites Selection    System:Comp
2

Update the order field below to enable and order cipher suites
Node: Company B

Default to Local Node . . . . N  (Y/N)

   Select suites order
Sel     All Available Cipher-Suites
        SSL_RSA_WITH_NULL_MD5
        SSL_RSA_WITH_NULL_SHA
        SSL_RSA_WITH_RC4_128_MD5
        SSL_RSA_WITH_RC4_128_SHA
 4      SSL_RSA_WITH_DES_CBC_SHA
 3      SSL_RSA_WITH_3DES_EDE_CBC_SHA
 2      SSL_RSA_WITH_AES_128_CBC_SHA
 1      SSL_RSA_WITH_AES_256_CBC_SHA

F1=Help   F3=Exit    F5=Make Selections
```

   b. Type numbers next to the cipher suites to enable in order of preference. Type 1 next to the most preferred cipher, 2 next to the 2nd most preferred cipher, and continue numbering until all preferred ciphers are enabled.
   c. To select ciphers suites selected and display the cipher suites selected in the order of preference, press **F5**.
   d. Press **Enter** to save the cipher suites selected and display the **SSL/TLS node** panel.

```
05/01/04              Suites Selected  System: Comp 2


Node: Company B


   Select suites order

Sel     All Available Cipher-Suites
 1      SSL_RSA_WITH_AES_256_CBC_SHA
 2      SSL_RSA_WITH_AES_128_CBC_SHA
 3      SSL_RSA_WITH_3DES_EDE_CBC_SHA
 4      SSL_RSA_WITH_DES_CBC_SHA

 F1=Help   F3=Exit    F5=Make Selections
```

9. Press **F3** to exit the **Suites Selected** panel and return to the **Update Secure**+ panel.

10. If client authentication is enabled, identify the name validation string to use in the **Cert Common Name** field.
11. Press **F3** to exit the **Update Secure+ Node** screen.

SSL and TLS Protocol Field Definitions

The following table describes the SSL and TLS protocol fields:

Field Name	Field Definition	Valid Values
Node Name	The name of the node being configured.	This field cannot be edited.
Security Enable	Enables all Secure+ Option functionality including strong authentication. If Override=Yes in the local node record, the remote node record may determine if Secure+ Option is enabled.	Y | N
Enable SSL	Set this field to Y to enable the SSL protocol.	Y | N
Enable TLS	Set this field to Y to enable the TLS protocol.	Y | N
Enable Client Auth	Set this field to Y to enable client authentication.	Y | N
Application ID	The application ID you created in the DCM when you added the Secure+ Option application in the **Application ID** field.	Valid application ID defined in the DCM.
Cipher Suites	Identify the cipher suites to enable and use for data encryption with the TLS or the SSL protocol.	SSL_RSA_WITH_NULL_MD5 SSL_RSA_WITH_NULL_SHA SSL_RSA_WITH_RC4_128_MD5 SSL_RSA_WITH_RC4_128_SHA SSL_RSA_WITH_DES_CBC_SHA SSL_RSA_WITH_3DES_EDE_CBC_SHA SSL_RSA_WITH_AES_128_CBC_SHA SSL_RSA_WITH_AES_256_CBC_SHA
Cert Common Name	The certificate common name validation string used for client authentication.	The validation string.

Configuring Remote Node Records

After you configure the local node, you are ready to update remote node records. When you imported the network map file, you created a remote node record in the parameters file for each remote node record defined in the network map. Depending upon how you configured the local node record, you may or may not need to update the remote node record.

The protocol you enabled in the local node record is automatically configured in each remote node record. If you plan to use the same protocol for all trading partners and the same security options, you are now ready to begin using Secure+ Option.

For each of the following configurations, you must configure a remote node record:

- Disable Secure+ Option in any remote node record that does not use Secure+ Option.
- Update all remote node records that use a different security protocol than the protocol defined in the local node record.
- Update any remote node record that uses the same protocol defined in the local node record but has different security options.

Disabling Secure+ Option in a Remote Node Record

Perform the following steps to disable Secure+ Option in a remote node record:

1. Type **SPADMIN** to start Secure+ Option. The **SECURE+ADMIN MAIN SCREEN** is displayed.
2. Type **2** next to the node name to configure and press **Enter**.

   The Update Secure+ Node screen is displayed.

```
05/01/04              Update Secure+ NodeSystem: Comp 2

Node Name . . . : Node A   Node Type. . . . .  : R
Security Enabled:   Y      Encrypt Data . . .  : Y
Override Security: Y       Digital Signature . : Y
Auto Update . . :   Y      Enabled STS . . . . : Y
Auth. Time Out. :   120

Algorithm Names:
Alias Names . . .: *LCLNODE

AUT. Pub Key  TO Rmt: 0301.546A.7049.F149.OeFD.26AD.E83F.0C33
    AUT. Pub Key Expires:  Replace Prev. AUT. Key   Y
SIG.Pub Key  TO  Rmt: 0301.5D82.869.CF69.A9E1.88AC.95C7.8CE5
    SIG. Prev Key Expires: Replace Prev. SIG. Key   Y
AUT. Pub Key FROM Rm   0305.T9DD.CE75.2444.7AE9.5C74.E10A.B9E9.94B1.FC50.0B6C
SIG.Pub Key FROM Rm:   0206.D240.B84A.20E9.8D5D.84D2.5054.F4DD.5D52.D229.3DEB


F1=Help       F2=Enbl STS      F3=Exit      F4=Enbl SSL/TLS    F6=Reset AuKey
F7=Gen AuKey   F8-Reset SigKey   F9=Gen SigKey       F12=Return    F24=More Keys
```

3. Type **N** in the **Security Enabled** field.
4. Press **F3** to exit the **Update Secure+ Node** screen and save the new settings.

Enabling the STS Protocol in a Remote Node Record

If you configured the STS protocol in the local node record, you do not have to configure the remote node record to use the STS protocol. However, if you configured the SSL or TLS protocol in the local node record and the trading partner will use the STS protocol for secure communications, you must enable the STS protocol in the remote node record. Perform the following steps to enable the STS protocol in a remote node record:

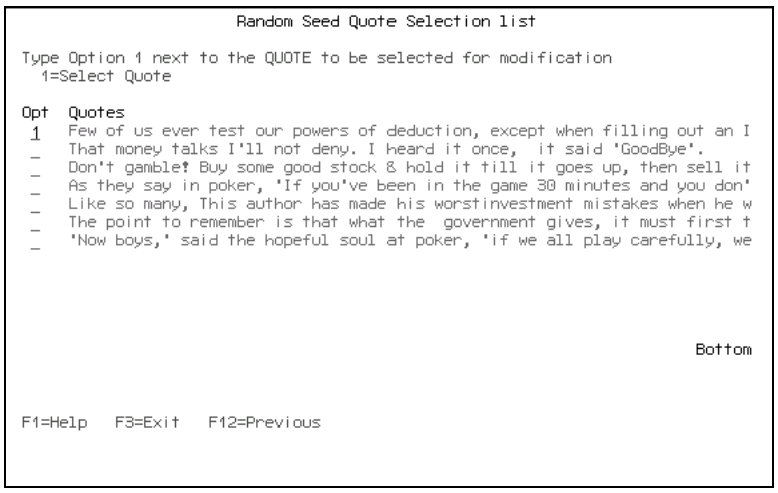
1. Type **SPADMIN**. The **SECURE+ADMIN MAIN SCREEN** is displayed.
2. Type **2** next to the node name to configure and press **Enter**.
3. Press **F2** to enable the STS protocol.
4. Set the following values:

Field Name	Field Definition	Valid Values
Security Enabled	Enables all Secure+ Option functionality including strong authentication. If Override=Yes in the local node record, the remote node record may determine if Secure+ Option is enabled.	Y | N
Override Security	Specifies whether values defined in a remote node record can override the values defined in the local node record.	Y | N
Auth. Time Out	Maximum time, in seconds, that the system waits to receive Connect:Direct control blocks exchanged during the Secure+ Option authentication protocol.	A numeric value from 0-999. The default is 120 seconds. 0=No time-out. Connect:Direct waits indefinitely to receive the next message. Specify a time to prevent malicious entry from taking as much time as necessary to attack the authentication protocol.

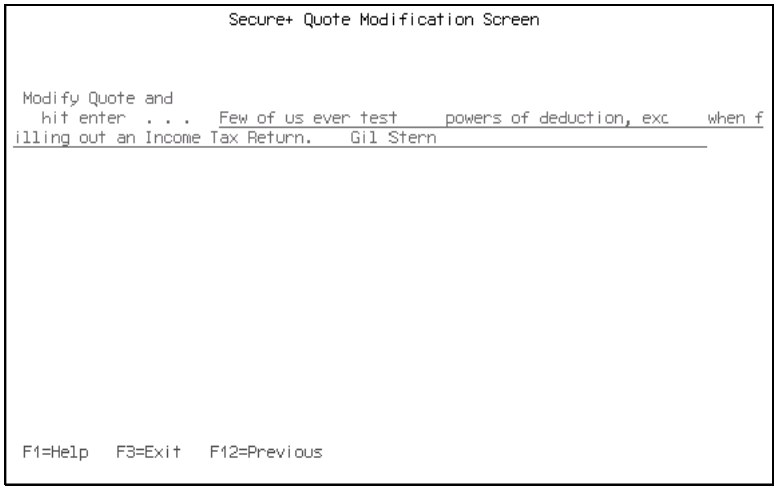
5. Type **Y** in the **Encrypt Data** field to enable data encryption during a file transfer.
6. Type **Y** in the **Digital Signature** field to enable digital signatures.
7. Type **Y** in the **Enabled STS** field to enable STS.
8. Identify the algorithm names to use for data encryption in the **Algorithm Names** field. If more than one algorithm is identified, identify them in the order in which to implement them.

9. To create the authentication keys in a remote node record:

   a. Press **F7**. The **Random Seed Quote Selection List** screen is displayed.



   b. Choose a quote to change by typing the number **1** next to the quote and press **Enter**. The **Secure Quote Modification** screen is displayed.

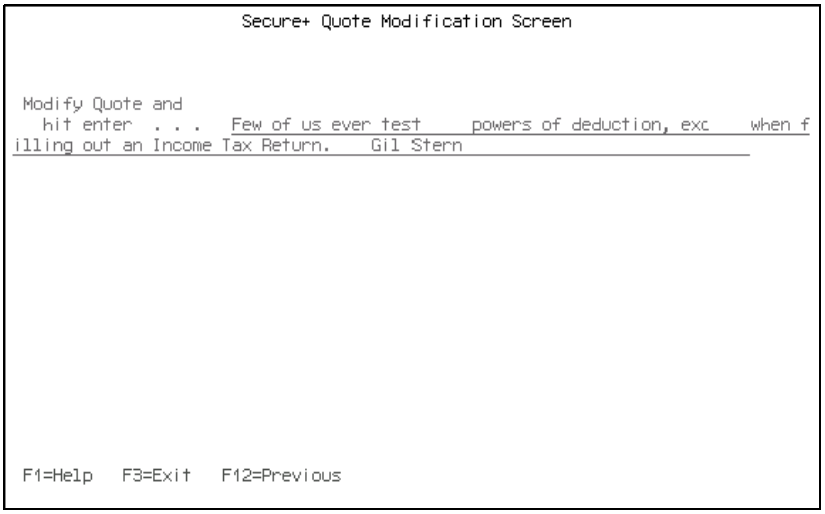


   c. Make at least one change to the quote and press **Enter** to save the changes. A message displays whether the key creation was successful. The **Update Secure+ Node** screen is displayed.

Note: If you have existing keys when you generate new authentication keys, the existing keys are saved as previous keys (if the **Replace Prev. AUT. Key** value is **Y**) with a default expiration date that is 30 days from the current date. Whether you choose manual or automated management of your keys, you must maintain a copy of the previous keys until your trading partner has the updated keys.

---

10. To create the signature keys in a remote node record:
    a. Press **F9**. The **Random Seed Quote Selection List** is displayed.
    b. Choose a quote to alter and type **1** next to it. The **Secure+ Quote Modification** screen is displayed.



    c. Make at least one change to the quote and press **Enter** to save the changes. A message displays whether the key creation was successful. The **Update Secure+ Node** screen is displayed.

> **Note:** If you have existing keys when you generate new signature keys, the existing keys are saved as previous keys (if the **Replace Prev.SIG. Key** value is **Y**) with a default expiration date that is 30 days from the current date. Whether you choose manual or automated management of your keys, you must maintain a copy of the previous keys until your trading partner has the updated keys.

11. If necessary, identify the date when the authentication keys expire in the **AUT Pub Key Expire Date** field.
12. Press **F3** to exit the **Update Secure+ Node** screen.

    Refer to the table on page 28 for the name, definition, and valid values for the STS protocol fields.

Enabling the SSL or TLS Protocol in a Remote Node Record

After you configure the local node record, all remote node records are automatically configured to use the settings defined in the local node record. If you configured the SSL protocol in the local node record, you do not have to configure the remote node record to use the SSL protocol. If you configured the TLS protocol in the

local node record, you do not have to configure the remote node record to use the TLS protocol. However, for all remote nodes that use a protocol that is different from the protocol defined in the local node record, you must enable the corresponding protocol in the remote node record.

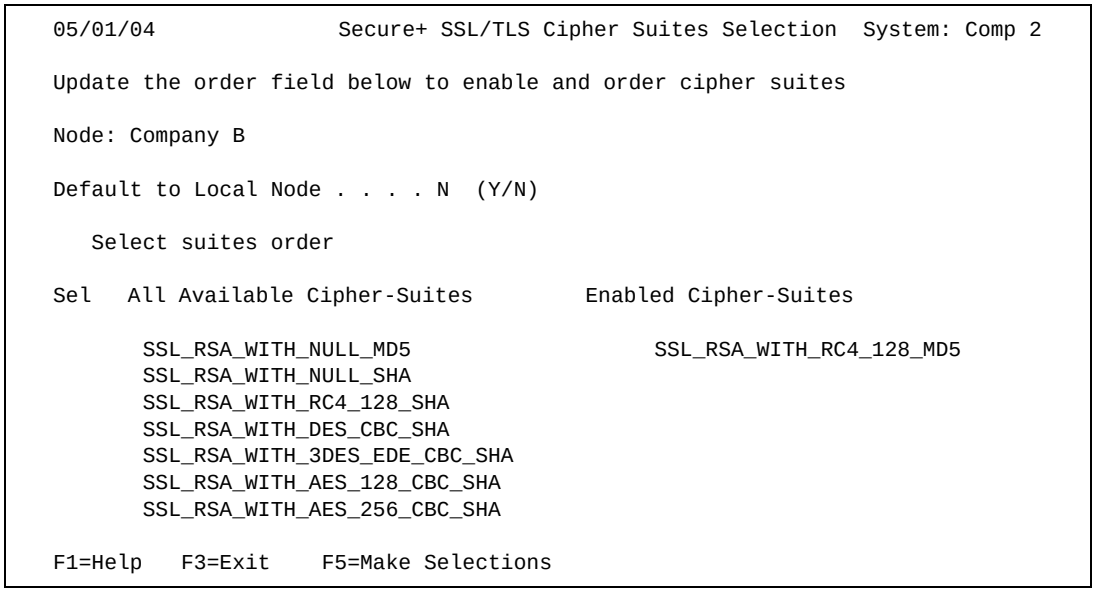
Perform the following steps to enable the SSL or TLS protocol in a remote node record:

1. Type **SPADMIN** to start Secure+ Option. The **SECURE+ADMIN MAIN SCREEN** is displayed.
2. Type **2** by the node record to configure and press **Enter**.

```
05/01/04              Update Secure+ NodeSystem: Comp 2

Node Name . . .: Node A
Security Enable:   Y
Node Type   . .:   L

Enable SSL    . . . : Y
Enable TLS    . . . : N
Enable Client Auth  : Y
Alias Names . . .: *

Application ID     :
Cipher Suites . .  :
Cert Common Name . :

F1=Help    F2=Enbl STS     F3=Exit   F4=Enbl SSL/TLS   F5=Refresh
F9=Cipher Suites        F12=Return
```

3. Type **Y** in the **Security Enable** field to enable security.
4. Do one of the following:
   a. To enable the SSL protocol, type **Y** in the **Enable SSL** field.
   b. To enable the TLS protocol, type **Y** in the **Enable TLS** field.
5. To enable client authentication, type **Y** in the **Enable Client Auth** field.
6. Type the application ID you created in the DCM when you added the Secure+ Option application in the **Application ID** field.

7. To enable cipher suites:
   a. Press **F4** to open the **Secure+ TLS/SSL Cipher Suites Selection** screen.



   b. Type numbers next to the cipher suites to enable in order of preference. Type 1 next to the most preferred cipher, 2 next to the 2nd most preferred cipher, and continue numbering until all preferred ciphers are enabled.
   c. To select ciphers suites selected and display the cipher suites selected in the order of preference, press **F5**.
   d. Press **Enter** to save the cipher suites selected and display the **SSL/TLS node** panel.
8. If client authentication is enabled, identify the name validation string to use in the **Cert Common Name** field.
9. Press **F3** to exit the **Update Secure+ Node** screen. Refer to the table on page 32 for a description of the SSL and TLS protocol fields.

Change Security Enable Settings in the Remote Node Record

Before you can test your Secure+ Option installation with a trading partner, you must both enable Secure+ Option.

Change the **Security Enable** value for the remote node record to **Y** and ensure that the **Override Security** value is **Y** in the local node record (*LCLNODE).

Preparing to Use the STS Protocol

For nodes that use the STS protocol, you are responsible for managing the keys that you create. The first time you use the STS protocol, you manually exchange keys with the trading partner. After you exchange keys for the first communications session, you can then turn on the automatic key management function. The automatic update function enables the public keys to be updated during a communications session, which simplifies key management for ongoing communications.

When you configure a remote node record to use the STS protocol, you must exchange keys with the trading partner before you can use Secure+ Option with that node. To maintain the keys for the STS protocol, perform the following tasks:

- Export keys and send this exported file to your trading partner
- Import keys received from your trading partner

Exporting Keys to Send to Trading Partners

After you create signature keys for a node record, you must send this information to the trading partner. Perform the following steps to export the authentication and signature public key values from a remote node record. You must be in edit mode for the remote node record that you are exporting keys for in order to perform this action.

1. From the **SECURE+ ADMIN MAIN SCREEN**, type **2** next to the remote node record for the trading partner and press **Enter**.
2. Press **F10** to start the export procedure. The following screen is displayed.

```
                    SPADMIN Distribution Public Keys

Node Name   hotpepper

EXPORTING Distribution File Name:
  Library Name . . . . . .   CDEXPORT
  File Name  . . . . . .     wanda
  Member Name (REQUIRED)     wanda




F1=Help   F3=Exit   F12=Previous
```

3. Type the library, file name and member names for an export file. Do not use the Connect:Direct Library name.
4. Press **Enter** to accept and process the procedure. The public keys are transferred to the specified library in the Export file with the member name.

> **Note:** Do not modify this member in any way or the keys will not work.

5. Send the export file you created to the trading partner and ask them to import the file into their parameters file.

Using CDSND to Send Exported Keys to a Trading Partners

Below is a sample script using the CDSND command to send export keys files to a Windows system:

```
CDSND  SNODE(Windows.NODE) SNODENVIRN(WINDOWS_NT)
        FDSN('CDEXPORT/FILENAME(MEMBERNAME)')
        TDSN('FILE-NAME') FMSYSOPTS('TYPE(MBR)')
        TOSYSOPTS('DATATYPE(BINARY) XLATE(NO) STRIP.BLANKS(NO)')
        PNAME(CDSNDKEY)
        SNODEID(USERID PASSWORD) TDISP(NEW)
```

Below is a sample script using the CDSND command to send the Export file to an OS/390 system.

```
 CDSND      SNODE(OS390.NODE) SNODENVIRN(OS390) +
                              FDSN('CDEXPORT/FILENAME(MEMBERNAME)') +
                              TDSN(USERNAME.DSN.PATH) +
                              FMSYSOPTS('TYPE(MBR)') PNAME(CDSNDKEY) +
                              SNODEID(USERID) TDISP(RPL CATLG DELETE) +
                              TDCB(*N 27208 *N *N 000 00000 27200 *N VB)
```

Below is a sample script using the CDSND command to send the Export file to a UNIX system.

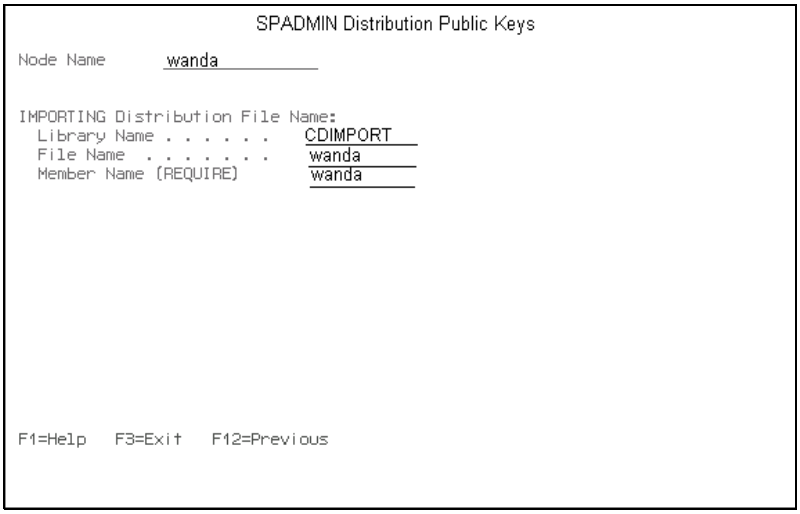
```
CDSND SNODE(UNIX.NODE) SNODENVIRN(UNIX) +
FDSN('CDEXPORT/FILENAME(MEMBERNAME)') +
TDSN('FILE-NAME') FMSYSOPTS('TYPE(MBR)') +
TOSYSOPTS(':DATATYPE=BINARY:XLATE=NO:STRIP.+
BLANKS=NO:') PNAME(CDSNDKEY) +
SNODEID(USERID PASSWORD) TDISP(NEW)
```

Importing Keys Received from a Remote Node

Before you can communicate with a trading partner, you must obtain keys from the trading partner and import this information into your parameters file. Perform the following steps to import the authentication and signature public key values received from the trading partner. Prior to importing public keys, you must copy the remote node's public key onto your system. The remote node's public key file should be received as a binary/data file.

1. Copy the trading partner's key file to your i5/OS computer.
2. From the **SECURE+ ADMIN MAIN SCREEN**, type **2** next to the remote node record for the trading partner and press **Enter**.

3. At the **Update Secure+ Node** screen, press **F11** to start the import procedure and display the following screen.



4. Type the Library, File, and Member names for the import file. Do not use the Connect:Direct Library name.
5. Press **Enter** to import the key information.

   The imported data is translated to EBCDIC and then parsed to extract and check node names, **AUT.Pub Key TO Rmt**, and **SIG.Pub Key TO Rmt** values. Node names are case sensitive. If the node name in the import file does not match the node name in the remote node record, a message is issued, “IMPORT name does not match.”

Note: The imported keys are validated for correct format and key size based on the remote node’s export limit(s).

Using CDRCV to Obtain the Import Key Files

Below is a sample script using CDRCV to pull the IMPORT keys from the Windows.receive key files to import keys from a Windows system:

```
CDRCV  SNODE(Windows.NODE) SNODENVIRN(WINDOWS_NT)
        FDSN('FILE-NAME')
        TDSN('CDIMPORT/FILENAME(MEMBERNAME)')
        TOSYSOPTS('TYPE(MBR) RCDLEN(80)
        FILETYPE(*DATA)')
        FMSYSOPTS('DATATYPE(BINARY) XLATE(NO) STRIP.BLANKS(NO)')
        PNAME(CDRCVKEY) SNODEID(USERID PASSWORD)
        TDISP(NEW)
```

Below is a sample script using the CDRCV command to obtain the Import key file from an OS/390 system.

```
     CDRCV      SNODE(OS390.NODE) SNODENVIRN(OS390) +
                          FDSN(USERNAME.DSN.PATH) +
                          TDSN('CDIMPORT/FILENAME(MEMBERNAME)') +
                          TOSYSOPTS('TYPE(MBR) RCDLEN(80) +
                          FILETYPE(*DATA)') PNAME(CDRCVKEY) +
                          SNODEID(USERID) TDISP(RPL)
```

Below is a sample script using the CDRCV command to obtain the Import key file from a UNIX system.

```
CDRCV SNODE(UNIX.NODE) SNODENVIRN(UNIX) +
FDSN('FILE-NAME') +
TDSN('CDIMPORT/FILENAME(MEMBERNAME)') +
TOSYSOPTS('TYPE(MBR) RCDLEN(80) +
FILETYPE(*DATA)') +
FMSYSOPTS(':DATATYPE=BINARY:XLATE=NO:STRIP.+
BLANKS=NO:') PNAME(CDRCVKEY) +
SNODEID(USERID PASSWORD) TDISP(NEW)
```

Overriding Secure+ Functions Using Connect:Direct Copy Commands

After you set up the Secure+ Option environment, security is implemented each time that you use Connect:Direct with any nodes that are also configured and enabled for Secure+ Option. If you are using the STS protocol for secure communications, you can override some Secure+ Option functions from copy commands (**CDSND**, **CDRCV**, and **CDSNDSPL**). This section instructs you on how to override the Secure+ Option features using the COPY statement.

Setting Secure+ Option Function Values from Copy Commands

Using the new SECOPTIONS (* *) command parameter, you set data encryption and digital signatures features from the Connect:Direct for i5/OS copy commands (CDSND, CDRCV, and CDSNDSPL). You can always enable these features from the copy command, but you cannot necessarily disable them.

The SECOPTIONS parameter value specified in copy commands overrides the value specified in the Secure+ Option remote node record *only* if the override security setting is **Y** (yes) in that remote node record. After the security settings are merged between the Pnode and Snode, the strongest setting is always used. Therefore, the value specified from the Copy command cannot disable data encryption or digital signatures if the Snode has it enabled.

If the override security setting is **N** (no) in that remote node record and the values specified on the copy command are different than the values specified in the remote node record, the copy operation fails with a return code of 8 and message ID CSPA011E indicating the error.

Secure+ Option Copy Command Parameters

The Connect:Direct for i5/OS copy commands (**CDSND**, **CDRCV**, **CDSNDSPL**) have Secure+ Option parameters. These parameters enable you to:

- Enable or disable digital signatures
- Enable or disable data encryption or specify a specific algorithm to use when encrypting data

Note: All parameters have a default value of '*.'. This value specifies that the value defined in the local node record (*LCLNODE) will be used at the run time of a copy command.
At runtime, the copy function goes through all of the values specified on the copy command and the values specified on the parameters file entries. It calculates the final value to use at the time of negotiation. At the time of negotiation, an OR operation is performed between the final calculated values of the two nodes and the most secure settings are used. The values of data encryption and digital signature apply only to copy steps.

The following table describes the SECURE parameters for the Secure+ Option copy commands.

Parameter Name and Syntax	Parameter Description	Valid Values
SECURE+ Options: Encryption valid value **or** CL/command format: SECOPTIONS (* *)	Enables/disables† Copy file encryption	* | Y| N | Algorithm name
SECURE+ Options: Digital Signature valid value **or** SECOPTIONS (* *)	Enables/disables† digital signature creation	*| Y | N

† Data encryption and digital signatures cannot always be disabled from the Copy command.

In a CL program, if both parameters are used on the Copy command, the syntax is as follows:

```
SECOPTIONS (Y Y)
```

Using the CDSND Command

Use the CDSND command to send a file to a remote node. This command sends files to any file systems supported by the i5/OS V3R1M0 or later. You can also specify a user-exit command (either **EXITCMD** or **FAILCMD**) to be executed after the completion of a Copy process in either the FMSYSOPTS or TOSYSOPTS parameter.

- Use the EXITCMD user exit to specify a command to be executed only if the Copy process is successful.
- Use the FAILCMD command to specify a command to be executed only if the Copy process is not successful

CDSND Command Example

The following figure illustrates a sample **CDSND** command.

```
CDSND SNODE(RMTNODE) SNODENVIRN(AS400) FDSN('FROMLIB/FROM FILE(FROMMBR)')
TDSN('TOLIB/TOFILE(TOMBR)') FMSYSOPTS('TYPE(MBR)') TOSYSOPTS('TYPE(MBR)')
SECOPTIONS(* *) TDISP(RPL)
```

The following **Connect:Direct Send File (CDSND)** screen displays the Secure+ Option fields.

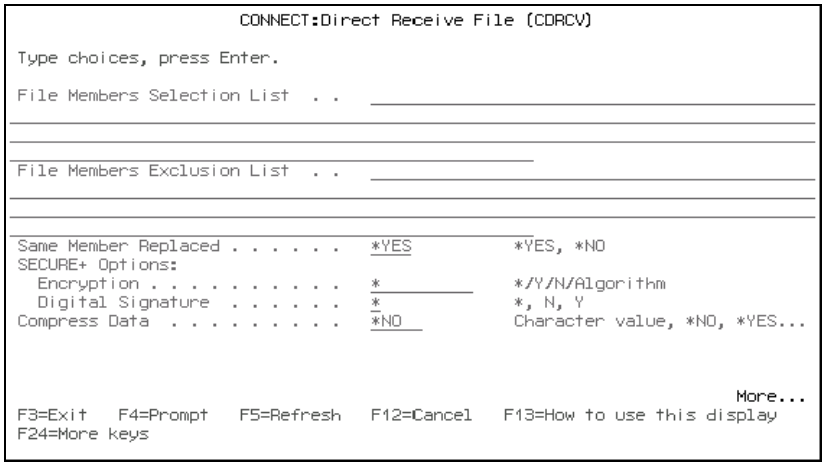
```
                    CONNECT:Direct Send File (CDSND)

Type choices, press Enter.

File Members Selection List  . .  _____________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________
File Members Exclusion List  . .  _____________________________________________
_____________________________________________________________________________
_____________________________________________________________________________
_____________________________________
Sending block size . . . . . .  *CALC          Character value, *CALC
Same Member Replaced . . . . . .  *YES         *YES, *NO
SECURE+ Options:
  Encryption . . . . . .  *_________           */Y/N/Algorithm
  Digital Signature  . . . . . .  *            *, N, Y
Compress Data  . . . . . .  *NO__              Character value, *NO, *YES...


                                                                       More...
F3=Exit   F4=Prompt   F5=Refresh   F12=Cancel   F13=How to use this display
F24=More keys
```

Using the CDRCV Command

Use the **CDRCV** command to receive a file from a remote node. This command receives files from any file systems supported by the OS/400 V3R1M0 or later. Following is an example of the **CDRCV** command:

```
CDRCV SNODE(RMTNODE) SNODENVIRN(AS400) FDSN('FROMLIB/FROMFILE(FROMMBR)')
TDSN('TOLIB/TOFILE(TOMBR)') TOSYSOPTS('TYPE(MBR)') FMSYSOPTS('TYPE(MBR)') SECOP-
TIONS(* *) TDISP(RPL)
```

The following **Connect:Direct Receive File (CDRCV)** screen displays the Secure+ Option fields.



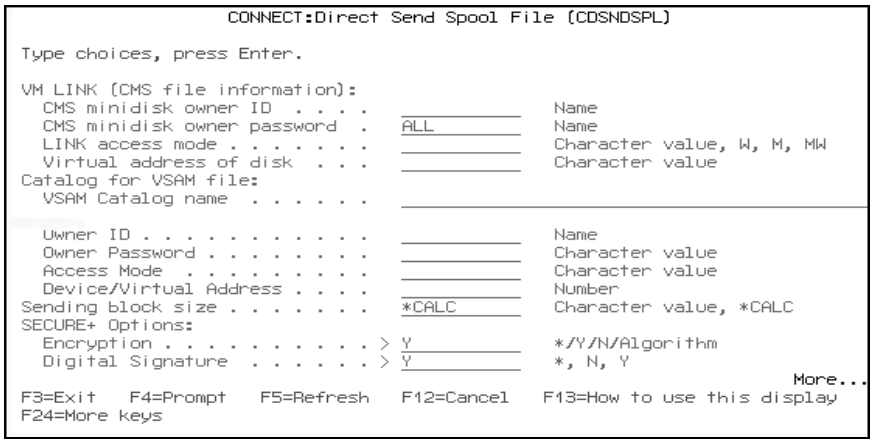
Using the CDSNDSPL Command

Use the **Connect:Direct Send Spool (CDSNDSPL)** screens to send a spooled file to a remote node.

The following figure displays a sample **CDSNDSPL** command.

```
CDSNDSPL SNODE(RMTNODE) SNODENVIRN(AS400) SPLF(SPLFILE) JOB(CDJOB CDUSER 004321)
SPLFNUM(1) TDSN('TOLIB/TOFILE(TOMBR)') CTLCHAR(*FCFC)
FDSNLEN(*SPLF)TOSYSOPTS('TYPE(MBR)') SECOPTIONS(* *) TDISP(RPL)
```

The following **Connect:Direct Send Spool File (CDSNDSPL)** screen displays Secure+ Option fields:



Chapter 3

Maintaining Secure+ Option

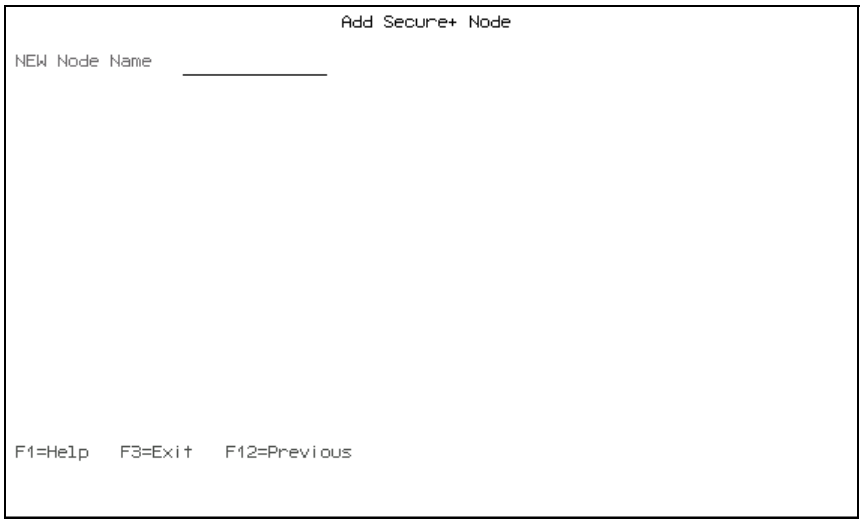
After you have set up your Secure+ Option environment, perform additional maintenance tasks as needed. This section provides procedures for performing the following Secure+ Option maintenance tasks:

- Adding a Secure+ Option remote node record
- Updating a Secure+ Option node record
- Displaying a Secure+ Option node record
- Deleting a Secure+ Option node record
- Viewing Secure+ Option node record change history
- Updating keys
- Resetting keys in remote node records
- Resecuring the Secure+ Option parameters file

Adding a Secure+ Option Remote Node Record

To add a remote node record to the Secure+ Option parameters file:

1. From the **SECURE+ ADMIN MAIN SCREEN**, press **F6**. The following screen is displayed.



2. Type the name of the remote node to add and press **Enter**. You are returned to the **SECURE+ ADMIN MAIN SCREEN**. A new remote node record is created with the default values.

Note: You must add an entry for the new remote node record to the Connect:Direct network map before you can communicate with that remote node.

---

3. From the **SECURE+ ADMIN MAIN SCREEN**, type **2** next to the name of the new remote node name. The **Update Secure+ Node** screen is displayed.
4. Make the changes as needed and press **Enter**. Refer to Chapter 2, *Installing and Setting Up Secure+ Option*, for definitions and valid values for the node record fields.
5. Press **F12** to return to the **SECURE+ ADMIN MAIN SCREEN**.

Updating a Secure+ Option Node Record

To update a node record in the Secure+ Option parameters file:

1. From the **SECURE+ ADMIN MAIN SCREEN**, type **2** next to the name of the node to update. The **Update Secure+ Node** screen is displayed.

```
05/01/04                  Update Secure+ Node                  System: Comp 2

Node Name . . . : Node A                 Node Type. . . . .  : R
Security Enabled:   Y                    Encrypt Data . . .  : Y
Override Security:  Y                    Digital Signature . : Y
Auto Update . . :   Y                    Enabled STS . . . . : Y
Auth. Time Out. :   120

Algorithm Names:
Alias Names . . .: *LCLNODE

AUT. Pub Key  TO Rmt:      0301.546A.7049.F149.OeFD.26AD.E83F.0C33
    AUT. Pub Key Expires:                Replace Prev. AUT. Key   Y
SIG.Pub Key  TO  Rmt:      0301.5D82.869.CF69.A9E1.88AC.95C7.8CE5
    SIG. Prev Key Expires:               Replace Prev. SIG. Key   Y
AUT. Pub Key FROM Rm   0305.T9DD.CE75.2444.7AE9.5C74.E10A.B9E9.94B1.FC50.0B6C
SIG.Pub Key FROM Rm:   0206.D240.B84A.20E9.8D5D.84D2.5054.F4DD.5D52.D229.3DEB


F1=Help       F2=Enbl STS      F3=Exit      F4=Enbl SSL/TLS    F6=Reset AuKey
F7=Gen AuKey   F8-Reset SigKey   F9=Gen SigKey       F12=Return    F24=More Keys
```

2. Make the changes as needed and press **Enter**. Refer to Chapter 2, *Installing and Setting Up Secure+ Option*, for definitions and valid values for the node record fields.
3. Press **F12** to return to the **SECURE+ ADMIN MAIN SCREEN**.

Updating Keys

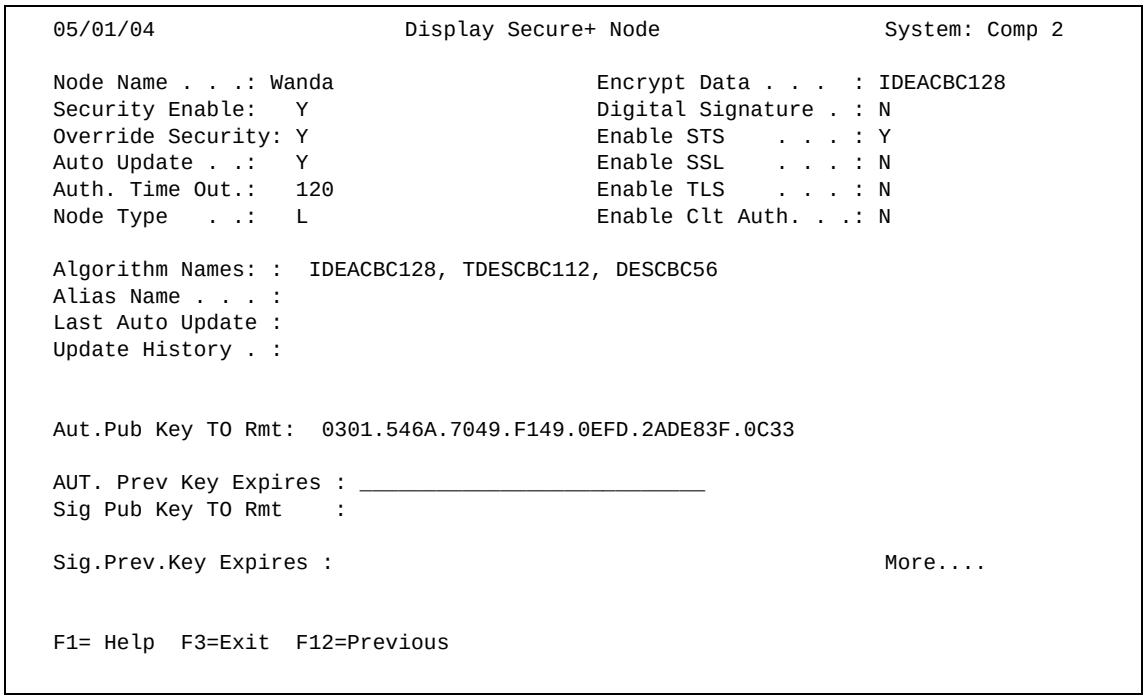
To update keys:

1. From the **SECURE+ ADMIN MAIN SCREEN**, type **2** next to the name of the node you need to update keys for and press **Enter**. The **Update Secure+ Node** screen is displayed.
2. To generate authentication keys:
   a. Press **F7**. The **Random Seed Quote Selection List** screen is displayed.
   b. Type **1** next to any quote and press **Enter**. The **Secure Quote Modification** screen is displayed.
   c. Make at least *one change* to the quote and press **Enter**. New keys are generated and saved.
3. To generate signature keys:
   a. Press **F9**. The **Random Seed Quote Selection List** screen is displayed.
   b. Type **1** next to any quote and press **Enter**. The **Secure Quote Modification** screen is displayed.
   c. Make at least *one change* to the quote and press **Enter**. New keys are generated and saved.
4. Press **F3** to exit the **Update Secure+ Node** screen and update the node record.

Displaying a Secure+ Option Node Record

To display a Secure+ Option node record:

1. Type **5** next to the name of the node to display and press **Enter.** The **Display Secure+ Node** screen is displayed.



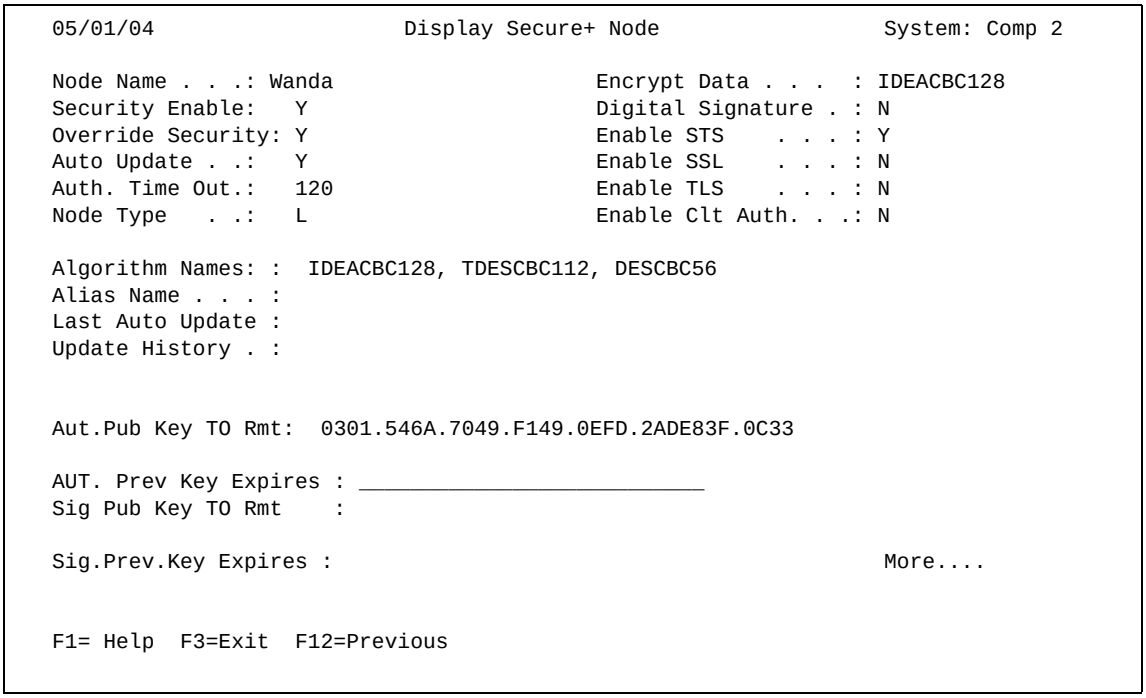
Note: The **Display Secure+ Node** panel in the above example represents a node record that enables the STS protocol. This panel displays different fields of information for nodes that enable the SSL or TLS protocol.

2. Press **F12** to go to the previous screen or **F3** to exit.

Viewing Secure+ Option Node Record Change History

Perform the following steps to view the history of changes to a Secure+ Option node record. This screen shows the last three updates made to the node.

1. From the **SECURE+ ADMIN MAIN SCREEN**, type **5** next to the name of the node to view.



2. View the **Update History** field to identify the date that the last three changes were made to the node and who made them. The Record Change History retains one copy per day of the last time-stamped activity for a given day.
3. Press **F3** to exit the screen.

Resetting Keys in Remote Node Records

Perform the following steps to reset the keys in remote node records to default to the local node record.

1. From the **SECURE+ ADMIN MAIN SCREEN**, type **2** next to the name of the node to reset keys for.
2. If you have existing Authentication keys for the node, press **F6** to reset the keys to default to the local node record value (*LCLNODE).

3. If you have existing Signature keys, press **F8** to reset the keys to default to the local node record value (*LCLNODE).
4. Press **F3** to exit the screen.

This resets the public and private keys used for authentication and digital signature in the remote node record to default to the local node record value.

Deleting a Secure+ Option Node Record

To delete a Secure+ Option node:

1. From the **SECURE+ ADMIN MAIN SCREEN**, type **4** next to the name of the node to delete and press **Enter**. To delete an alias record, type **4** next to the alias record name and press **Enter**.

> Note: The Secure+ Option parameters file local node record (*LCLNODE) *cannot* be deleted.

The **Delete Secure+ Node** confirmation screen is displayed.

```
                          Delete Secure+ Node

Press Enter to confirm the delete OR F12 to return to your choices.
---- NOTE If Node Name (T)YPE is "R" all ALIAS will be deleted ALSO.

     Node
   T Name              BASE/ALIAS-NAME   ALIAS-Name-2      ALIAS-Name-3
     R vincent
4    R vncntvango
     R wanda
4    R welcom
4    R xxx

                                                                  Bottom

F1=Help   F3=Exit   F5=Refresh   F12=Previous
```

2. Press **Enter** to delete the node record.

> Note: The Delete Node record function deletes all aliases associated with the primary remote node record.

3. Press **F12** to return to the previous screen or **F3** to exit.

Resecuring the Secure+ Option Parameters File

You should routinely rekey your Secure+ Option parameters file to maintain security. When you rekey the Secure+ Option parameters file, the parameters file is decrypted, new keys are generated, and the file is re-encrypted. To rekey the file:

1. From the **SECURE+ ADMIN MAIN SCREEN**, press **F13**. The **Random Seed Quote Selection List** screen is displayed.
2. Type **1** next to any one of the quotes and press **Enter**. The **Secure+ Quote Modification** screen is displayed.
3. Make at least *one change* to the quote and press **Enter**. The system takes you back to the **SECURE+ ADMIN MAIN SCREEN** where you receive a series of messages at the bottom of the screen explaining that the keys are being updated.

Note: Connect:Direct processing is suspended during this time.

Chapter 4

Accessing Secure+ Option Statistics

Connect:Direct logs statistics for Connect:Direct Process activities, including Secure+ Option information for a Process, such as whether or not encryption or digital signatures were enabled for both the session and the merged value between the communicating nodes for the security functions and algorithms.

You can view the results of Processes by using the **CDSELSTAT** command.

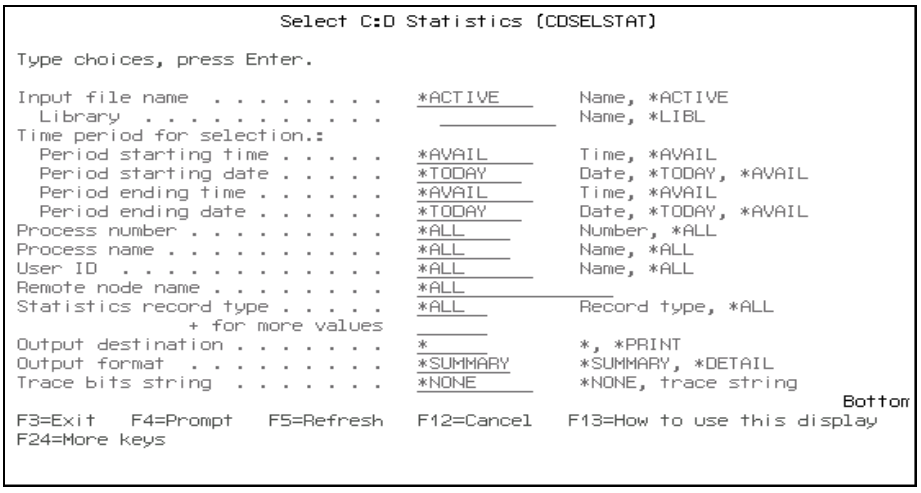
This chapter provides examples of Connect:Direct Process statistics records with the information that was added for Secure+ Option support. For information about accessing Connect:Direct for i5/OS Process statistics, refer to the *Connect:Direct for i5/OS Installation and Administration Guide*.

Statistics Record Examples

The Connect:Direct for i5/OS Process statistics records provides Secure+ Option information about the Process. The following examples show the new fields. A description of these new fields follows the examples.

When you use the **CDSELSTAT** command:

1. Press **F4** to view the information about a Connect:Direct Process. The **Select C:D Statistics (CDSELSTAT)** screen is displayed.



2. Type information in appropriate fields to identify which Processes to view statistics for.

When you use the **CDSELSTAT** command to view information about a Connect:Direct Process, the following fields display information about Secure+ Option:

Field	Description
Encryption = Yes | No | Algorithm	Specifies whether encryption is enabled in the Secure+ Option parameters file. This information is displayed in the statistics separately for the Pnode, Snode, and Merge values.
Algorithm List = Yes | No | Algorithm	Specifies whether algorithms are enabled in the Secure+ Option parameters file. This information is displayed in the statistics separately for the Pnode, Snode, and Merge values.
Digital Signature = Yes | No	Specifies whether digital signatures are enabled in the Secure+ Option parameters file. This information is displayed in the statistics separately for the Pnode, Snode, and Merged values.
Merge values	Specifies the resulting value of the merge between the Pnode and Snode Secure+ Option parameters file *and* the copy parameters for data encryption, digital signature, and algorithm List.

When you use the **CDSELSTAT** command with the *Detail option enabled to view the information about a Connect:Direct Process, you will see information similar to the following.

```
C:D/i5/OS          Select Statistics (CDSELSTAT) Detail  08/13/04  15:15:22   Page 0001
------------------------------------------------------------------------------
Event:         Communications Session Started (SMSES)
Process Name:  CDSND                                Process Number:  00000034
Date/Time:     08/13/04     15:10:06
Local Node:    CDQA33DM                             Local User:  SFLAN1
Remote Node:   CDQB33ST
Protocol:      TCPIP                                Type:  INIT
Message Id:    ASMT292I
------------------------------------------------------------------------------
                     Encryption  Digital-Signature  Algorithm List
                     ==========  ================= ======================
PNODE                N                NO          DESCBC56
SNODE                N                NO          DESCBC56
MERGE                N                NO
------------------------------------------------------------------------------
Event:         Process Started (SMPST)
Process Name:  CDSND                                Process Number:  00000034
Date/Time:     08/13/04     15:10:28
Local Node:    CDQA33DM                             Local User:  SFLAN1
Remote Node:   CDQB33ST
Submit User:   SFLAN1                               Pnode:  Y
------------------------------------------------------------------------------
```

Continued

```
      Event:         Process Step Started (SMSTST)
Process Name:  CDSND                                Process Number:  00000034
Date/Time:     08/13/04     15:10:28
Local Node:    CDQA33DM                             Local User:  SFLAN1
Remote Node:   CDQB33ST
Step Name:     STEP001                              Action:  CPY
------------------------------------------------------------------------------
Encryption Digital Signature
      ========== =================
PNODE N                NO
SNODE N                NO
MERGE N                NO
```

The following screen displays the CDSELSTAT *Summary output screen view.

```
                          Select C:D Statistics

Process number . .   1273____            Statistics file  :   CDSTATFILE
Process name . . .   *ALL____             Library  . . . . :   CDQA3300AU
                                          Position to date     ________

Type options, press Enter.
  5=Display details   6=Print details

     Process   Process   Record  Step        Event     Event     Message
Opt  number    name      type    name        time      date      ID
 _   00001273  CDSND     SMSES               15:12:04  07/16/99
 _   00001273  CDSND     SMPST               15:12:05  07/16/99
 5   00001273  CDSND     SMSTST              15:12:05  07/16/99
 _   00001273  CDSND     SMCOMP  STEP001     15:12:30  07/16/99  SCPA000I
 _   00001273  CDSND     SMSTTM              15:12:30  07/16/99
 _   00001273  CDSND     SMSTM               15:12:31  07/16/99


                                                                      Bottom
Parameters or command
===> ____________________________________________________________________________
F3=Exit   F4=Prompt   F9=Retrieve   F11=View 2   F12=Cancel
```

To view the individual record statistics, type **5** in the option column and press **Enter.** The detailed statistics for that Process are displayed.

```
                          Display Spooled File
File  . . . . . :   CDSELSTAT                      Page/Line   1/6
Control . . . . .   ________                       Columns     1 - 78
Find  . . . . . .   ____________________________
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
C:D/400          Select Statistics (CDSELSTAT) Detail  08/03/99  12:11:13   Pa
------------------------------------------------------------------------------
Event:         Communications Session Started (SMSES)
Process Name:  CDSND                               Process Number:  00001221
Date/Time:     07/12/99     14:22:13
Local Node:    hotpepper                           Local User:  QACDADMIN
Remote Node:   wanda
Protocol:      TCPIP                               Type:  INIT
Message Id:    ASMT292I
------------------------------------------------------------------------------
                       Encryption  Digital-Signature  Algorithm List
                       ==========  ================= ==========================
PNODE                  N                NO          IDEACBC128 TDESCBC112 DESC
SNODE                  Y                NO          DESCBC56   TDESCBC112 IDEA
MERGE                  DESCBC56         NO
------------------------------------------------------------------------------
                                                                      Bottom
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

Chapter 5

Troubleshooting

Use the following table to help troubleshoot problems that you may encounter when using Secure+ Option.

Problem	Possible Cause	Solution
Secure+ Option features are enabled in the Secure+ Option parameters file, but the statistics record indicates that these functions are disabled.	• The Connect:Direct network maps do not contain entries for the Pnode and Snode. The Connect:Direct for i5/OS license management key file is not valid for use with Secure+ Option. The node that you are connecting with is a V1 flow (such as LU0). Secure+ Option is not supported for V1 flows.	• Verify that the network map entries for both the Pnode and the Snode exist, and use a V2 protocol such as LU62 or TCP/IP. • Replace the Connect:Direct for i5/OS license management key file with one that is valid for use with Secure+ Option. • Verify that the Secure+ Option node that you are connecting to is a V2 flow.
Secure parameters specified from a copy command cause the copy step to fail with message CSPA077E.	• The node that you are connecting with is a V1 flow (such as LU0). Secure+ Option is not supported for V1 flows.	• Verify that the Secure+ Option node that you are connecting to is a V2 flow.
Secure+ Option is installed, but error message CSPA001E occurs on non-Secure+ Option transfers.	• A remote node record does not exist in the Secure+ Option parameters file for that node and your local Secure+ Option is enabled from the local node record.	• Use the **F7=Sync-Add** feature to add node records from your Connect:Direct network map that do not exist in your Secure+ Option parameters file. • You can also disable Secure+ Option from the local node by setting the Local Node record to **Security Enable = N** and **Override Security = N**.

Problem	Possible Cause	Solution
An error occurs while running a process with a remote node and the process fails.	• The Limited Export value defined in the remote node record does not match the Secure+ Option export version Limited Export or Export) for that remote node. If NodeA is Export version and the value defined in the NodeB's remote node record for communicating with NodeA is Limited Export, the elliptic curves used to create keys and generate Diffie-Hellman shared secrets are not correct.	• Verify that the remote node definitions on both sites accurately state the Secure+ Option Export information.
Running a Process with a remote node fails with an authentication error.	• Unique public/private key pairs are generated for the remote node record and the local node record **Override Security** value is **N**.	• Change the local node record **Override Security** value to **Y** *or* do not use unique public/private key pairs in the remote node record.
The Secure+ Option options: Encryption:* * / Y / N / Algorithm specified from the copy command causes the copy step to fail with error message CSPA080E.	• The algorithm name used in the copy command is not in the supported algorithm list for both nodes.	• Verify that the algorithm name in the copy command is in the supported algorithm list for both nodes.
Secure+ Option session fails with CSPA080E.	• There are no common algorithms in the algorithm list for both nodes.	• Verify the algorithm list for both nodes contains at least one common algorithm name.
Session failing with CSPA002E error.	• Configuration settings missing or incorrect.	• If this is a non-Secure node, make sure the remote node record value for **Security Enable** is **N**.
Running a Process with a remote node fails with an authentication error, CSPA008E.	• The **Aut.Prev.Keys Expires** date is exceeded or keys have been changed.	• If **Auto Update** is disabled, check the authentication previous key pair expiration date for both nodes. Check the update history log on both nodes for the last change to the record. Verify the authentication public key is correct for both nodes.
Signature verification fails with error message CSPA007E.	• The **Sig.Prev.Keys Expires** date is exceeded or keys have been changed.	• If **Auto Update** is disabled, check the signature previous key pair expiration date for both nodes. Check the update history on both nodes for the last change to the record. Verify the signature public key is correct for both nodes.
The session fails with error message CSPA215E	• The remote node rejected the handshake.	• Make sure that the certificate is valid. • Check the SSL configuration on both nodes and make sure that the settings match.
The session fails with the error message CSPA216E	• A socket error occurred during loopback processing.	• Verify the TCP settings.

Appendix A

Configuration Worksheets

Use the worksheets in this appendix to record the configuration information for Connect:Direct Secure+ Option.

- The *Local Node Record Security Feature Definition Worksheet* is a record of the defined security functions for the local Connect:Direct node.
- The *Remote Node Record Security Feature Definition Worksheet* is a record of the defined security functions for connections with remote nodes with which the local Connect:Direct node communicates. Make a copy of the blank *Remote Node Record Security Feature Definition Worksheet* for each remote node record that you are configuring for Secure+ Option operations.
- The *Export/Import Worksheet* is a record used for importing and exporting your public keys.

Local Node Record Security Feature Definition Worksheet

Record the security feature definitions for the Secure+ Option local node record on this worksheet.

Local Node Name: _____

TLS protocol enabled: Yes _____ No _____

SSL protocol enabled: Yes _____ No _____

STS protocol enabled: Yes _____ No _____

Configured Security Functions

SSL and TLS Options

Application ID: _____

Cipher Suite(s) to Enable: _____

STS Options

Override enabled: Yes _____

Autoupdate enabled: Yes _____

Auth. Time Out: _____ (A numeric value equal to or greater than 0 seconds)

Enable Digital Signatures: Yes _____ No _____

Create Public Key : Yes _____ No _____

Enable Encryption: Yes _____ No _____

Algorithm Names: _____

Algorithms Enabled ___ DES ___ TDES ___ IDEA

Local Node Name: _____

Remote Node Record Security Feature Definition Worksheet

Record the security feature definitions for a remote node record on this worksheet. Make a copy of this worksheet for each remote node defined in the Secure+ Option parameters file that you are configuring for Secure+ Option operations.

Security Options

TLS protocol enabled: Yes _____ No _____

SSL protocol enabled: Yes _____ No _____

STS protocol enabled: Yes _____ No _____

Enable Override: Yes _____ No _____

Note: If you want to use the COPY statement to override settings in the parameters file for STS-enabled nodes, enable **Override** for the remote node. The COPY statement cannot override settings in the SSL protocol.

Authorization Timeout: _____ (A numeric value equal to or greater than 0 seconds)

TLS or SSL Protocol Functions

If you did not define this information in the local node record, set one or more of the following functions:

Application ID: _____

Cipher Suite(s) Enabled: _____

Note: Ask the trading partner which cipher suites are enabled.

If you want to enable Client authentication, set the following two options:

Enable Client Authentication: Yes _____ No _____

If client authentication enabled, Client Authentication Common Name: _____

_____

Note: If you want to add a second level of security, enable client authentication for the remote node and type the certificate common name.

STS Protocol Functions

If you enabled the STS protocol, set one or more of the following functions:

Enable Digital Signatures: Yes _____ No _____

Enable Public Key Auto Updates: Yes _____ No _____

Limited Export Version: Yes _____ No _____

Note: If the trading partner uses an earlier version of Secure+ Option, you need identify the version of Secure+ Option the partner is using.

Enable Encryption: Yes _____ No _____

Algorithm Names: _____

Algorithms Enabled ___ DES ___ TDES ___ IDEA

Export/Import Worksheet

Record the information needed to import and export keys and activate the STS protocol on this worksheet.

Local Node Name _____

Remote Node Name _____

Export File / _____/

Export Library Import Library Export Member

Import File / _____/

Export Library Import Library Export Member

Local Node Name _____

Remote Node Name _____

Export File / _____/

Export Library Import Library Export Member

Import File / _____/

Export Library Import Library Export Member

Local Node Name _____

Remote Node Name _____

Export File / _____/

Export Library Import Library Export Member

Import File / _____/

Export Library Import Library Export Member

---

† Export or Import libraries cannot be given a Connect:Direct library name and must exist prior to execution of the Import or Export function. Recommended default names for the libraries are CDEXPORT and CDIMPORT. Both of these libraries are created at the time of Secure+ Option installation.
It is recommended that file and member names be unique for each remote node entry.
The transfer of these files from one node to another must be done in a binary format.

Appendix B

Testing Secure+ Option with the STS Protocol

Before using the STS protocol in a production environment, test the installation to ensure Connect:Direct Secure+ Option for i5/OS is operational. The parameters file is initially populated with two node records: the local record *and* a remote record for the associated Connect:Direct node. The remote record for this local node is created to support loopback (Pnode=Snode) processing to test Connect:Direct Secure+ Option for i5/OS. This section provides information for setting the options for the initial Pnode=Snode testing of Connect:Direct Secure+ Option for i5/OS for use with the STS protocol.

Setting Up the Local and Remote Node for Testing

To use the loopback node to test the Secure+ Option installation, you must:

- Change the override option in the local node record
- Set security options in the remote node record
- Export the remote node record's public keys
- Import the local node record (*LCLNODE) public keys
- Send a test file and verify the results

Configuring the Local Node Record

Complete the following steps to set up the local node record:

1. Start the **Secure+ Administration Tool** by executing the **SPADMIN** command. The **SECURE+ ADMIN MAIN SCREEN** is displayed.
2. Type **2** next to the local node record (*LCLNODE) and press **Enter** to open the local node record. The **Update Secure+ Node** screen is displayed.

```
05/01/04                  Update Secure+ Node                  System: Comp 2

Node Name . . .: *LCLNODE                Limited Export . .  : N
Security Enable:   Y                     Encrypt Data . . .  : Y
Override Security: Y                     Digital Signature . : Y
Auto Update . .:   Y                     Enable STS    . . . : Y
Auth. Time Out.:   120                   Enable SSL    . . . : N
Node Type   . .:   L                     Enable TLS    . . . : N
                                         Enable Client Auth. : N

Alias Names . . .: *LCLNODE
Algorithm Names:

                                                               Expire Date Rpl
AUT. Pub Key  TO Rmt:                                          5/01/06
SIG.Pub Key  TO  Rmt:                                          5/1/07
AUT.Pub Key FROM Rmt:
SIG. Pub  Key TO Rmt:
Application ID      :
Cipher Suites . .  :
Cert Common Name . :

F3=Exit     F4=Cipher Suites   F5=Refresh   F6=Reset Auth Key    F7=GEN. Auth. Key
F8=Reset Sig. Key          F9=GEN. Sig. Key        F12=Return       F24=More Keys
```

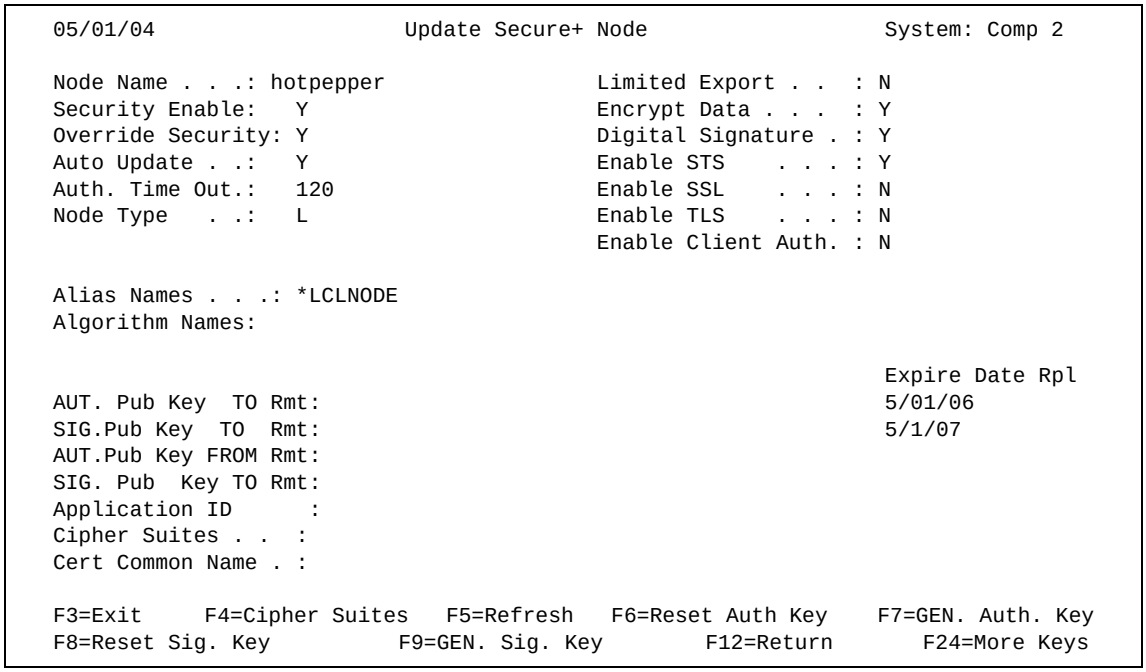
3. Type **Y** in the **Override Security** field to turn override security on.
4. Press **Enter**.

Note: The first time that you initialize the Secure+ Option Admin Tool, the *to remote* keys for the local node record and the *from remote* keys for the remote node record are automatically generated.

Configuring the Remote Node Record for Testing

To test the installation using the loopback remote node record created at installation, complete the following procedure to update the remote record named for your local node:

1. From the **SECURE+ ADMIN MAIN SCREEN**, type **2** next to the loopback node name and press **Enter**. The **Update Secure+ Node** screen is displayed.



2. Modify the field values as follows:
   - Security Enable=**Y**
   - Override Security=**Y**
   - Digital Signature=**Y**
3. Press **Enter**.

Note: The first time that you initialize the Secure+ Option Admin Tool, the *to remote* keys for the local node record and the *from remote* keys for the remote node record are automatically generated.

Sending a Test File and Verifying Results

Perform the following steps to test your installation with the loopback node and to verify the results:

1. Transfer a small file using a copy command such as **CDSND** with this loopback entry setup.
2. Verify results through the statistics file (**CDSELSTAT**) command. You should be able to see Secure+ Option related information within the event records similar to the following example.

```
                              Display Spooled File
File  . . . . . :   CDSELSTAT                          Page/Line   1/6
Control . . . . .   ________                           Columns     1 - 78
Find  . . . . . .   ____________________
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
C:D/400          Select Statistics (CDSELSTAT) Detail  07/23/99  10:15:18   Pa
------------------------------------------------------------------------------
Event:          Communications Session Started (SMSES)
Process Name:   *UNKNOWN                               Process Number:  00000000
Date/Time:      07/23/99     10:13:53
Local Node:     AGUSTA.3.3.00                          Local User:  *UNKNOWN
Remote Node:    DB.MVS.RTINN1
Protocol:       TCPIP                                  Type:  EVK
Message Id:     Session is connected to the remote node (informational).
------------------------------------------------------------------------------
                      Encryption  Digital-Signature  Algorithm List
                      ==========  ================== ==========================
PNODE                 N
SNODE                 N
MERGE                 N
------------------------------------------------------------------------------
                                                                         Bottom
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

3. Once you have defined Secure+ Option for the loopback node installation test node and tested the installation using the loopback node, test the Secure+ Option functionality with a trading partner. You can then proceed with setting up all nodes with which you exchange data.

Testing with a Remote Node

Once you have installed Secure+ Option, to test your installation and verify that Secure+ Option is working as intended, you can test with a remote node (trading partner) rather than your loopback node. You and your trading partner must perform all of the tasks described in this section, including the creation and transfer of public keys. You must perform the following tasks to set up your Secure+ Option environment for testing with a trading partner:

- Set Secure+ Option options in the local node record
- Set Secure+ Option options and generate keys in the remote node record
- Export public keys and send to your trading partner
- Import public keys received from your trading partner
- Send a test file and verify results

Setting Secure+ Options in the Local Node Record

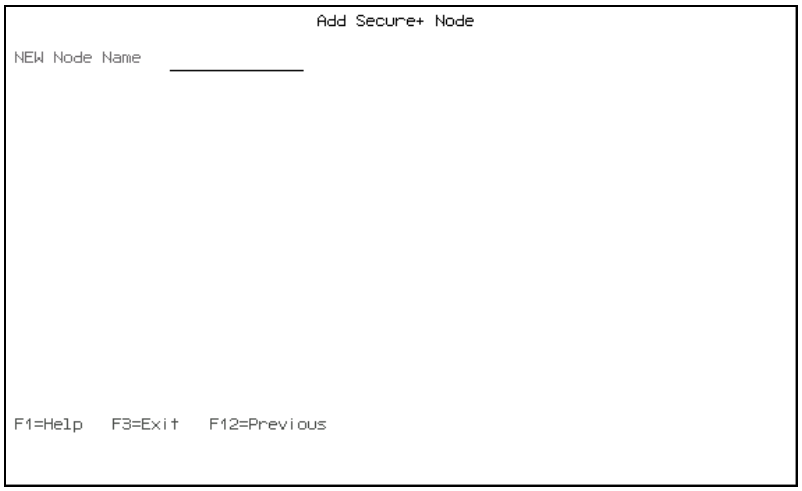
Perform the following steps to set the Secure+ Option parameters in the local node record (*LCLNODE) for testing with a trading partner:

1. Start the **Secure+ Administration Tool** by executing the **SPADMIN** command.
2. From the **SECURE+ ADMIN MAIN SCREEN**, open the local node (*LCLNODE) record by typing option **2** next to the node name and press **Enter** to update its attributes. The **Update Secure+ Node** screen is displayed.
3. Change the value for **Override Security** to **Y**.
4. Press **Enter**.

Creating a Remote Node Record and Changing Secure+ Options

Perform the following steps to create a the remote node record for the trading partner that you are testing with and change the Secure+ Option configuration:

1. On the **SECURE+ ADMIN MAIN SCREEN**, press the **F6** key to add a node. The **Add Secure+ Node** screen is displayed.



2. Type the trading partner's node name in the **New Node Name** field.

Note: A matching node entry in the Connect:Direct network map does not have exist at this time, however, it needs to be created prior to file transfer. For best results, test Secure+ Option with a trading partner that is already defined in your Connect:Direct network map.

3. Press **Enter**. The **Update Secure+ Node** screen is displayed.

```
05/01/04                    Update Secure+ Node                    System: Comp 2

Node Name . . .: wanda                    Limited Export . .  : N
Security Enable:   Y                      Encrypt Data . . .  : Y
Override Security: Y                      Digital Signature . : Y
Auto Update . .:   Y                      Enable STS    . . . : Y
Auth. Time Out.:   120                    Enable SSL    . . . : N
Node Type   . .:   L                      Enable TLS    . . . : N
                                          Enable Client Auth. : N

Alias Names . . .: *LCLNODE
Algorithm Names:

                                                                 Expire Date Rpl
AUT. Pub Key  TO Rmt:                                            5/01/06
SIG.Pub Key  TO  Rmt:                                            5/1/07
AUT.Pub Key FROM Rmt:
SIG. Pub  Key TO Rmt:
Application ID      :
Cipher Suites . .  :
Cert Common Name . :

F3=Exit     F4=Cipher Suites   F5=Refresh   F6=Reset Auth Key    F7=GEN. Auth. Key
F8=Reset Sig. Key          F9=GEN. Sig. Key        F12=Return       F24=More Keys
```

4. Modify the field values as follows:
   - Security Enable=**N**
   - Override Security=**Y**
   - AUTO Update=**Y**
   - Digital Signature=**Y**

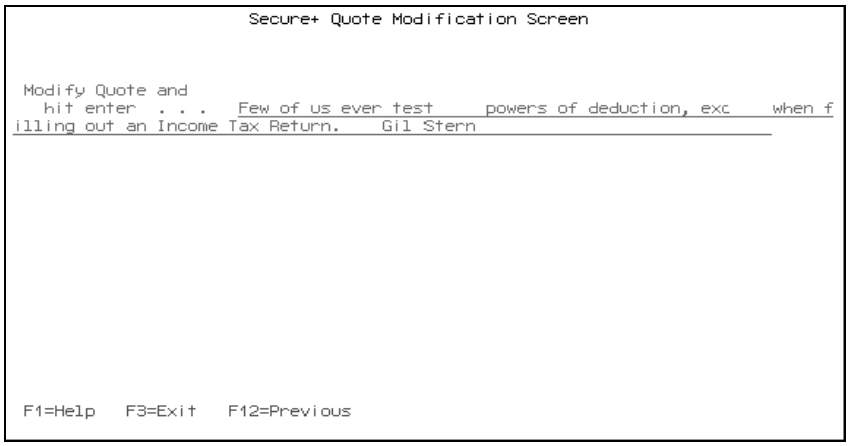
> Note: You must set Security Enable to **N** until you have exchanged keys with your trading partner.

5. To create the Authentication keys for the remote node record:
   a. Press **F7**. The **Random Seed Quote Selection List** screen is displayed.

```
                        Random Seed Quote Selection list

Type Option 1 next to the QUOTE to be selected for modification
  1=Select Quote

Opt  Quotes
 1   Few of us ever test our powers of deduction, except when filling out an I
 _   That money talks I'll not deny. I heard it once,  it said 'GoodBye'.
 _   Don't gamble! Buy some good stock & hold it till it goes up, then sell it
 _   As they say in poker, 'If you've been in the game 30 minutes and you don'
 _   Like so many, This author has made his worstinvestment mistakes when he w
 _   The point to remember is that what the  government gives, it must first t
 _   'Now boys,' said the hopeful soul at poker, 'if we all play carefully, we

                                                                      Bottom

F1=Help   F3=Exit   F12=Previous
```

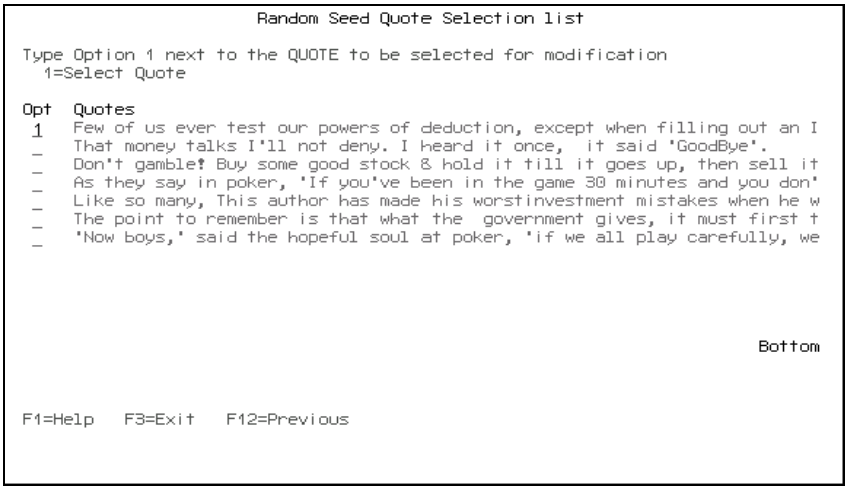
   b. Choose a quote to change by typing the number **1** next to the quote and press **Enter**.

c. At the **Secure Quote Modification** screen, make at least one change to the chosen quote and press **Enter** to generate your Authentication key.

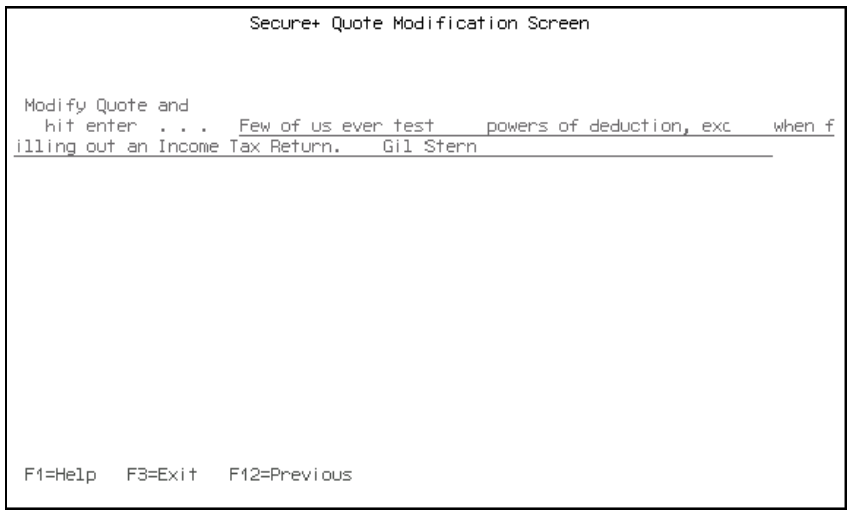


6. To create the Signature keys for the remote node record:

a. Press **F9**. The **Random Seed Quote Selection List** screen is displayed.

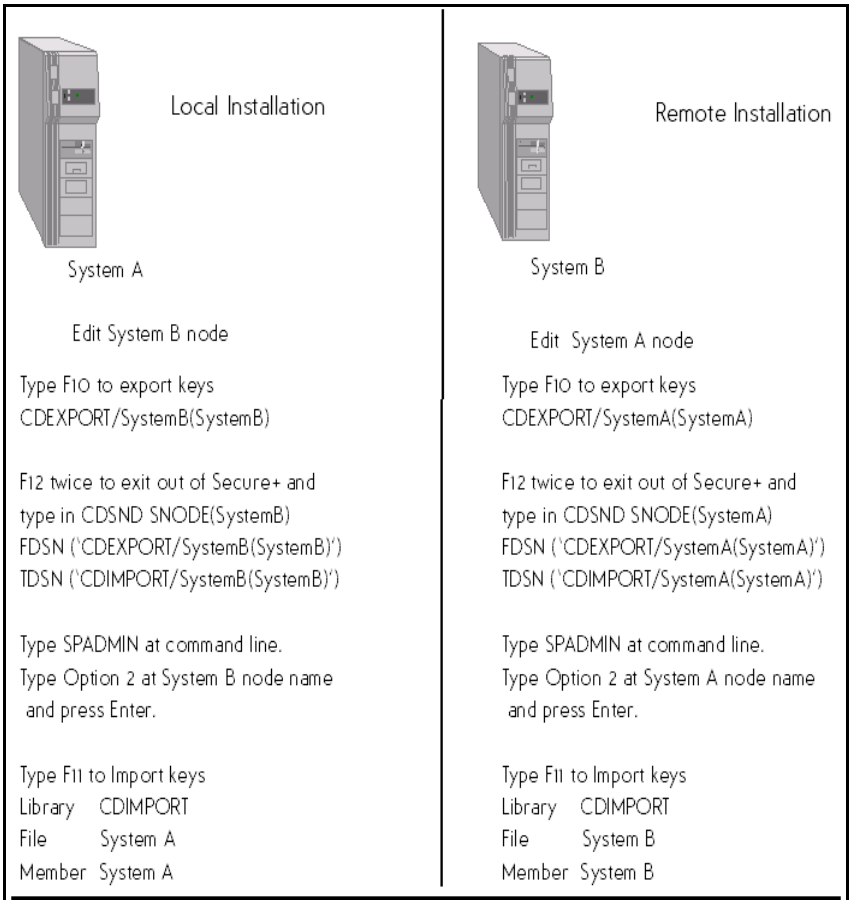


b. Choose a quote to alter and type the number **1** next to it. The **Secure+ Quote Modification** screen is displayed.



c. Make at least one change to the quote and press **Enter** to save your work. A message is displayed to indicate whether or not the key creation was successful and you are returned to the **Update Secure+ Node** screen.

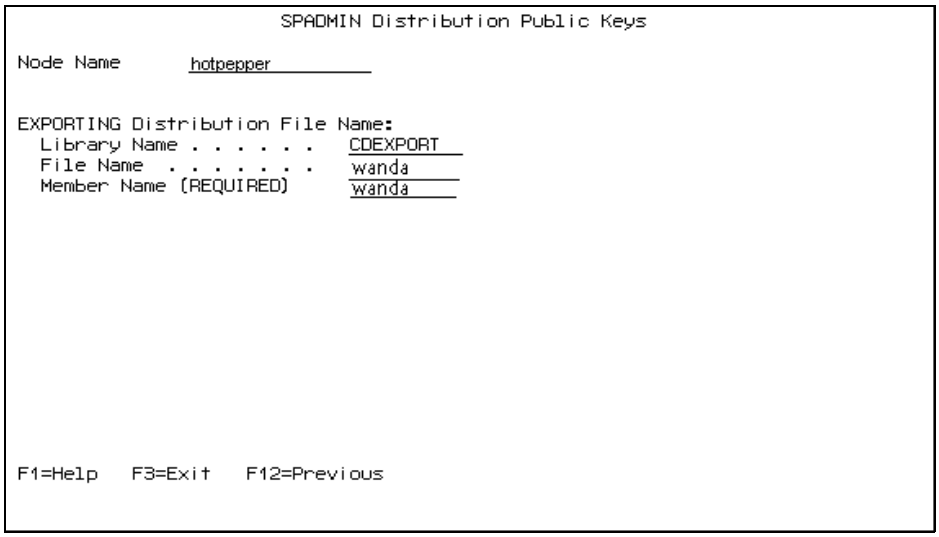
The following example illustrates the manual key exchange process between two Connect:Direct Secure+ Option for i5/OS installations.



Export Public Keys

Perform the following steps to export the public keys to a file to send to your trading partner:

1. From the **Update Secure+ Node** screen, press **F10**. The following screen is displayed.



2. Type in the Library, File, and Member names for an export file. Do not use your Connect:Direct Library name.
3. Press **Enter** to accept and process the procedure. The public keys are transferred to the specified library in the Export file with the member name.

Note: The imported keys are validated for correct format and key size based on the remote node's (trading partner) Limited Export settings.
Do not store the file export keyfile in the Connect:Direct library.

Changing the Limited Export value in a remote node record causes the "from remote" keys used for authentication and digital signatures to be cleared in that remote node record and new "to remote" to be generated. If this happens, your keys and the remote keys will not match and you must each export, exchange keys, and import the new keys before another Secure+ Option transfer occurs.

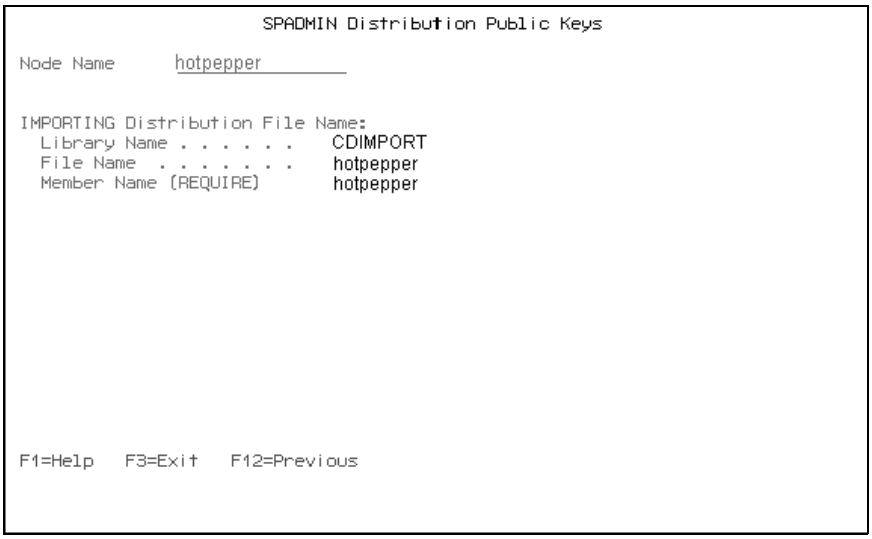
---

4. Send the exported keyfile to your trading partner using Connect:Direct for i5/OS. Check with your trading partner for the proper file structure for sending the export keyfile.

Import Public Keys

Perform the following steps to manually import the public keys sent to you from your trading partner. The remote node's Public Key file should be received as a binary/data file.

1. From the **Update Secure+ Node** screen, press **F11** to import the Authentication and Signature public key values received from your trading partner.



2. Fill in the field values as follows:
   - Library name = CDIMPORT
   - File Name = local node name
   - Member Name = local node name
3. Press **Enter** to accept and complete the process. The Authentication and Signature keys are imported from the file and member specified on the screen. The import data is translated to EBCDIC then parsed to extract and check node names, and key values. Node names are case sensitive. If the node name in the import file does not match the node name in the remote node record, a message is issued, *IMPORT name does not match*.
4. Compare results. All key values must match.
5. Press the **F12** key twice to return to the main screen and exit the **SPADMIN** command.

Change Security Enable Settings in the Remote Node Record

Before you can test your Secure+ Option installation with a trading partner, you must both enable Secure+ Option.

❖ Change the **Security Enable** value for the remote node record to **Y** and ensure that the **Override Security** value is **Y** in the local node record (*LCLNODE). Refer to the instructions beginning on page 69 for information about accessing the remote node record and changing the value.

Send a Test File and Verify Results

Perform the following steps to test your Secure+ Option installation and verify the results:

1. Transfer a small file using a copy command such as **CDSND** with this remote node entry setup.
2. Verify results through the statistics file (**CDSELSTAT**) command. You should be able to see Secure+ Option related information within the event records similar to the following example.

```
                              Display Spooled File
File  . . . . . :   CDSELSTAT                        Page/Line   1/6
Control . . . . .   ________                         Columns     1 - 78
Find  . . . . . .   ____________________________
*...+....1....+....2....+....3....+....4....+....5....+....6....+....7....+...
C:D/400          Select Statistics (CDSELSTAT) Detail  08/03/99  11:35:09   Pa
------------------------------------------------------------------------------
Event:         Communications Session Started (SMSES)
Process Name:  CDSND                                 Process Number:  00001273
Date/Time:     07/16/99     15:12:04
Local Node:    hotpepper                             Local User:  QACDADMIN
Remote Node:   hotpepper
Protocol:      TCPIP                                 Type:  INIT
Message Id:    ASMT292I
------------------------------------------------------------------------------
                      Encryption  Digital-Signature  Algorithm List
                      ==========  ================= ==========================
PNODE                 Y                NO           DESCBC56
SNODE                 Y                NO           DESCBC56
MERGE                 DESCBC56         NO
------------------------------------------------------------------------------
                                                                        Bottom
F3=Exit   F12=Cancel   F19=Left   F20=Right   F24=More keys
```

3. Once you have defined Secure+ Option for the test node and tested the operation, proceed with setting up all nodes with which you exchange data.

Glossary

A

Application ID

A variable defined in the IBM Digital Certificate Manager (DCM) to identify support for Secure+ Option.

Asymmetric key encryption

Process using two separate, but closely integrated, encryption keys: one public and one private. Each key is one way in that something encrypted using the public key can only be decrypted using the private key and vice versa. Something encrypted using the public key cannot be decrypted using the same public key.

Asymmetric keys

A separate, but integrated, user key pair comprised of one public key and one private key. Each key is one way meaning that a key used to encrypt information cannot be used to decrypt information (e.g. data encrypted with the public key can only be decrypted using the private key of that particular key pair).

Authentication

The process of ensuring the identity of various connecting user or device participants exchanging electronic data. Verifies the person or server at either end of a message is in fact who they/it claim to be and not an impostor. Provides assurance that participants are legitimate persons or devices with appropriate information access permissions.

C

Certificate

A document obtained from a certificate authority (CA) by generating a certificate signing request (CSR) that contains specific information in a specific format about the requester. It typically contains (1) distinguished name and public key of the server or client; (2) distinguished name and digital signature of the CA; (3) period of validity (certificates expire and must be renewed); and (4) administrative and extended information. The CA analyzes the CSR fields, validates the accuracy of the fields, generates a certificate, and sends it to the requester.

Certificate Authority

A company responsible for verifying and processing certificate requests, and issuing and managing certificates. The CA you choose should be one that your trading partners trust. You must meet the requirements for the CA you choose.

Certificate Revocation List

A list of certificates that have been revoked.

Certificate Signing Request (CSR)

An output file sent through E-mail to a certificate authority to request an X.509 certificate.

Cipher Suite

A cryptographic key exchange algorithm that enables you to encrypt and decrypt files and messages with the SSL protocol.

Cipher Text

Data that is encrypted. Cipher text is unreadable until it is converted into plain text (decrypted) with a key.

Client

The entity that initiates a communication session. See also Primary Node.

Client Authentication

A level of security that requires the client to authenticate its identity to the server by sending its certificate. The server must request a certificate before the client sends it.

Configuration File

A file that contains instructions and definitions upon which the system bases its processing.

Commands

Connect:Direct commands initiate and monitor activity within the Connect:Direct system.

Confidentiality

Assurance that data is not read or accessed by unauthorized persons.

D

Decryption

The process of transforming encrypted data back into meaningful information.

Digital Certificate Manager (DCM)

An IBM tool used to create and manage certificates and trusted root files.

Digital signature

Process using public and private keys to verify participant identity in the exchange of electronic information. A digital signature uniquely authenticates the person "signing" an electronic document much like a human signature uniquely identifies the person signing their name to a physical document. Because a private key is unique to each person, a value encrypted using the senders private key and subsequently decrypted using the senders public key authenticates the senders identity.

E

Encryption

The process of converting meaningful data into a meaningless form to protect the confidentiality of sensitive information.

Encryption algorithm

The set of mathematical logic that converts (encrypts/decrypts) data.

I

Integrity

Assurance that data is not modified (by unauthorized persons) during storage or transmittal.

K

Key

A unique numerical value which feeds into an encryption algorithm, setting the encryption or decryption process into motion.

N

Network map (Netmap)

The Network Map (netmap) is a file that identifies all valid Connect:Direct nodes in the network. One Network Map is associated with each Connect:Direct local node. The netmap has one entry for each of the other Connect:Direct nodes to which the local Connect:Direct node communicates. The netmap entries also contain the rules or protocol that the nodes adhere to when communicating.

P

Private key

The privately held "secret" component of an integrated asymmetric key pair.

Public key

The publicly available component of an integrated asymmetric key pair.

S

Secondary Node (SNODE)

The Connect:Direct node that interacts with the primary node (PNODE) during Connect:Direct Process execution and is the non-controlling node. Every Process has one secondary node and one primary node.

Secure Sockets Layer (SSL)

A protocol that provides secure communications with transport protocols, including FTP, over TCP/IP. It is an open, non-proprietary Internet protocol that is widely adopted as standard. SSL ensures point-to-point security, meaning that the data is secured as it is transmitted across a single socket.

Self-Signed Certificate

A self-generated certificate that identifies your organization. It is often used during the period between your request and receipt of a certificate from a public certificate authority. If self-signed certificates are used, the trusted root signing certificate must be installed in the client manually.

Server

The location that receives communication from a client.

Session Key

Cryptography key intended to encrypt data for a limited period of time, typically only for a single communications session between a pair of entities. When the session is over, the key is discarded and a new one established when a new session takes place.

Station-to-Station Protocol (STS)

A three-pass variation of the basic Diffie-Hellman protocol. It allows you to establish a shared secret key between two nodes with mutual entity authentication. Nodes are authenticated using digital signatures to sign and verify messages or control blocks.

T

Transport Layer Security (TLS)

A protocol that provides secure communications with transport protocols, including FTP, over TCP/IP. It is an open, non-proprietary Internet protocol that is widely adopted as standard. TLS ensures point-to-point security, meaning that the data is secured as it is transmitted across a single socket.

Both the SSL protocol and the TLS protocol manage secure communication in a similar way. However, TLS provides a more secure method for managing authentication and exchanging messages. TLS uses Key-Hashing for Message Authentication Code (HMAC), to ensure that a record cannot be altered during transmission over an open network such as the Internet. TLS uses the Enhanced Pseudorandom Function (PRF), to generate key data, with the HMAC and uses two hash algorithms to guarantee security. Two algorithms increase security by preventing the data from being changed if only one algorithm is compromised. The data remains secure as long as the second algorithm is not exposed.

Trusted Root Certificate File

A file stored in a library on the client that contains a list of trusted sources. During FTP connections, the client compares the server certificate, or vice versa, to the trusted root certificate file to determine if the server certificate was signed by a trusted source. The client can establish a secure FTP connection if a trusted source signed the server certificate.

U

Unsecure Connection

An FTP connection that has no security.

X

X.509 Certificate

Public key certificate specification developed as part of the X.500 directory specification, and often used in public key systems.

Index

H

I

K

L

M

N

O

P

R

S

T

U