

Connect:Direct® Secure+ Option for z/OS

Implementation Guide

Version 5.0

**Connect:Direct Secure+ Option for z/OS Implementation Guide
Version 5.0
First Edition**

(c) Copyright 1999-2010 Sterling Commerce, Inc. All rights reserved. Additional copyright information is located at the end of the release notes.

STERLING COMMERCE SOFTWARE

TRADE SECRET NOTICE

THE CONNECT:DIRECT SECURE+ OPTION SOFTWARE ("STERLING COMMERCE SOFTWARE") IS THE CONFIDENTIAL AND TRADE SECRET PROPERTY OF STERLING COMMERCE, INC., ITS AFFILIATED COMPANIES OR ITS OR THEIR LICENSORS, AND IS PROVIDED UNDER THE TERMS OF A LICENSE AGREEMENT. NO DUPLICATION OR DISCLOSURE WITHOUT PRIOR WRITTEN PERMISSION. RESTRICTED RIGHTS.

This documentation, the Sterling Commerce Software it describes, and the information and know-how they contain constitute the proprietary, confidential and valuable trade secret information of Sterling Commerce, Inc., its affiliated companies or its or their licensors, and may not be used for any unauthorized purpose, or disclosed to others without the prior written permission of the applicable Sterling Commerce entity. This documentation and the Sterling Commerce Software that it describes have been provided pursuant to a license agreement that contains prohibitions against and/or restrictions on their copying, modification and use. Duplication, in whole or in part, if and when permitted, shall bear this notice and the Sterling Commerce, Inc. copyright notice. As and when provided to any governmental entity, government contractor or subcontractor subject to the FARs, this documentation is provided with RESTRICTED RIGHTS under Title 48 52.227-19. Further, as and when provided to any governmental entity, government contractor or subcontractor subject to DFARs, this documentation and the Sterling Commerce Software it describes are provided pursuant to the customary Sterling Commerce license, as described in Title 48 CFR 227-7202 with respect to commercial software and commercial software documentation.

These terms of use shall be governed by the laws of the State of Ohio, USA, without regard to its conflict of laws provisions. If you are accessing the Sterling Commerce Software under an executed agreement, then nothing in these terms and conditions supersedes or modifies the executed agreement.

Where any of the Sterling Commerce Software or Third Party Software is used, duplicated or disclosed by or to the United States government or a government contractor or subcontractor, it is provided with RESTRICTED RIGHTS as defined in Title 48 CFR 52.227-19 and is subject to the following: Title 48 CFR 2.101, 52.227-19, 227.7201 through 227.7202-4, FAR 52.227-14, and FAR 52.227-19(c)(1-2) and (6/87), and where applicable, the customary Sterling Commerce license, as described in Title 48 CFR 227-7202 with respect to commercial software and commercial software documentation including DFAR 252.227-7013, DFAR 252.227-7014, DFAR 252.227-7015 and DFAR 252.227-7018, all as applicable.

The Sterling Commerce Software and the related documentation are licensed either "AS IS" or with a limited warranty, as described in the Sterling Commerce license agreement. Other than any limited warranties provided, NO OTHER WARRANTY IS EXPRESSED AND NONE SHALL BE IMPLIED, INCLUDING THE WARRANTIES OF MERCHANTABILITY AND FITNESS FOR USE OR FOR A PARTICULAR PURPOSE. The applicable Sterling Commerce entity reserves the right to revise this publication from time to time and to make changes in the content hereof without the obligation to notify any person or entity of such revisions or changes.

Connect:Direct is a registered trademark of Sterling Commerce. Connect:Enterprise is a registered trademark of Sterling Commerce, U.S. Patent Number 5,734,820. All Third Party Software names are trademarks or registered trademarks of their respective companies. All other brand or product names are trademarks or registered trademarks of their respective companies.

Sterling Commerce, Inc.
4600 Lakehurst Court Dublin, OH 43016-2000 *
614/793-7000

Contents

Chapter 1 About Connect:Direct Secure+ Option 9

Security Concepts	9
Security Protocols	10
Transport Layer Security Protocol and Secure Sockets Layer Protocol	10
Station-to-Station Protocol	11
Secure+ Option Tools	11
Administration Tool	11
Using the Native ISPF User Interface	12
Using the GUI Interface	13
Secure+ Parameter File	13
Access File	14
Sterling Control Center	14
Before You Begin	15
Identify Expert Security Administrator	15
Allocate Connect:Direct ISPF Libraries in TSO	15
Assess Security Requirements of Trading Partners	15
Plan Your Implementation of Connect:Direct Secure+ Option	16
Complete the Worksheets	16
Connect:Direct Secure+ Option Documentation	16
About This Guide	16
Task Overview	16

Chapter 2 Plan Your Implementation of the SSL or TLS Protocol 19

Transport Layer Security Protocol and Secure Sockets Layer Protocol	19
Summary of Processing Using the TLS or SSL Protocol	21
Secure+ Data Exchange	21
Authentication	21
Send/Receive Customer Data	22
Connect:Direct Access to System Resources for SSL or TLS	22
Self-Signed and CA-Signed Certificates	24
Terminology and Security Applications for SSL and TLS Certificates	25
General Requirements for Certificates	28
Application-Specific Requirements	28
Obtain Server Certificates and Set Up Connect:Direct for Certificates	28
Obtain a Certificate	28
Set Up Connect:Direct to Use Certificates	29

Chapter 3 Plan Your Implementation of the STS Protocol 31

Station-to-Station Protocol	31
STS Data Security	31
Encryption Options	32
Summary of Processing Using the STS Protocol	32
Authentication	33
Sending Customer Data	33
Receiving Customer Data	34
Merging Secure+ Option Settings Using the STS Protocol	34
Digital Signature	34
Algorithm for Encrypting Control Blocks	35
Data Encryption	35
Override STS Functions in a Connect:Direct Process	35
Key Management for the STS Protocol	35
Exchange Public Keys Using Autoupdate	36
Key Update Frequency	36
Import Key File Management	36

Chapter 4 Using the Secure+ Admin Tool and Populating the Parameter File 37

Start the Secure+ Option Administration Tool	37
About the Secure+ Option Admin Tool	38
Protocol-Specific Parameters and Panels	39
Navigate the Secure+ Admin Tool	43
Secure+ Admin Tool Help	44
Ways to Populate the Parameter File and Configure Nodes	44
Decide How to Create the Secure+ Parameter File	44
Decide How to Configure Nodes	45
Populate the Secure+ Parameter File Using Quick Start	47
Create the Secure+ Parameter File Manually	49

Chapter 5 Create the Parameter File Manually for the SSL or TLS Protocol 51

Configuration Guidelines	51
Add the Local Node Record to the Parameter File Manually for the SSL or TLS Protocol	52
Add a Remote Node Record to the Parameter File Manually for the SSL or TLS Protocol	60

Chapter 6 Create the Parameter File Manually for the STS Protocol 67

Configuration Guidelines	67
Add the Local Node Record to the Parameter File Manually for the STS Protocol . .	68
Add a Remote Node Record to the Parameter File Manually for the STS Protocol .	74

Chapter 7 Additional Configuration Options	83
Add a Remote Node Record for the EA Server	83
Establishing Secure TCP API Connections to a Secure+ Option-Enabled Server . .	86
Implementing Strong Password Encryption	90
Disabling Strong Password Encryption	91
Troubleshooting Possible SPE Problems	92
Chapter 8 Configure the Local Node Record Imported from the Network Map	95
Configuration Guidelines	95
Configure the Local Node Record for the SSL or TLS Protocol	96
Configure the Local Node Record for the STS Protocol	103
Chapter 9 Configure Remote Node Records Imported from the Network Map	109
Configuration Guidelines	110
Configure a Remote Node Record for the SSL or TLS Protocol	110
Configure a Remote Node Record for the STS Protocol	114
Disable Secure+ Option in a Remote Node Record	119
Chapter 10 Manage Keys for the STS Protocol	123
Initial Exchange of STS Keys	123
Export STS Keys	124
Import STS Keys from a File	126
Import STS Keys Manually	129
Chapter 11 Enable and Validate Secure+ Operation	131
Save and Submit the Secure+ Parameter File	131
Prepare Connect:Direct for Secure+ Option Operations	133
Save the Parameter File After the Initial Setup	133
Validate and Test Connections by Session	134
Chapter 12 Override Settings in Connect:Direct Processes	135
Overriding Secure+ Defaults in the PROCESS Statement	136
Overriding Secure+ Defaults in the COPY Statement	136
Examples of Overriding Security Settings	137
Default is Secure Sessions	137
Default is Non-Secure Sessions	138
Default Cipher Suite is TLS_RSA_WITH_RC4_128_MD5	138
Replace Single Default Cipher Suite with Cipher List	138
Default Encrypts Both Control Block Information and Data Being Sent	139
Overriding Secure+ Option Settings in an STS Protocol Environment	139

Chapter 13 Maintain Secure+ Option	141
Maintaining the Secure+ Parameter File	141
Open a Secure+ Parameter File	143
View Information about the Secure+ Parameter File	143
Save Changes to Node Records Using the Save Active Option.	145
Resecure the Secure+ Parameter File and Access File	145
Maintaining Node Records	146
Disable Secure+ Option	147
Change the Cipher Suites.	147
Change the Encryption Algorithm Names.	148
Insert a Node.	149
Delete a Secure+ Option Remote Node Record.	149
Modify STS Keys.	150
Update Keys in Node Records Configured for the STS Protocol	150
Reset Keys in Remote Node Records Configured for STS.	151
 Chapter 14 Secure+ Option Statistics	 153
SSL or TLS Statistics Record	153
SSL or TLS Extended Option Statistics Record	154
STS Statistics Record	156
STS Extended Option Statistics Records	157
Copy Termination (CT) Record	160
 Chapter 15 Troubleshooting	 161
 Appendix A Definitions of Certificate Parameters	 169
Parameter Definitions for Certificates Generated with the RACF Application	169
Parameter Definitions for Certificates Generated with the GSKKYMANN Utility	171
Parameter Definitions for Certificates Generated with the CA-ACF2 Application	173
Parameter Definitions for Certificates Generated with the CA-Top Secret Application	175
 Appendix B Configuration Worksheets	 179
Local Node Security Feature Definition Worksheet.	180
Remote Node Security Feature Definition Worksheet.	181
.EASERVER Node Security Feature Definition Worksheet.	183
.CLIENT Node Security Feature Definition Worksheet	184

Appendix C Test Secure+ Option with the STS Protocol **185**

Task Summary	185
Access the Secure+ Option Admin Tool	186
Define Secure+ Option for Node A	186
Create the Secure+ Option Local Node Record and Keys for Node A	186
Create the Secure+ Option Remote Node Record and Keys for Node B	190
Export Node A's Public Keys	194
Save the Secure+ Parameter File for Node A	195
Define Secure+ Option for Node B	196
Create the Secure+ Option Local Node Record and Keys for Node B	196
Create the Secure+ Option Remote Node Record and Keys for Node A	197
Export the Public Keys of Node B	199
Import the Public Keys from Node A	199
Save the Secure+ Parameter File for Node B	200
Import the Public Keys of Node B to Node A	201
Save the Secure+ Parameter File for Node A	201
Update Connect:Direct Network Maps for Node A and Node B	202
Modify Connect:Direct Initialization Parameters	203
Restart Connect:Direct	203
Verify Secure+ Option is Enabled	203
Exchange Data and Compare Results	203

Appendix D Configure for a Secure Connection between z/OS and OpenVMS Nodes **205**

Defining Records in the z/OS Secure+ Parameter File	205
Defining the z/OS Remote Node Record in the OpenVMS Secure+ Parameter File	207

Glossary **209**

Index **217**

About Connect:Direct Secure+ Option

Connect:Direct Secure+ Option provides enhanced security for Connect:Direct. It uses cryptography to secure data transmission. You select the security protocol to use with the Secure+ product. Connect:Direct Secure+ Option for z/OS supports IBM System Modification Program Extended (SMP/E) and non-SMP/E installations. For optimum performance, use an SMP/E installation.

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 distributed, 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.

Connect:Direct Secure+ Option enables you to implement multiple layers of security. You can select 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). Depending on the security needs of your environment, you can also validate certificates using the Sterling External Authentication Server application.

Connect:Direct also allows you to implement security and encryption to the degree appropriate for your environment. For example, if your company has a universal policy that you want to enforce, you can elect to encrypt all files at all times. To provide flexibility, you can allow a trading partner to override your security settings by specifying any of the following conditions:

- ◆ Turning Connect:Direct Secure+ Option on or off for a particular session
- ◆ Specifying one or more ciphers to use for encryption instead of the default cipher suites
- ◆ Encrypting only the control block information contained in Function Management Headers (FMHs), such as a user ID, password, and filename, instead of the files being transferred if performance is a factor.

Security Protocols

Before you configure Connect:Direct Secure+ Option, you must determine the protocol that you and your trading partners will use to secure communications sessions. For planning information, see Chapter 2, *Plan Your Implementation of the SSL or TLS Protocol*, and Chapter 3, *Plan Your Implementation of the STS Protocol*.

Transport Layer Security Protocol and Secure Sockets Layer Protocol

The Transport Layer Security protocol (TLS) and the Secure Sockets Layer (SSL) protocol use certificates to exchange a session key between the node that initiates the data transfer process (the primary node, or PNODE) and the other node that is part of the communications session (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. Certificates can be self-issued or issued by a certificate authority (see *Self-Signed and CA-Signed Certificates* on page 24 for details). When a CA receives an application for a certificate, the CA validates the applicant's identity, creates a certificate, and then signs the certificate. A certificate authority (CA) issues and revokes CA-issued certificates. Self-signed certificates are created and issued by the owner of the certificate, who must export the certificate in order to create a trusted root for the certificate and supply the trusted root of the self-signed certificate to the partner in a connection.

The Sterling External Authentication Server application enables you to validate certificates that are passed during an SSL or TLS session. Using the Sterling External Authentication Server application, you can configure certificate chain validation, including the option to validate certificates against one or more Certificate Revocation Lists (CRLs) that are stored on an LDAP server. You can also configure the Sterling External Authentication Server application to return attributes associated with the incoming certificate, such as group information, that are stored on an LDAP server. See *Sterling External Authentication Server Implementation Guide* for installation information.

To use the Sterling External Authentication Server application, you configure your application to connect to the host name and port where the Sterling External Authentication Server application resides and specify a certificate validation definition. See the instructions for creating the Secure+

parameter file manually or using the network map for the TLS or SSL protocols for instructions to create the remote node record for the Sterling External Authentication Server application (.EASERVER).

Station-to-Station Protocol

The Station-to-Station (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. When you use the STS protocol, you are responsible for generating and managing authentication and signature public keys and exchanging these keys with your trading partners.

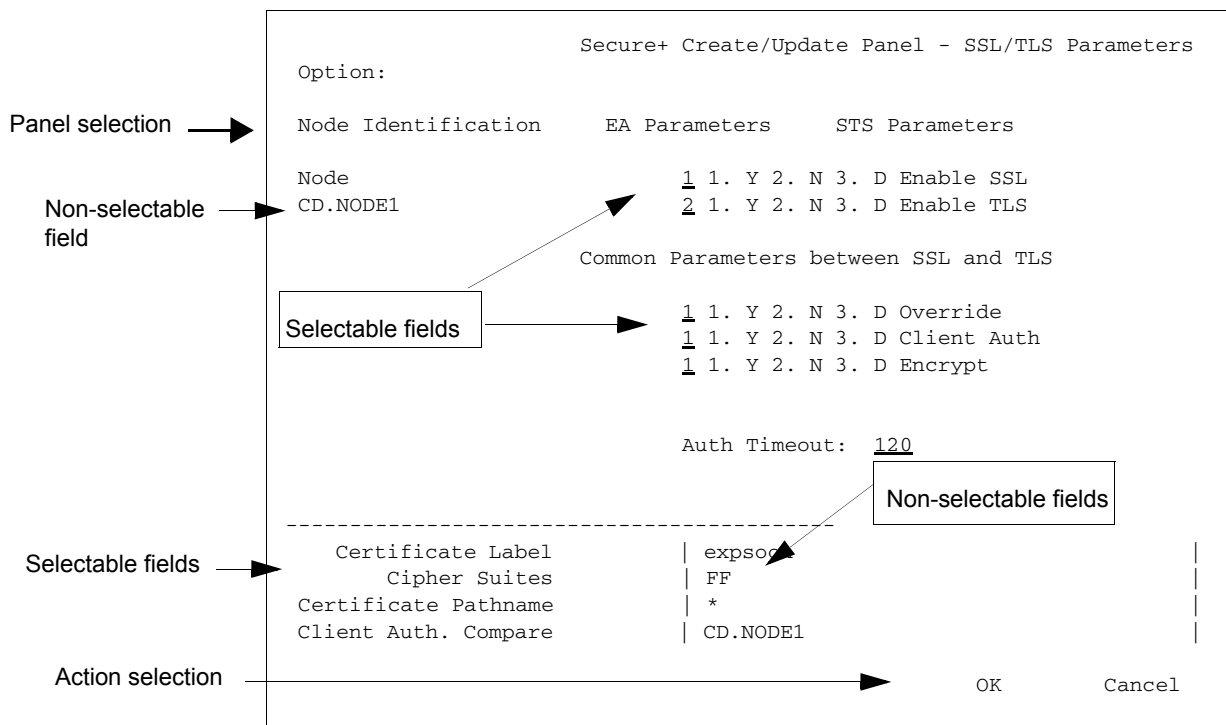
Secure+ Option Tools

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

Administration Tool

The Secure+ Admin Tool enables you to configure and maintain the Secure+ environment. The Admin Tool is the only interface for creating and maintaining the Secure+ parameter file. Other operating system utilities and editing tools do not work.

Two interface modes are available for the Admin Tool: native ISPF or graphical user interface (GUI). Both of these modes are driven by ISPF, so the screen content and functionality are identical, but the elements of the interfaces are different. The following sample illustrates the native ISPF interface display of the Create/Update Panel for SSL/TLS parameters. If you use the native ISPF interface, you can change the ISPF settings (Option 0) for the action bar choices and point-and-shoot fields. Changing these settings to fit your personal preferences can enhance operation and navigation in the Secure+ Option Admin Tool.



Using the Native ISPF User Interface

The Native ISPF User Interface uses the standard mouse method of pointing and clicking to position the cursor to select an item, or to access a different panel. You can also tab to move the cursor from field to field within a panel.

Panels that make up the Secure+ Admin tool can contain the following components:

- ◆ The Panel Selection bar lists the other panels you can access from the current panel. These panels are listed from left to right in the typical order you would enter information. For example, the first panel listed on the sample panel, Node Identification, contains basic information about the node that already exists in the network map, such as its node name and communication information (TCP/IP address). The Secure+ Create/Update Panel - SSL/TLS Parameters panel is the next panel used to enter protocol information for communicating with the current node, followed by the EA Parameter panel if External Authentication is implemented, and lastly, the STS Parameters panel used to disable that protocol.
- ◆ A selectable field is one in which you can either enter new information or edit existing information. On the sample panel, the fields you can enter information are those related to enabling SSL or TLS and the common parameters. You enter information in the underlined area next to the selectable field, such as the underlined value of 120 seconds next to the Auth Timeout field.
- ◆ The other selectable fields on the Secure+ Create/Update Panel - SSL/TLS Parameters panel take you to different panels where you can continue entering information related to the enabled protocol. For example, to choose cipher suites, you would select Cipher Suites by

moving the cursor to that label and pressing **Enter**. A new panel displays allowing you to select and prioritize the ciphers you want to use.

- ◆ Non-selectable fields contain reference information that is display-only, such as the node name, and existing certificate and cipher information.
- ◆ Action selection allows you to accept and save the information entered on the panel in the Secure+ parameter file or cancel. After you select an action, the panel which was displayed prior to the current panel, is redisplayed.

For more information on navigation and help using the panels that make up the Secure+ Admin tool, see page 43.

Using the GUI Interface

The GUI interface mode display uses command buttons, option buttons, and text boxes. To use the GUI mode, you must download an IBM workstation agent and change your ISPF settings (Option 0) to set up and use the workstation agent. See the IBM z/OS reference manuals for more information about downloading and using an IBM workstation agent. The following GUI interface sample illustrates the view when you customize Secure+ Option to display all parameters in a single panel.

Secure+ Parameter File

The Secure+ parameter file contains information that determines the protocol and encryption method used during security-enabled Connect:Direct operations. To configure Secure+, each site

must have a parameter file that contains one local node record and a remote node record for each trading partner who uses Secure+ to perform a secure connection. The local node record defines the most commonly used security and protocol settings at the site and can be used as a default for one or more remote node records. Each remote node record defines the specific security and protocol used by a trading partner.

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

Passwords are protected in the TCQ and AUTH files by encrypting them with Connect:Direct's proprietary "Polyalphabetic Substitution Cipher" which is a weak encryption. A stronger encryption algorithm, TDESCBC112, can be used if you add a .PASSWORD record to the Secure+ parameter file. After you create this record, enable the Strong Password Encryption (SPE) feature, and restart Connect:Direct, SPE protects Connect:Direct passwords stored in the TCQ and AUTH files with the stronger algorithm. For more information on using this feature, refer to *Implementing Strong Password Encryption* on page 90.

Access File

The Secure+ access file is generated automatically when you create the Secure+ parameter file for the first time. You type a passphrase when you first initialize Secure+. This passphrase is used to generate the keys necessary to encrypt and decrypt the entries in the Secure+ parameter file. The passphrase itself is not retained.

Your Secure+ administrator must secure the access file. This requires full create and update capability. The Connect:Direct server must have read authority. To maintain a secure access file, the general user community should not have access permission.

This file can be secured with any available file access restriction tools. Availability of the access file to unauthorized personnel can compromise the security of data exchange.

Sterling Control Center

Once you have created your Secure+ parameter file, you can use Sterling Control Center to perform the following functions when implementing the SSL or TLS protocol:

- ◆ Create and update a remote node
- ◆ Update the local node
- ◆ Add and update certificates
- ◆ Create an alias node
- ◆ Select cipher suites
- ◆ Delete a remote node

To perform these functions, you must have administration authority and the privileges to read and write to the Secure+ parameter file. For more information, see *Customizing Levels of Connect:Direct Functional Authority* in the *Connect:Direct for z/OS Administration Guide*. For more information on how to perform these functions, see the documentation for Sterling Control Center.

Before You Begin

Before you configure Connect:Direct Secure+ Option on the z/OS operating system, ensure that you complete the following tasks.

Identify Expert Security Administrator

The instructions and information provided to assist you in implementing the SSL/TLS protocol assume that you have an expert z/OS security administrator who is familiar with the terms associated with digital certificates and has experience using the tools required to generate and manage certificates, including:

- ◆ UNIX System Services
- ◆ IBM ICSF application and Crypto Hardware device
- ◆ System security applications, for example, gskkyman, RACF, CA-Top Secret, or CA-ACF2
- ◆ Security terminology associated with digital certificates (see *Terminology and Security Applications for SSL and TLS Certificates* on page 25)
- ◆ Working knowledge of the Connect:Direct application and its environment

Allocate Connect:Direct ISPF Libraries in TSO

To ensure that you can perform Secure+ parameter file functions and generate the SAVE AS JCL for the Secure+ parameter file, you must allocate the same release of the following Connect:Direct ISPF libraries in your TSO session before you try to perform Secure+ parameter file functions and generate and submit the Save As JCL as described in Chapter 11, *Enable and Validate Secure+ Operation* or the Save Active JCL as described in Chapter 13, *Maintain Secure+ Option*:

- ◆ ISPCLIB (must be allocated as SYSPROC)
- ◆ ISPLLIB
- ◆ ISPPLIB
- ◆ ISPSLIB
- ◆ ISPMLIB

If these required libraries have not been allocated, or have been allocated incorrectly, when you perform the save and submit procedure, the JCL for the SAVE AS job is not generated, and you have to repeat the configuration tasks. For more information on the required libraries and how to allocate them, see the *Connect:Direct for z/OS Installation Guide*.

Assess Security Requirements of Trading Partners

Security planning is a collaborative effort between you and your trading partners. You must know the expectations of your trading partners and plan your security implementation to meet these requirements. Consider the following guidelines for configuring communications sessions using the SSL or TLS protocol:

- ◆ You must acquire the certificates before you configure Secure+ Option.
- ◆ Determine whether you and your trading partner will use self-signed certificates or certificates signed by a Certificate Authority.

- ◆ Determine whether to use client authentication.
- ◆ Using the Sterling External Authentication Server application in conjunction with Secure+ Option to validate the other node's certificate for a secure session requires the following:
 - ◆ Using the TLS or SSL protocol for connections to the EA server
 - ◆ Enabling client authentication in remote node records so that the SNODE can validate the PNODE certificate
 - ◆ Exchanging certificates between the Connect:Direct for z/OS and the Sterling External Authentication Server node

Plan Your Implementation of Connect:Direct Secure+ Option

After you have identified your security administrator and determined the security requirements of your trading partners, review Chapter 2, *Plan Your Implementation of the SSL or TLS Protocol*, or Chapter 3, *Plan Your Implementation of the STS Protocol*, for protocol-specific configuration information.

Complete the Worksheets

Before you configure Connect:Direct Secure+ Option for z/OS, complete the worksheets in Appendix B, *Configuration Worksheets*. Use this information to configure the local and remote nodes to use Connect:Direct Secure+ Option for z/OS.

Connect:Direct Secure+ Option Documentation

See the *Connect:Direct for z/OS Release Notes* for a complete list of the product documentation.

About This Guide

The *Connect:Direct Secure+ Option for z/OS Implementation Guide* describes how to implement peer-to-peer security into Connect:Direct operations with Secure+. This document includes information to plan, configure, and use Secure+. The *Connect:Direct Secure+ Option for z/OS Implementation Guide* is for programmers and network operations staff who install and maintain Secure+.

This guide assumes knowledge of the following:

- ◆ Connect:Direct system, including its applications, network, and environment
- ◆ Security policies and applications used in your environment

Task Overview

The following table guides you to the information required to perform Secure+ tasks.

Task	Reference
Understanding Secure+ and assessing your security requirements	Chapter 1, <i>About Connect:Direct Secure+ Option</i>
Planning to use the TLS or SSL protocol	Chapter 2, <i>Plan Your Implementation of the SSL or TLS Protocol</i> Appendix A, <i>Definitions of Certificate Parameters</i>
Planning to use the STS protocol	Chapter 3, <i>Plan Your Implementation of the STS Protocol</i>
Navigating the Secure+ Admin Tool and populating the parameter file	Chapter 4, <i>Using the Secure+ Admin Tool and Populating the Parameter File</i> Appendix D, <i>Configure for a Secure Connection between z/OS and OpenVMS Nodes</i>
Setting up local and remote node records for the SSL or TLS protocol	Chapter 5, <i>Create the Parameter File Manually for the SSL or TLS Protocol</i> Chapter 8, <i>Configure the Local Node Record Imported from the Network Map</i> Chapter 9, <i>Configure Remote Node Records Imported from the Network Map</i> Appendix B, <i>Configuration Worksheets</i> Appendix D, <i>Configure for a Secure Connection between z/OS and OpenVMS Nodes</i>
Setting up and testing local and remote node records for the STS protocol	Chapter 6, <i>Create the Parameter File Manually for the STS Protocol</i> Chapter 8, <i>Configure the Local Node Record Imported from the Network Map</i> Chapter 9, <i>Configure Remote Node Records Imported from the Network Map</i> Appendix B, <i>Configuration Worksheets</i> Appendix C, <i>Test Secure+ Option with the STS Protocol</i>
Configuring special-purpose remote node records to perform one of the following functions: <ul style="list-style-type: none"> ◆ Validate certificates using the Sterling Authentication Server application ◆ Block nonsecure TCP API connections ◆ Secure passwords at rest within the Connect:Direct TCQ and AUTH files. 	Chapter 7, <i>Additional Configuration Options</i>
Managing STS keys	Chapter 10, <i>Manage Keys for the STS Protocol</i>
Saving the parameter file the first time after creating it and preparing Connect:Direct for operation	Chapter 11, <i>Enable and Validate Secure+ Operation</i>

Task	Reference
Validating and testing connections by session	Chapter 11, <i>Enable and Validate Secure+ Operation</i>
Performing exception processing by overriding Secure+ settings in the PROCESS statement	Chapter 12, <i>Override Settings in Connect:Direct Processes</i>
Maintaining the Secure+ parameter file and individual remote nodes	Chapter 13, <i>Maintain Secure+ Option</i>
Viewing Secure+ statistics	Chapter 14, <i>Secure+ Option Statistics</i>
Understanding error messages and resolving errors	Chapter 15, <i>Troubleshooting</i>

Plan Your Implementation of the SSL or TLS Protocol

Before you configure Connect:Direct Secure+ Option, review the following concepts, requirements, terms, and tool descriptions to ensure that you have all the resources and information necessary to implement the Transport Layer Security (TLS) protocol or the Secure Sockets Layer (SSL) protocol.

Transport Layer Security Protocol and Secure Sockets Layer Protocol

The TLS and the SSL protocols use certificates to exchange a session key between the node that initiates the data transfer process (the primary node, or PNODE) and the other node that is part of the communications session (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. Certificates can be self-issued or issued by a certificate authority (see *Self-Signed and CA-Signed Certificates* on page 24 for details). When a certificate authority (CA) receives an application for a certificate, the CA validates the applicant's identity, creates a certificate, and then signs the certificate. You use the CA signature to authenticate CA-issued trading partner certificates. A certificate authority issues and revokes CA-issued certificates. Self-signed certificates are created and issued by the owner of the certificate, who must export the certificate in order to create a trusted root for the certificate and supply the trusted root of the self-signed certificate to the partner in a connection.

The TLS and SSL protocols provide three levels of security:

- ◆ During the first level of authentication called server authentication, the site initiating the session (PNODE) requests a certificate from its trading partner (SNODE), during the initial handshake. The SNODE returns its ID certificate (read from its key certificate file) and the PNODE authenticates it using one or more trusted root certificates stored in a trusted root certificate file (the name and location of which are specified in the remote node record for that specific trading partner in the PNODE's Secure+ parameter file). Root certificates are signed by a trusted source—either a public certificate authority, such as Thawte, or by the trading partner acting as its own CA. If the ID certificate from the SNODE cannot be validated using any root certificate found in the trusted certificate file, or if the root certificate has expired, the

PNODE terminates the session. Connect:Direct writes entries to the statistics logs of both nodes, and the session is aborted.

- ◆ The second level of authentication is optional and is called client authentication. If this option is enabled in the SNODE's Secure+ parameter file definition for the PNODE, the SNODE will request a certificate from the PNODE, and authenticate it using the information in its trusted root certificate file. If this authentication fails, the SNODE terminates the session and Connect:Direct writes information about the failure to the statistics logs of both nodes.

In order to perform this security check, 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 either the Certificate Authority (CA) who issued the key certificate or the entity that created the certificate, if it is self-signed.

- ◆ The third authentication level is also optional and consists of validating the PNODE's certificate common name. When the security administrator enables client authentication, they can also specify the common name (CN) contained in the PNODE's ID certificate. During client authentication, the SNODE compares the common name it has specified for the PNODE in its Secure+ parameter file with the common name contained in the certificate sent by the PNODE. If the compare fails, that is, the information is not identical, the SNODE terminates the session, and Connect:Direct writes information about the failure to the statistics logs of both nodes.

The SSL and TLS protocols provide data security in the following areas:

- ◆ Authentication—Certificates used in the SSL or TLS session are digitally signed by a CA through an established procedure to validate an applicant's identity or digitally signed by the certificate owner-issuer. The SSL or TLS protocol validates the digital signature of the certificate being used.
- ◆ Proof of data origin and data integrity validation—The certificate provides proof of origin of electronic transmission and encryption validates data integrity. Message digest (hashing) and encrypting the message digest ensure that the data is not altered.
- ◆ Data confidentiality—Cipher suites encrypt data and ensure that the data remains confidential. The sending node converts sensitive information to an unreadable format (encryption) before it is sent to the receiving node. The receiving node then converts the information back into a readable format (decryption).

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 that must be exchanged between nodes.

- ◆ TLS provides more specific alerts about problems with a session and documents when certain alerts are sent.

Summary of Processing Using the TLS or SSL Protocol

After you configure Secure+ Option, you are ready to exchange data securely with other security-enabled Connect:Direct nodes. Data is securely exchanged between two nodes using the protocol defined in the parameter file.

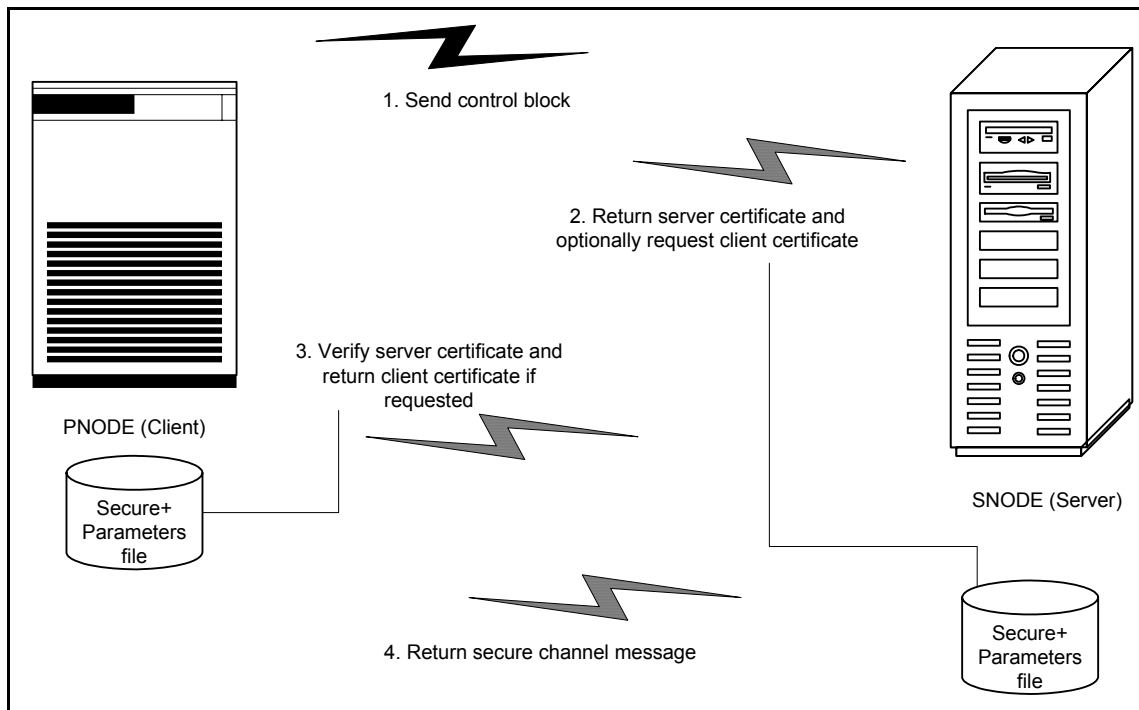
Note: You can implement the protocol you want to use for all data transfers or on a Process-by-Process basis. To specify a protocol each time you submit a Process, you must disable the protocol (but allow overrides) when you create the local and remote nodes in the Secure+ parameter file, and then specify it in the PROCESS statement using the SECURE parameter. For more information, see Chapter 12, *Override Settings in Connect:Direct Processes*.

Secure+ Data Exchange

Data exchange consists of two processes: authentication and sending/receiving data. The TLS or SSL protocol data exchange process is described in the following sections.

Authentication

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



The following steps occur during authentication:

1. The PNODE (client) sends a control block containing protocol (TLS or SSL) and cipher information to the SNODE (server). The SNODE confirms that it has a record defined in its Secure+ parameter file for the PNODE, and determines if a common cipher can be found and used for secure communication. Cipher suites are used to encrypt the data being sent between nodes. If the SNODE finds a record for the PNODE in its Secure+ parameter file and verifies it has a cipher defined in common with the PNODE, a common cipher is negotiated and the session continues.
2. The SNODE sends its ID certificate to the PNODE who confirms that it has a record defined in the Secure+ parameter file. Information for creating a public key is included. The PNODE verifies the ID certificate of the SNODE using the trusted root certificate file defined in its Secure+ parameter file, and generates a session key.
3. If client authentication is enabled on the SNODE, the SNODE requests an ID certificate from the PNODE. The PNODE sends its ID certificate defined in its Secure+ parameter file to the SNODE for verification against the trusted root certificate file specified in the SNODE's Secure+ parameter file. If a common name was also specified in the Secure+ parameter file for the PNODE, this name is used to verify the common name field of the PNODE's certificate.
4. The SNODE confirms that a secure environment is established and returns a secure channel message.

Send/Receive Customer Data

Once a Secure+ session has been established, all control blocks and customer data transmitted between the PNODE and SNODE are encrypted using the negotiated cipher.

Note: You can override certain security settings including what is encrypted during a session. If encrypting all data files is excessive in your environment, you can encrypt only the information necessary to establish a session and not the data files being transferred. For more information, see *Default Encrypts Both Control Block Information and Data Being Sent* on page 139.

Connect:Direct Access to System Resources for SSL or TLS

Before you can configure the Connect:Direct Secure+ Option records to use the SSL or TLS protocol, you must ensure that the Connect:Direct components have access to the resources listed in the following table.

Component	Access to Resource
Connect:Direct	UNIX System Services (USS), or POSIX environment, must be installed and set up for Connect:Direct access.

Component	Access to Resource
	<p>Access to the following APF-authorized IBM system libraries through the STEPLIB or LINKLST:</p> <ul style="list-style-type: none"> ◆ CEE.SCEERUN and CEE.SCEERUN2 (language environment) ◆ CBC.SCLBDLL (C/C++ environment) ◆ SYS1.SIEALNKE for IBM z/OS (System SSL Environment) <hr/> <p>For end-user server certificates with ICSF private key type:</p> <ul style="list-style-type: none"> ◆ The ICSF application must be running on the same environment as the Connect:Direct application. ◆ The Crypto Hardware device and the ICSF application must be running and accessible by the Connect:Direct application.
Connect:Direct User ID (under which DTF runs)	<p>Address space uses the maximum sockets (and other TCP/IP configurations) assigned by the UNIX System Services</p> <hr/> <p>OMVS access</p> <hr/> <p>A default UNIX directory</p> <hr/> <p>UPDATE authority to the BPX.SERVER facility</p>
SSL/TLS	<p>Access to key database or key ring as follows:</p> <ul style="list-style-type: none"> ◆ gskkyman key database ◆ RACF, CA-ACF2, or CA-Top Secret key ring <hr/> <p>Access to the following APF-authorized IBM system library through the STEPLIB or LINKLST:</p> <ul style="list-style-type: none"> ◆ SYS1.SIEALNKE for IBM z/OS (System SSL Environment) <hr/> <p>Permission to read Connect:Direct key ring that is created using RACDCERT, as follows:</p> <ul style="list-style-type: none"> ◆ Define the IRR.DIGTCERT.LIST and IRR.DIGTCERT.LISTRING resources with universal access of None. ◆ Grant the CONNECT:DIRECT User ID read access to the IRR.DIGTCERT.LIST and IRR.DIGTCERT.LISTRING resources in the FACILITY class. ◆ Activate the FACILITY general resource class. ◆ Refresh the FACILITY general resource class.
Connect:Direct User ID key database or key ring	<p>Verification of other certificates requires access to the trusted root certificate of either:</p> <ul style="list-style-type: none"> ◆ A trusted CA certificate ◆ Copy of a self-signed trusted certificate without private key

Component	Access to Resource
Secure+ Parameter file	Your node must have a remote node record in the parameter file of each of your trading partners that will use secure connections.

Self-Signed and CA-Signed Certificates

Determining the type of certificate to use for secure communications sessions and the method to generate the certificate is challenging. Self-signed certificates and digital certificates issued by certificate authorities offer advantages and disadvantages. You may also be required to use both types of certificates, depending on the security requirements of your trading partners. The following table compares the advantages and disadvantages of self-signed and CA-signed certificates.

Type of Certificate	Advantages	Disadvantages
Self-signed certificate	No cost	Requires you to distribute your certificate, minus its private key, to each trading partner in a secure manner.
	Easy to generate	Difficult to maintain; anytime the certificate is changed, it must be distributed to all clients.
	Self-validated	Not validated by a third-party entity
	Efficient for small number of trading partners	Inefficient for large number of trading partners
CA-signed certificate	Not required to store the public key of trading partner The public key signed by the CA is exchanged at SSL negotiation and authenticated against the CA's Trusted Root Key, which is stored in the Trusted Root directory and the USS key database or key ring of the Secure+ server.	Must be purchased from third-party vendor
	Tools used to generate certificates typically load the currently available CA certificates to the key database or key ring being generated, which means that you can connect your trading partner's certificates to the key ring to verify its trustworthiness.	

Type of Certificate	Advantages	Disadvantages
	Eliminates having to send your certificate to each trading partner	Trading partners must download digital CA-signed certificate used to verify the digital signature of trading partner public keys only if the CA certificate is not available
	Requires the remote client to store only the CA's digitally signed certificate (trusted key) in the Trusted Root directory	Must store the CA-signed certificate in the UNIX System Services (USS) key database or key ring and in the Trusted Root file
	No changes are required on the trading partner's system if you recreate the CA digitally signed certificate using the same CA	

Terminology and Security Applications for SSL and TLS Certificates

The following table defines the security terms associated with SSL and TLS certificates and applies them to communications sessions between a Connect:Direct PNODE (client) and SNODE (server). The terms are listed in alphabetical order.

Term	Definition
CA-Signed Certificate	Digital document issued by a certificate authority that binds a public key to the identity of the certificate owner, thereby enabling the certificate owner to be authenticated. An identity certificate issued by a CA is digitally signed with the private key of the certificate authority.
Certificate Authority (CA)	An organization that issues digitally-signed certificates. The certificate authority authenticates the certificate owner's identity and the services that the owner is authorized to use, issues new certificates, renews existing certificates, and revokes certificates belonging to users who are no longer authorized to use them. The CA digital signature is assurance that anybody that trusts the CA can also trust that the certificate that it signs is an accurate representation of the certificate owner.

Term	Definition
Certificate Signing Request (CSR)	Message sent from an applicant to a certificate authority in order to apply for a digital identity certificate. Before creating a CSR, the applicant first generates a key pair, keeping the private key secret. The CSR contains information identifying the applicant (such as a directory name in the case of an X.509 certificate), and the public key chosen by the applicant. The CSR may be accompanied by other credentials or proofs of identity required by the certificate authority, and the certificate authority may contact the applicant for further information.
Key Database	Database generated by the GSKKMAN utility for creating and managing public and private keys and certificates. Typically, the files in this database are password-protected to ensure that they are inaccessible to unauthorized users.
Key ring	File that contains public keys, private keys, trusted roots, and certificates. A key ring is a collection of certificates that identify a networking trust relationship (also called a trust policy) and are stored in a database. Key rings are associated with specific user IDs, which can have more than one key ring. Key rings enable you to share key rings across multiple servers.
Private Key	String of characters used as the private, "secret" part of a complementary public-private key pair. The asymmetric cipher of the private key is used to sign outgoing messages and decrypt data that is encrypted with its complementary public key. Data that is encrypted with a Public Key can only be decrypted using its complementary Private Key. The private key is never transmitted.
Public Key	String of characters used as the publicly distributed part of a complementary public-private key pair. The asymmetric cipher of the public key is used to confirm signatures on incoming messages and encrypt data for the session key that is exchanged between server and client during negotiation for an SSL/TLS session. The public key is part of the ID (public key) certificate. This information is stored in the key certificate file and read when authentication is performed.
Self-Signed Certificate	Digital document that is self-issued, that is, it is generated, digitally signed, and authenticated by its owner. Its authenticity is not validated by the digital signature and trusted key of a third-party certificate authority. To use self-signed certificates, you must exchange certificates with all your trading partners.
Session Key	Asymmetric cipher used by the client and server to encrypt data. It is generated by the SSL software.

The following table describes some system security applications available for generating certificates. Review the documentation for your security application for detailed instructions for generating certificates. See Appendix A, *Definitions of Certificate Parameters*, for more information on creating certificates using these tools.

Certificate Tool	Description
gskkyman	<p>IBM utility for creating and managing digital certificates and public and private keys stored in a key database. Files created using the gskkyman utility have the following default names:</p> <ul style="list-style-type: none"> ◆ key.kdb = private key file ◆ certreq.arm = Certificate Signing Request (CSR) file ◆ cert.arm = public key file <p>The gskkyman utility loads currently available CA certificates to the key database.</p>
Resource Access Control Facility (RACF)	<p>An IBM application that provides access control by identifying users to the system; verifying users of the system; authorizing access to protected resources; logging detected, unauthorized attempts to enter the system; and logging detected accesses to protected resources. The RACF utility can be used to create, store, and manage keys, digital self-signed or CA-signed certificates, and key rings. Because the RACF application can manage multiple key rings, certificates and key rings are added to the RACF database independently and then a certificate is associated with one or more key rings. For example, you can add the CA public key to your database and associate the certificates of your trading partners created by that CA with its public key.</p> <p>The RACF utility does not assign default names to the files you generate with it.</p>
Computer Associates Access Control Facility (CA-ACF2)	<p>Security application, similar to the RACF application, that enables you to authenticate users and to protect a variety of z/OS resources. You can generate, administer, and process certificate requests, export keys, and manage key rings.</p> <p>The CA-ACF2 application does not assign default names to the files you generate with it.</p>
CA-Top Secret	<p>Security application, similar to the RACF application, that protects your mainframe computer systems and data by controlling access to resources and enables you to generate, administer, and process certificate requests, export keys, and manage key rings.</p> <p>The CA-Top Secret application does not assign default names to the files you generate with it.</p>

General Requirements for Certificates

The certificate for the Connect:Direct Secure+ Option server defined in the local node record has the following general requirements:

- ◆ X.509 version 3 end-user server certificate that can interpret digital signatures and can encrypt and decrypt data
- ◆ Must be defined to the key database or key ring
- ◆ Must be stored in the key database or key ring
- ◆ Must have a private key
- ◆ Must be valid and not expired
- ◆ Must be signed by a CA or self-signed
- ◆ Must be set as default in the key database or key ring

Application-Specific Requirements

In addition to the general requirements for certificates, see Appendix A, *Definitions of Certificate Parameters*, for details on the minimum parameter definitions required for the security applications described in *Terminology and Security Applications for SSL and TLS Certificates* on page 25.

Obtain Server Certificates and Set Up Connect:Direct for Certificates

To use the SSL or the TLS protocol to perform a secure connection, you must obtain a server certificate and set up Connect:Direct to use certificates.

Note: An optional feature provides the ability to monitor certificates and issue automatic and on-demand warnings when certificates expire and are soon to expire within a specified number of days. Refer to *Checking the Validity of Certificates Used by Secure+* in the *Connect:Direct for z/OS Administration Guide*.

Obtain a Certificate

Certificates require key settings that define the type of security to implement at your site, including authentication, non-repudiation, data integrity, and data confidentiality, as described in *Security Concepts* on page 9. Although the security application that you use to create a digital certificate may use different terms to describe these security concepts (for example, digital signature, key encipherment, data encipherment, and non-repudiation), both self-signed certificates and certificate

requests sent to a certificate authority must designate all these key usage items to ensure that Secure+ Option can use the certificates to perform the intended security functions.

You can use the following methods to obtain an X.509 version 3 server certificate:

- ◆ Your registration authority can contract with a formal certificate authority (CA) to obtain a server certificate. When you obtain the server certificate, you then import this certificate into the IBM System SSL toolkit key database or key ring.
- ◆ Your registration authority can create a self-signed private and public key using one of the system security applications described in *Terminology and Security Applications for SSL and TLS Certificates* on page 25.
- ◆ Using one of the system security applications described in *Terminology and Security Applications for SSL and TLS Certificates* on page 25, your registration authority can generate a certificate signing request (CSR) for submission to third-party Certificate Authority to obtain a CA-signed public key. You forward this certificate to a certificate authority to be signed. When you receive the signed certificate, you import this certificate into the IBM System SSL key database or key ring. Refer to the IBM documentation *IBM Cryptographic Services System Secure Sockets Layer Programming Guide and Reference* for details.

Set Up Connect:Direct to Use Certificates

Before using the TLS or SSL protocol, you must set up Connect:Direct to use certificates.

To set up Connect:Direct to use certificates:

1. Ensure that the local Connect:Direct node to be configured for the TLS or SSL protocol has either a key ring or a key database on the z/OS image that contains its certificate.
2. Record the following information on your local node record worksheet for use when you configure the local node record in the Secure+ parameter file:
 - ◆ Name of the key ring or full file name of the key database
 - ◆ Label of the certificate in your key ring or key database
 - ◆ Password used when the key database was created

Note: Key rings do not use passwords.

3. If you are using a key database, issue the UNIX command **chmod 666** to ensure that Connect:Direct has permission to read from and write to the key database.
4. Issue the **INQUIRE APFILE** command to verify that the Connect:Direct license management key for Product Name (authorized abbreviated version) and the STS, TLS, and SSL protocols are installed. The license management file must contain feature bits **SECURE_PLUS** and **SECURE-SSL** to enable the STS, TLS, and SSL protocols.

Plan Your Implementation of the STS Protocol

The Station-to-Station (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.

Station-to-Station Protocol

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 is completed successfully, control blocks are exchanged in an encrypted format for the entire session.

STS 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

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)

Connect:Direct also gives you the ability to select what you want encrypted:

- ◆ Only the control block information contained in Function Management Headers (FMHs), such as a user ID, password, and filename
- ◆ Both control block information and the files being transferred

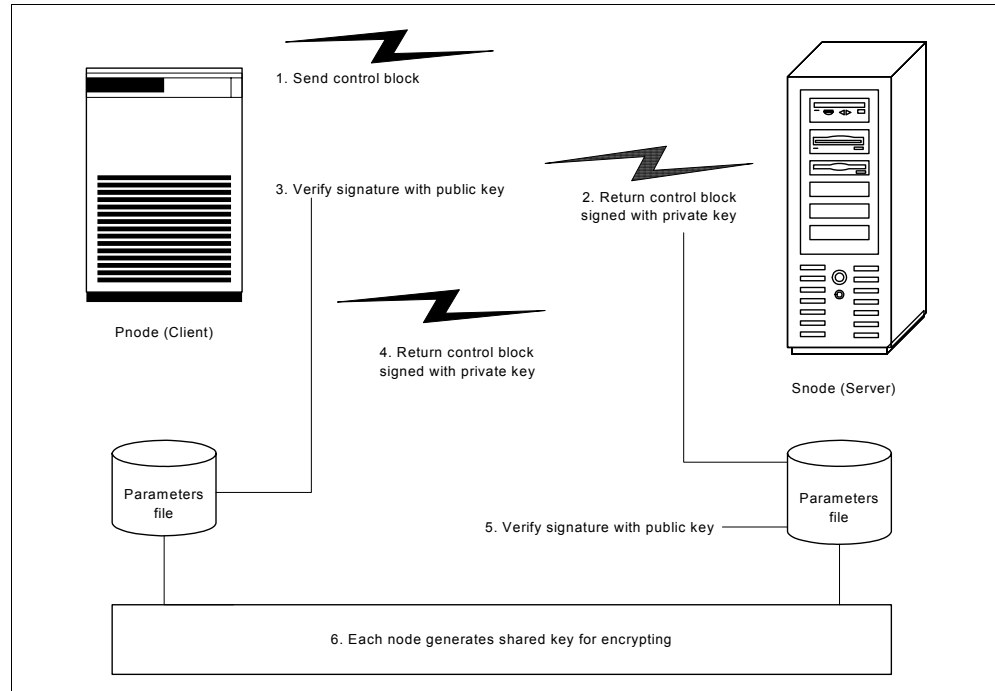
By restricting encryption to the control block information rather than both the preliminary FMH information exchanged during the handshake to set up the session and the actual data being transferred, CPU utilization will decrease dramatically.

Summary of Processing Using the STS Protocol

Data exchange consists of three steps: authentication, sending data, and receiving data. The primary node (PNODE) initiates the data transmission, and the secondary node (SNODE) 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 trading partner (SNODE). Information for creating an encryption key for the PNODE is included. The SNODE confirms that it has a record defined in the Secure+ parameter 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 and returns a control block signed with its private 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 PNODE 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, based on settings defined in Connect:Direct.
- ◆ 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.

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).

See *Digital Signature* on page 34 and *Algorithm for Encrypting Control Blocks* on page 35 to illustrate how the results of the merged PNODE and SNODE values is used to achieve the most secure connection.

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, the copy function fails.

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	First algorithm ID in the SNODE list that is in the PNODE list.
N	algorithm ID	SNODE algorithm ID if it is in the PNODE list.
Y	N Y algorithm ID	First algorithm ID in the PNODE list that is in the SNODE list.
algorithm ID	N Y algorithm ID	PNODE algorithm ID if it is in the SNODE list.

Override STS Functions in a Connect:Direct Process

When you configure a node to use the STS protocol, you can use the PROCESS or COPY statement in a Connect:Direct Process to override the settings in the parameter file, if override is enabled in the remote node record. Secure+ Option uses the most secure connection available. Therefore, if the remote node record enables digital signatures or encryption, the PNODE can not turn those options off by using the COPY statement override. For more information on overriding default settings in the parameter file, see Chapter 12, *Override Settings in Connect:Direct Processes*.

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 their node. In order to communicate with the trading partner, all four keys must be defined in the parameter file for both your configuration and the trading partner's configuration. Therefore, you and your trading partner must exchange these keys.

For the initial configuration, you manually exchange these keys. You export keys and send them to the trading partner. Then you import the keys you receive from the trading partner into the parameter file. After the initial exchange, you can automate the exchange of key information by defining the appropriate options in the remote node record.

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.

Exchange Public Keys Using Autoupdate

After you exchange keys with a trading partner, both partners can enable the automatic key update feature for easier key management. 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. Both you and your trading partner must enable automatic key update in order to use this feature. Enabling autoupdate eliminates much of the work that has to be performed by the Secure+ Option administrator for maintaining the keys.

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. Connect:Direct 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 directory for each remote node and store each key file separately in the associated directory.

See Chapter 10, *Manage Keys for the STS Protocol*, for instructions on importing and exporting keys.

Using the Secure+ Admin Tool and Populating the Parameter File

Use the following information to familiarize yourself with the Secure+ Option administration tool and to determine whether using Quickstart to populate the parameter file from the network map or populating the parameter file manually is the most efficient method to create your Secure+ parameter file.

Note: You can only use the Quick Start option the first time you create a parameter file, and if your Connect:Direct TCP/IP netmap entries use standard IPV4 addresses.

Start the Secure+ Option Administration Tool

Use the Administration Tool (Admin Tool) to set up and maintain a Secure+ Option operation.

To start the Secure+ Option Admin Tool:

1. To start the Secure+ Admin Tool, take one of the actions:
 - ◆ On the Connect:Direct Administrative Options Menu command line, type **SA** and press **Enter**.
 - ◆ On the Connect:Direct Administrative Options Menu command line, type **S** and press **Enter** to select the Execute Secure Plus Commands option, and then type **SA** on the command line and press **Enter**.
 - ◆ From the Connect:Direct Administrative Options Menu, select **Secure+** from the action bar and press **Enter**. Type **1** to select **Secure+ Admin Tool** and press **Enter**.

After the Secure+ Option Admin Tool is initialized, the **Secure+ Admin Tool: Main Screen** is displayed.

```

File  Edit  Key Management  Help
-----
                                Secure+ Admin Tool: Main Screen
Option ==>                                Scroll CSR

                                Table Line Commands are:

E Export pub. key          H View History          D Delete node
U Update node             I Insert node

                                Secure
LC Node Name              Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
***** BOTTOM OF DATA *****
    
```

2. To continue configuring Secure+ Option, refer to *Ways to Populate the Parameter File and Configure Nodes* on page 44.

About the Secure+ Option Admin Tool

When you start the Secure+ Option Admin Tool and open a parameter file, the panel displays all node records that are defined in the parameter file including a summary of the attributes for each node, unless you have filtered the records by node name using **Options** on the **Edit** menu. The following table describes the fields that are displayed on the **Secure+ Admin Tool Main Screen**, including a field description and valid values for each field, according to the protocol to which they apply.

Field Name	Field Description	Valid Values
All Protocols		
LC	Line Command. For more information, see <i>Maintaining Node Records</i> on page 146.	E = Export public key H = View History D = Delete node U = Update node I = Insert node
Node Name	Displays the node record name.	Node name
Override (set in local node record)	Displays the status of override. When override is enabled in the local node record, values set in the the remote node record override the values set in the local node record.	Y = enabled N = disabled
Override (set in remote node record)	Displays the status of override. Enabling override in a remote node record allows values specified in the PROCESS statement to override values set in the remote node record. For more information, see Chapter 12, <i>Override Settings in Connect:Direct Processes</i> .	Y = enabled N = disabled * = default to local node

Field Name	Field Description	Valid Values
Type	Displays the current record type.	L = local node record R = remote node record A = alias record Alias is valid only for remote records.
Secure		
1	Identifies the status of STS security.	Y=Yes
2	Indicates the status of SSL security.	N=No
3	Indicates the status of TLS security.	* = default to local node
C	Identifies the status of client authentication. Client authentication is valid only for the SSL and TLS protocols.	* is not a valid option in the local node record.
STS Protocol-Specific		
Override (set in remote node record for STS)	Displays the status of Override. Enabling Override in a remote node record that uses the STS protocol allows values specified in the COPY statement or the PROCESS statement to override values set in the remote node record. For more information, see Chapter 12, <i>Override Settings in Connect:Direct Processes</i> .	Y = enabled N = disabled * = default to local node
Encryption	Indicates if data encryption is enabled in the STS protocol.	Y = enabled N = disabled * = default to local node
Signature	Identifies if digital signature is enabled in the STS protocol.	Y = enabled N = disabled * = default to local node
Autoupd	Indicates if the option to automatically update STS key values during communications is enabled.	Y = enabled N = disabled * = default to local
SSL and TLS Protocol-Specific		
ExtAuth	Identifies whether external authentication is enabled for the node. Valid only for the SSL or TLS protocol. For more information, see <i>Add a Remote Node Record for the EA Server</i> on page 83.	Y = enabled N = disabled * = default to local node

Protocol-Specific Parameters and Panels

The default Create/Update panel display has changed. In versions prior to 5.0, the information was divided into four panels for External Authentication (EA), SSL, TLS, and STS. In version 5.0, SSL

and TLS have been combined so that the panel selection bar shows three choices, as illustrated in the following Node Identification panel.

```

Secure+ Create/Update Panel - Node Identification
Option:

EA Parameters          SSL/TLS Parameters    STS Parameters

Node
2 1. Local
   2. Remote

Alias
Names:                TCP Information:
                       IPaddr:
                       Port:

Import Remote Keys    Get Record           OK           Cancel
    
```

The Node Identification panel is the panel displayed when you create a record manually or when you want to display the Node Name and Type fields. The following tables list the Secure+ parameters according to the type of record (local or remote) to which they apply.

Node Identification Panel	Valid for Local Node Record?	Valid for Remote Node Record?
Node	Yes	Yes
Local/Remote	Yes	Yes
TCP Information/IP Address and Port	No	No
<p>Note: When you create the Secure+ parameter file from the NETMAP, the TCP Information field is populated automatically; however, data in the TCP Information field of the Secure+ remote record is not used to initiate Connect:Direct communications sessions. IP address and port number are acquired only from the NETMAP.</p>		
Alias Names	No	Yes

The following table describes the parameters displayed in the STS Parameters panel (see page 70 for a sample of this panel).

STS Parameters Panel	Valid for the Local Node Record?	Valid for the Remote Node Record?
Override	Yes. Valid for all protocols. Enable to allow turning security on or off in the PROCESS or COPY statement.	Yes. Enable to allow turning security on or off in the PROCESS or COPY statement. Consider the effects of settings for Signature and Encrypt when you set this parameter.
Autoupdt	Yes. Enable to allow automatic updates of keys used for the STS protocol.	Yes. Enable to allow automatic updates of keys used for the STS protocol.
Enable STS	Yes	Yes
Signature	Yes. Valid only for nodes that use the STS protocol.	Yes. Valid only for nodes that use the STS protocol.
Encrypt	Yes	Yes
Auth Timeout	Yes. Valid for all protocols.	Yes. Valid for all protocols.
Algorithm	Yes. Valid only for the STS protocol.	Yes. Valid only for the STS protocol.
Create/Reset Auth. Pubkey	Yes. Valid and required for all protocols.	Yes. Valid and required only for remote nodes that use the STS protocol.
Create/Reset Sig. Pubkey	Yes. Valid and required for all protocols.	Yes. Valid and required only for remote nodes that use the STS protocol.
Algorithm Names	Yes. Valid only for the STS protocol.	Yes. Valid only for the STS protocol.
Auth. Rmt.Key	Yes. Valid only for nodes that use the STS protocol.	Yes. Valid only for nodes that use the STS protocol.
Auth.Sig.Key	Yes. Valid only for nodes that use the STS protocol.	Yes. Valid only for nodes that use the STS protocol.
Import Remote Keys	Yes. Valid only for nodes that use the STS protocol.	Yes. Valid only for nodes that use the STS protocol.

Because the **Override**, **Encrypt**, and **Signature** parameters work together, review the following scenarios to determine the values to set for these parameters in a remote node record that uses the STS protocol.

Scenario	Setting for Override Parameter	Setting for Encrypt and Signature
All files must be encrypted and use signature.	Disable Override by setting to 2.	Enable Signature and Encrypt by setting to 1. Note: If you disable Override, you cannot disable security in the PROCESS statement.
A few files must be encrypted and use signature.	Enable Override by setting to 1.	Disable Signature and Encrypt by setting to 2. You can change these settings in the COPY statement Process so that the individual files use encryption and signature. See Chapter 12, <i>Override Settings in Connect:Direct Processes</i> . For additional information, you can also go to the Connect:Direct Processes Web site .

The following table describes the parameters displayed in the EA Parameters panel (see page 84 for a sample of this panel).

Note: If you have configured an .EASERVER remote node record, the following fields are populated but unavailable from any record except the .EASERVER record: External Auth Server Def, External Auth Server Address, and External Auth Server Port. The External Auth field can be modified from any record.

EA Parameters	Valid for Local Node Record?	Valid for Remote Node Record?
External Auth	Yes. Not a good idea to enable this parameter in the local node record.	Yes. Valid only for .EASERVER remote node record.
External Auth Server Def	No	Yes. Valid only for .EASERVER remote node record.
External Auth Server Address	No	Yes. Valid only for .EASERVER remote node record.
External Auth Server Port	No	Yes. Valid only for .EASERVER remote node record.

The following table describes the parameters displayed in the SSL/TLS Parameters panel (see page 12 for a sample of this panel). On this panel, you must enable only one protocol—either SSL or TLS—by entering a **1** next to the protocol you want to enable and a **2** next to the protocol to disable. Information common to both protocols is grouped under the Common Parameters heading followed by the certificate and cipher parameters related to the selected protocol selected.

SSL/TLS Parameters Panel	Valid for the Local Node?	Valid for the Remote Node?
Enable SSL	Yes.	Yes.
Enable TLS	Yes.	Yes.
Override	Yes. Valid for all protocols.	Yes. Valid for all protocols to allow Secure+ settings to be overridden in a PROCESS statement. See Chapter 12, <i>Override Settings in Connect:Direct Processes</i> .
Client Auth	Not a good idea to enable this parameter in the local node record.	Yes. Valid only for remote nodes that use the SSL or TLS protocol.
Encrypt	Yes.	Yes
Auth Timeout	Yes.	Yes.
Certificate Label	Yes. Valid only for the SSL or TLS protocol.	Yes. Valid only for the SSL or TLS protocol.
Cipher Suites	Yes. Valid only for the SSL or TLS protocol.	Yes. Valid only for the SSL or TLS protocol.
Certificate Pathname	Yes. Valid only for the SSL or TLS protocol.	Yes. Valid only for the SSL or TLS protocol. Note: The Certificate Pathname field is automatically set to "*" (Default to Local) in the remote node record. You are not allowed to update this field for a remote node.
Client Auth. Compare	No	Yes. Requires the certificate common name of the local node certificate when client authentication is enabled. Valid only for the SSL or TLS protocol.

Navigate the Secure+ Admin Tool

Use the following standard function keys to navigate the Admin Tool:

Key	Function
PF8	Move forward
PF7	Move backward
PF12	Back up to a previous panel
PF3	Exit
Enter	To select an option

Secure+ Admin Tool Help

You can access several types of Help information within the **Secure+ Option Admin Tool** as described in the following table:

Help	How to Access
General Help	From any Secure+ Option Admin Tool screen, select Help from the action bar and press Enter . Type 1 to select the general Help option.
Action Bar Help	Position the cursor on the action bar item and press PF1 , or position the cursor next to an option of an action bar item and press PF1 .
Screen and Panel-Level Help	Position the cursor in any uneditable part of the screen or panel and press PF1 .
Field-Level Help	Position the cursor in the editable part of a field and press PF1 .

Ways to Populate the Parameter File and Configure Nodes

You must configure Secure+ Option before you begin using it for secure communications. You create and save a Secure+ parameter file that contains a single local node record and a remote node record for every trading partner that uses Secure+ Option. The way you populate the parameter file depends on your environment. *Decide How to Create the Secure+ Parameter File* on page 44 and *Decide How to Configure Nodes* on page 45 describe two common scenarios and the most effective method of creating and populating the parameter file and configuring the local and remote nodes records for each scenario.

Decide How to Create the Secure+ Parameter File

The configuration procedures are based on the scenarios described in this section. Use the following table to help you decide how to create a Secure+ parameter file.

Scenario	Method to Create parameter file	Result
<ul style="list-style-type: none"> ◆ First time to create a parameter file. ◆ Large number of trading partners that use the same protocol. 	Use Quickstart to copy the network map file and save it as the Secure+ parameter file. See <i>Populate the Secure+ Parameter File Using Quick Start</i> on page 47.	<ul style="list-style-type: none"> ◆ File is created automatically with a single local node record and a record for each remote node in the network map that uses the TCP, UDT, or LU6.2 protocol. ◆ You must configure Secure+ Option for all remote node records, including trading partners that do not use Secure+ Option. ◆ Secure+ Option protocols are disabled for all records created from the network map. ◆ Establishes default settings for most parameters in the local node record.
<ul style="list-style-type: none"> ◆ First time to create a parameter file. ◆ Large number of trading partners. ◆ Few trading partners use Secure+ Option. 	Manually create a parameter file and add the local node record and remote node records. See <i>Create the Secure+ Parameter File Manually</i> on page 49.	<ul style="list-style-type: none"> ◆ You create the local node record and a record for each remote node that uses Secure+ Option. ◆ Reduces the number of records to configure. ◆ No default settings are established for parameters in the local node record. You must define all settings.

Decide How to Configure Nodes

After you create and populate the parameter file, you decide how to configure the local node record. The method that you use to configure the local node record then determines how you configure remote node records.

Use the following table to help you decide how to configure the local node:

Scenario	How to Configure Node Records	Result
Most trading partners use the same protocol.	Enable the most commonly used protocol in the local node record. Depending on the protocol, see the relevant procedure in Chapter 8, <i>Configure the Local Node Record Imported from the Network Map</i> .	<ul style="list-style-type: none"> ◆ Enables the same protocol in all remote node records. ◆ You have to modify only the records for remote nodes that do not use the settings for the local node.

Scenario	How to Configure Node Records	Result
Most trading partners do not use Connect:Direct Secure+ Option.	<p>Disable the Secure+ Option protocols in the local node record and enable the Override parameter. Depending on the protocol, see one of the following procedures:</p> <ul style="list-style-type: none"> ◆ <i>Add the Local Node Record to the Parameter File Manually for the SSL or TLS Protocol</i> on page 52 ◆ <i>Add the Local Node Record to the Parameter File Manually for the STS Protocol</i> on page 68 <p>Configure remote node records only for those trading partners who use Secure+ Option.</p>	<ul style="list-style-type: none"> ◆ You define default settings for all protocols (TLS, SSL, STS) in the local node record so remote nodes can use default values. ◆ You configure only those remote node records that use Secure+ Option.
Trading partners need to disable or enable security for a session.	Set OVERRIDE=Y in both the local and remote node records in the parameter files of both trading partners.	Either trading partner can disable or enable security for a particular session by setting the SECURE keyword in a Process or Copy statement. See Chapter 12, <i>Override Settings in Connect:Direct Processes</i> for more information.
Some trading partners use Connect:Direct Secure+ Option and the Sterling External Authentication Server application.	<p>Disable external authentication in the local node record and enable the Override parameter.</p> <p>Create a .EASERVER remote record. See <i>Add a Remote Node Record for the EA Server</i> on page 83.</p>	<ul style="list-style-type: none"> ◆ You can enable external authentication only for those remote nodes that use it with Secure+ Option. ◆ You can verify certificates exchanged during an SSL or TLS session using the Sterling External Authentication Server application.
Nonsecure TCP API connections are not allowed to connect to a Connect:Direct for z/OS server.	Create a .CLIENT remote node record and disable override. See <i>Establishing Secure TCP API Connections to a Secure+ Option-Enabled Server</i> on page 86.	◆ Communications from nonsecure connections is not allowed.

To see a scenario for setting up a secure connection between a Connect:Direct for OpenVMS node and a Connect:Direct for z/OS node, see Appendix D, *Configure for a Secure Connection between z/OS and OpenVMS Nodes*. That appendix provides a concrete example for defining a remote node record in both a Connect:Direct for z/OS Secure+ parameter file and a Connect:Direct for OpenMVS Secure+ parameter file.

Populate the Secure+ Parameter File Using Quick Start

The Quick Start option enables you to create the parameter file by importing information from the Connect:Direct network map and requires that you configure all remote node records, including those of trading partners that do not use Secure+ Option.

Note: You can only use the Quick Start option the first time you create a parameter file, and if your Connect:Direct TCP/IP netmap entries use standard IPV4 addresses.

To configure only the nodes that use Secure+ Option, refer to *Create the Secure+ Parameter File Manually* on page 49.

To import node records to the Secure+ parameter file from the Connect:Direct network map:

1. With the **Secure+ Admin Tool Main Screen** open, select **File** on the action bar and press **Enter**.

```

File  Edit  Key Management  Help
+-----+-----+-----+-----+
| 1. New          |                               | Row 4 of 6 |
| 2. Open        | Secure+ Admin Tool: Main Screen |             |
| 3. Close       |                               | Scroll CSR |
| 4. Info...     |                               |             |
| *. Rekey       | Table Line Commands are:      |             |
| 6. Save Active |                               |             |
| 7. Save as...  | H View History                | D Delete node |
| 8. Unload      | I Insert node                  |             |
| 9. Exit        |                               |             |
+-----+-----+-----+-----+
Secure
LC Node Name      Type 123C  Override Encryption Signature ExtAuth Autoupd
+-----+-----+-----+-----+
***** BOTTOM OF DATA *****

```

2. Type **2** to select **Open** and press **Enter**.

```

Secure+ Admin Tool: File Selection

Enter file name for: INPUT SECURE PARM FILE

File
Name: $CD.SECURE.NETMAP                               Browse

File System Type:
1 1. MVS  2. HFS                                       Cancel

```

3. Type the Connect:Direct network map file name prefix or partial prefix followed by an asterisk (*) select **Browse**, and press **Enter**.

Note: You can also type the complete Connect:Direct network map file name and press **Enter**.

4. Type **S** next to the file name of the network map you want to use and press **Enter**.

```
Secure+ Admin Tool: File Selection                               Row 1 of 3
Option: _____ Scroll CSR
Enter "S" on the line of the file for for MVS.
LC Filename or Directory
S $CD.NETMAP
_ $CD.NETMAP.DATA
_ $CD.NETMAP.INDEX
***** Bottom of data *****
```

5. When the Quick Start prompt screen is displayed, select **Yes** and press **Enter**.

After a few seconds, the **Secure+ Admin Tool: Main Screen** displays nodes populated from the Connect:Direct network map:

```
File Edit Key Management Help
-----
Row 9 to 13 of 13
Q2A.ZOS.PARIS          Secure+ Admin Tool: Main Screen
Option ==>                               Scroll CSR

Table Line Commands are:
E Export pub. key      H View History        D Delete node
U Update node         I Insert node

Secure
LC Node Name          Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
Q2A.ZOS.PARIS        L    NNNN   Y      N      N      N      N
Q3A.ZOS.PARIS        R    ***N   N      *      *      *      *
Q3B.ZOS.PARIS        R    ***N   N      *      *      *      *
SOL36SP              R    ***N   N      *      *      *      *
W2S.4200.CDWOPS8     R    ***N   N      *      *      *      *
***** BOTTOM OF DATA *****
```

6. Update the local and remote node records using the following procedures:
 - ◆ Chapter 8, *Configure the Local Node Record Imported from the Network Map*
 - ◆ Chapter 9, *Configure Remote Node Records Imported from the Network Map*

Create the Secure+ Parameter File Manually

If you determine that populating the Secure+ parameter file manually is most efficient for your environment, refer to the following instructions for configuring the local and remote node records:

- ◆ Chapter 5, *Create the Parameter File Manually for the SSL or TLS Protocol*
- ◆ Chapter 6, *Create the Parameter File Manually for the STS Protocol*

Create the Parameter File Manually for the SSL or TLS Protocol

If you communicate with a large group of trading partners, but only a few trading partners use Secure+ Option, you can manually create and populate the parameter file by creating a single local node record and a remote node record for each trading partner that uses Secure+ Option. This method minimizes the number of remote node records to configure in the parameter file.

For instructions on additional configuration options, see:

- ◆ *Add a Remote Node Record for the EA Server* on page 83
- ◆ *Establishing Secure TCP API Connections to a Secure+ Option-Enabled Server* on page 86

To validate and test a Secure+ connection between two business partners, see *Validate and Test Connections by Session* on page 134.

Configuration Guidelines

When you use the manual method to populate the parameter file, you should disable all protocols and external authentication and allow override in the local node record. Review the table on page 45 to determine the configuration approach that best suits your needs, and use the following guidelines when you configure the local node record manually:

- ◆ Disable the Secure+ Option protocols (TLS, SSL, STS) in the local node record. Then configure each remote node record with the protocol used by that trading partner. To disable all protocols and the External Authentication Server application, you must change Default to Local Node settings in the following panels: SSL/TLS Parameters, EA Parameters, and STS Parameters. Allow overrides in the Local Node settings.
- ◆ Disable external authentication.
- ◆ Create keys for the STS protocol because this action also creates the key that encrypts the Secure+ parameter file.

- ◆ For all environments, you must define required settings in the local node record, including certificate information used with the TLS or SSL protocol. You can also define optional settings in the local node record and use them in all remote node records.
- ◆ Read all warning and error messages. You can continue configuring the environment without resolving warning messages, but you may be unable to perform secure communications. You must resolve all errors before saving the parameter file.
- ◆ To enable secure connections using Secure+ Option, you must complete the procedures in *Add the Local Node Record to the Parameter File Manually for the SSL or TLS Protocol* on page 52, *Add a Remote Node Record to the Parameter File Manually for the SSL or TLS Protocol* on page 60, and Chapter 11, *Enable and Validate Secure+ Operation*.
- ◆ Before you start the Secure+ Admin Tool to add your local and remote node records, verify that you have allocated the ISPF libraries in your TSO session that are required to save the Secure+ parameter file (see *Allocate Connect:Direct ISPF Libraries in TSO* on page 15 for details).

Add the Local Node Record to the Parameter File Manually for the SSL or TLS Protocol

When you perform this procedure, refer to the *Local Node Security Feature Definition Worksheet* on page 180 that you completed for the local node.

To add the local node record manually:

1. Select **Edit** from the **Secure+ Admin Tool Main Screen** and press **Enter** to display the **Edit** menu.

```

File  Edit  Key Management  Help
-----
                                Secure+ Admin Tool: Main Screen
Option ==>                                Scroll CSR

                                Table Line Commands are:

E Export pub. key      H View History      D Delete node
U Update node         I Insert node

                                Secure
LC Node Name          Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
***** BOTTOM OF DATA *****
    
```

2. On the **Edit** menu, select **1** to select **Create/Update Record** and press **Enter** to display the **Secure+ Create/Update - Node Identification** panel:

Add the Local Node Record to the Parameter File Manually for the SSL or TLS Protocol

```

Secure+ Create/Update Panel - Node Identification
Option:

EA Parameters          SSL/TLS Parameters      STS Parameters

Node
                        1 1. Local
                        2. Remote

Alias
Names:                 TCP Information:
                        IPAddr:
                        Port:

Import Remote Keys    Get Record          OK          Cancel
    
```

3. On the **Node Identification** panel:
 - a. Type a name for the local node in the **Node** field.
 - b. Type **1** next to the **Local** field.
 - c. Leave the **TCP Information** fields (**IP addr** and **Port**) blank because they do not apply to the local node record.
 - d. Leave the **Alias Names** field blank because it is not valid for the local node.
4. Select **SSL/TLS Parameters** from the panel selection bar and press **Enter**.

```

Secure+ Create/Update Panel - SSL/TLS Parameters
Option:

Node Identification    EA Parameters          STS Parameters

Node
SC.OKC.PARIS1         2 1. Y 2. N 3. D Enable SSL
                     2 1. Y 2. N 3. D Enable TLS

Common Parameters between SSL and TLS

1 1. Y 2. N 3. D Override
2 1. Y 2. N 3. D Client Auth
1 1. Y 2. N 3. D Encrypt

Auth Timeout: 120

Certificate Label     | * |
Cipher Suites         | FF |
Certificate Pathname  | * |
Client Auth. Compare |   |

OK          Cancel
    
```

5. In the **SSL/TLS Parameters** panel:
 - a. To disable the SSL protocol, type **2** beside the **Enable SSL** field.
 - b. To disable the TLS protocol, type **2** beside the **Enable TLS** field.
 - c. To enable the Override function, type **1** beside the **Override** field.
 - d. To disable the Client Authentication function, type **2** beside the **Client Auth** field.
 - e. To enable the default Encrypt function, type **1** beside the **Encrypt** field.
 - f. Modify the **Auth Timeout** value, if necessary, using the following table as a guide:

Field Name	Description	Valid Value
Auth Timeout	Identifies the maximum time, in seconds, that the system waits to receive Connect:Direct control blocks exchanged during the authentication protocol.	0=No timeout. 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. Set the value equal to or greater than the value set for the Connect:Direct TCP.TIMER initialization parameter. The default is 120 seconds.

6. Specify the certificate label:
 - a. Select the **Certificate Label** field and press **Enter**.
 - b. Press **F8** to move to the editable portion of the panel containing the label field.
 - c. This field is case sensitive; therefore, type the certificate label exactly as you defined it when you generated it using one of the security applications described in Appendix B and press **Enter**.

Add the Local Node Record to the Parameter File Manually for the SSL or TLS Protocol

```

Secure+ Create/Update Panel - SSL/TLS Parameters

Option:

Node Identification      EA Parameters      STS Parameters

Node                    2 1. Y 2. N 3. D Enable SSL
SC.OKC.PARIS1          2 1. Y 2. N 3. D Enable TLS

Common Parameters between SSL and TLS

1 1. Y 2. N 3. D Override
2 1. Y 2. N 3. D Client Auth
1 1. Y 2. N 3. D Encrypt

Auth Timeout: 120

Certificate Label      |-----|
Cipher Suites         | keylabel |
Certificate Pathname  | FF      |
Client Auth. Compare  | *       |
                       |-----|

OK                      Cancel
    
```

7. Identify where the certificate information is stored:
 - a. Select the **Certificate Pathname** field and press **Enter**.
 - b. Press **F8** to scroll to the editable portion of the panel containing the **Certificate Path** field.
 - c. Type the UNIX path name of the key database (.kdb) or the security system key ring name that contains all certificates referred to in the parameter file.

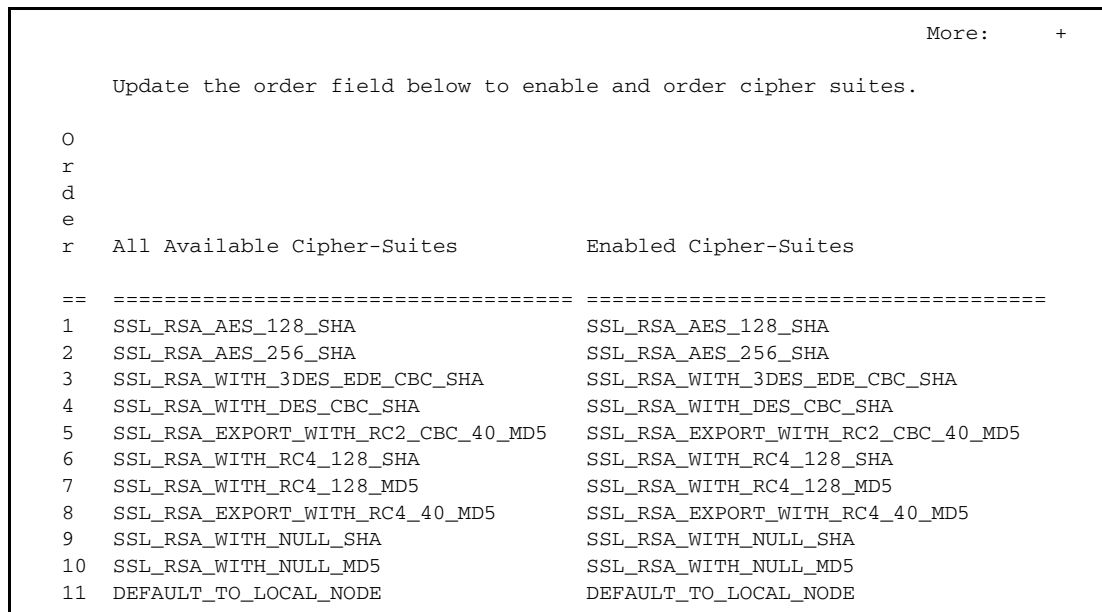
Note: This value is case sensitive. Ensure that you type it exactly as it appears in the certificate file. Refer to the information you recorded in *Local Node Security Feature Definition Worksheet* on page 180.

- d. If you are using a key database:
 - a. Press **F8** to scroll to the editable portion of the panel containing the password field.
 - b. Type the password used when the key database was created and press **Enter**.

Note: This value is case sensitive. Ensure that you type it exactly as it appears in the certificate file. Refer to the information you recorded in *Local Node Security Feature Definition Worksheet* on page 180.

Note: If you are using a key ring, leave the password field blank.

8. To enable and prioritize cipher suites:
 - a. Select the **Cipher Suites** field and press **Enter** to display the Update Cipher Suites panel.



- b. Type **1** by the cipher you want to enable and give the highest priority.
 - c. Type **2** by the cipher you want to enable and place second in priority.
 - d. Continue typing numbers next to the ciphers you want to enable, in order of priority.

The ciphers you enable appear in the order of priority in the **Enabled Cipher-Suites** list.

- e. Press **F3** when you have enabled and ordered all necessary ciphers.

Note: To identify the ciphers available, run a trace on the Connect:Direct system. Setting **debug=8C0000AE** in the initialization parameter file dynamically allocates DD R00000001. Available ciphers are listed in the trace DD. Turn global tracing off before you continue.

Note: When the SSL or TLS environment is correctly set up and the cipher suites selected, the textual names in the Update Cipher Suites panel are set based on the protocol in effect.

9. To disable the use of the External Authentication Server application:
 - a. Select **EA Parameters** from the panel selection bar and press **Enter**.
 - b. Type **2** in the **External Auth** field.

The remaining external authentication fields are unavailable because they are valid only for the .EASERVER remote node record.

Note: Values set for parameters that are displayed in multiple panels (Override, for example) are retained in a record after you set them the first time they are displayed.

```

Secure+ Create/Update Panel - EA Parameters
Option:
Node Identification      SSL/TLS Parameters      STS Parameters
Node                   1 1. Y 2. N 3. D Override
SC.OKC.PARIS1         1 2 1. Y 2. N 3. D External Auth

External Auth Server Def   THISISATEST
External Auth Server Address 10.20.201.2
External Auth Server Port   23234

                                OK          Cancel
    
```

10. Select **STS Parameters** and press **Enter**.

```

Secure+ Create/Update Panel - STS Parameters
Option:
Node Identification      EA Parameters      SSL/TLS Parameters
Node                   1 1. Y 2. N 3. D Override
SC.OKC.PARIS1         2 1. Y 2. N 3. D Autoupdt
                        2 1. Y 2. N 3. D Enable STS
                        2 1. Y 2. N 3. D Signature
                        2 1. Y 2. N 3. D Encrypt
                        Auth Timeout: 120
                        Algorithm      *

Create / Reset Auth. Prev. Keys           Expire Date MDY
Create / Reset Sig. Prev. Keys           Expire Date MDY

Create / Reset Auth. Pubkey | * |
Create / Reset Sig. Pubkey | * |
                        Algorithm Names | DESCBC56,TDESCBC112,IDEACBC128 |
                        Auth. Rmt. Key | 0000 |
                        Sig. Rmt. Key | 0000 |

                                < >

Import Remote Keys      Get Record          OK          Cancel
    
```

11. In the **STS Parameters** panel:
 - a. To disable the STS parameters, type **2** beside the following fields: **Autoupdt**, **Enable STS**, **Signature**, and **Encrypt**.
 - b. Type an asterisk (*) in the **Algorithm** field.
12. Generate the STS protocol authentication key, which is used to encrypt the parameter file:
 - a. Select **Create/Reset Auth. Pubkey** and press **Enter** to display the **Generate Seed** screen.

```

Secure+ Admin Tool: Generate Seed

2 1. Specify Value          Specify the seed value by typing it
                             into the text field.
   2. Sample Value         Generate a seed by processing text
                             entered from the keyboard.

Random Number
Seed:
    
```

- b. Press **Enter** to accept the default value of **2-Sample Value**.
- c. When the following screen is displayed, edit or add data on any line and press **PF3** to save the information. Changing data creates a unique key value.

```

File  Edit  Edit_Settings  Menu  Utilities  Compilers  Test  Help
-----
EDIT          SYS06254.T160411.RA000.SSCHR1.R0207298          Columns 00001 00072
Command ===>                                     Scroll ===> PAGE
024: This process cannot proceed if the data in the edit file is unchanged.
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 262144
000002
000003 134217728
000004
000005 32
000006
000007 4
000008
000009 8589934592
000010
000011 131072
000012
000013 8388608
000014
000015 2097152
000016
000017 1073741824
    
```

- d. When the message *This is Loop 2 of 10* is displayed, type over data on any line and press **PF3** up to 10 times. This step is only necessary the first time you generate keys within the Secure+ Option Admin Tool for each parameter file. Repeating the process increases the randomness of keys.

- e. When the **Pass Phrase Generation** panel is displayed, type a string at least 32 characters long containing at least one uppercase character, one lowercase character, and one numeric value and press **Enter**.

Note: You do not need to remember the pass phrase.

```

Secure+ Create/Update Panel - STS Parameters
Option:
Node Identification      EA Parameters      SSL/TLS Parameters
Node
SC.OKC.PARIS1          1 1. Y 2. N 3. D Override
                       2 1. Y 2. N 3. D Autoupdt
                       2 1. Y 2. N 3. D Enable STS
Auth Timeout: 120      2 1. Y 2. N 3. D Signature
Algorithm *            2 1. Y 2. N 3. D Encrypt

Create / Reset Auth. Prev. Keys      Expire Date MDY
Create / Reset Sig. Prev. Keys      Expire Date MDY

Create / Reset Auth. Pubkey | 0203.093F.5D89.9024.5080.FE6D.7574.F55B. |
Create / Reset Sig. Pubkey | * |
Algorithm Names | DESCBC56,TDESCBC112,IDEACBC128 |
Auth. Rmt. Key | 0000 |
Sig. Rmt. Key | 0000 |

----- < > -----
Import Remote Keys      Get Record      OK      Cancel
    
```

When the **Secure+ Option Create/Update Panel - STS Parameters** displays the message *Seed generation complete*, your public key for strong authentication has been generated and the **Create/Reset Auth. Pubkey** field is populated, as shown in the preceding illustration.

13. Generate the signature key, which is part of the key pair used to encrypt the parameter file:
 - a. Select **Create/Reset Sig. Pubkey** and press **Enter**.
 - b. Press **Enter** to accept the default value (**2-Sample Value**).

```

Secure+ Create/Update Panel - STS Parameters
Option:
Node Identification      EA Parameters      SSL/TLS Parameters
Node
SC.OKC.PARIS1
Auth Timeout: 120
Algorithm *
Create / Reset Auth. Prev. Keys      Expire Date MDY
Create / Reset Sig. Prev. Keys      Expire Date MDY
----- < > -----
Create / Reset Auth. Pubkey | 0203.093F.5D89.9024.5080.FE6D.7574.F55B. |
Create / Reset Sig. Pubkey | 0201.C2DB.B318.1B91.3A11.7FD3.3553.37EA. |
Algorithm Names | DESCBC56,TDESCBC112,IDEACBC128 |
Auth. Rmt. Key | 0000 |
Sig. Rmt. Key | 0000 |
-----
Import Remote Keys      Get Record      OK      Cancel

```

When the **Secure+ Option Create/Update Panel - STS Parameters** displays the message *Seed generation complete*, your public key for digital signature has been generated and the **Create/Reset Sig. Pubkey** field is populated, as shown in the preceding illustration.

14. Select **OK** and press **Enter** to display the values for the local node record.
15. Read all warning and error messages. You can continue configuring the environment without resolving warning messages, but you must resolve all errors before you save the parameter file.
16. After you configure the local node record, you can save and submit the parameter file using the procedures in Chapter 11, *Enable and Validate Secure+ Operation*, but if you have not added a remote node record, connections are not secure.

Add a Remote Node Record to the Parameter File Manually for the SSL or TLS Protocol

Refer to the *Remote Node Security Feature Definition Worksheet* on page 181 that you created for the remote node you are adding when you complete this procedure. The following procedure assumes that this remote node uses the SSL or TLS protocol and client authentication with Connect:Direct Secure+ Option unless you want to override the Secure+ Option parameter settings from the PROCESS statement. For more information, see Chapter 12, *Override Settings in Connect:Direct Processes*.

To add a remote node record manually for the SSL or TLS protocol:

1. Select **Edit** from the **Secure+ Admin Tool Main Screen** and press **Enter** to display the **Edit** menu. Settings for configured node records are displayed.

```

File  Edit  Key Management  Help
-----
                                         Row 1 of 1
                Secure+ Admin Tool: Main Screen
Option ==>                                         Scroll CSR

                Table Line Commands are:

E Export pub. key      H View History      D Delete node
U Update node          I Insert node

                Secure
LC Node Name          Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
MYLOCAL              L    NNNN   Y          N          N          N          N
***** BOTTOM OF DATA *****
    
```

2. On the **Edit** menu, type **1** to select **Create/Update Record** and press **Enter**.

Note: Values set for parameters that are displayed in multiple panels (Override, for example) are retained in a record after you set them the first time they are displayed.

```

Secure+ Create/Update Panel - Node Identification
Option:

EA Parameters          SSL/TLS Parameters      STS Parameters

Node

                1. Local
                2. Remote

Alias
Names:                TCP Information:
                    IPaddr:
                    Port:

Import Remote Keys    Get Record            OK            Cancel
    
```

3. On the **Node Identification** panel:
 - a. In the **Node** field, type the name for the remote node that corresponds to its name in the network map.
 - b. Type **2** next to the **Local/Remote** field.

- c. Leave the **TCP Information** fields (**IP addr** and **Port**) blank because Connect:Direct always obtains the IP address and port for a remote node from the network map.
 - d. In the **Alias Names** field, type any alternative name for this remote node that uses the same Secure+ Option parameters. This alias name must also exist as a valid remote node entry in the network map.
4. Select **SSL/TLS Parameters** from the panel selection bar and press **Enter**.

```

Secure+ Create/Update Panel - SSL/TLS Parameters
Option:
Node Identification      EA Parameters      STS Parameters
Node                    1 1. Y 2. N 3. D Enable SSL
REMOTE01                2 1. Y 2. N 3. D Enable TLS

Common Parameters between SSL and TLS

2 1. Y 2. N 3. D Override
1 1. Y 2. N 3. D Client Auth
2 1. Y 2. N 3. D Encrypt

Auth Timeout: 120

Certificate Label      | * |
Cipher Suites          | FF |
Certificate Pathname   | * |
Client Auth. Compare   |   |

OK                      Cancel
    
```

In the **SSL/TLS Parameters** panel:

- a. To implement SSL, take one of the following actions, depending on whether you want to use SSL for all data transfers or on a Process-by-Process basis:
 - Type **1** beside the **Enable SSL** field to enable the SSL protocol for this remote node.
 - Type **2** beside the **Enable SSL** field to disable the SSL protocol but enable it later in a **PROCESS** statement.
- b. To implement TLS, take one of the following actions, depending on whether you want to use TLS for all data transfers or on a Process-by-Process basis:
 - Type **1** beside the **Enable TLS** field to enable the TLS protocol for this remote node.
 - Type **2** beside the **Enable TLS** field to disable the TLS protocol but enable it later in a **PROCESS** statement.

Note: If you attempt to set both SSL and TLS to 1 (or Y for Enabled), a warning displays indicating that the Enable SSL setting was changed to 2 (disabled) because you can only enable one protocol at a time for a particular node.

- c. Take one of the following actions, depending on whether you want to use the Secure+ Option parameter settings override feature.

- To enable the Secure+ Option parameter settings override feature in the PROCESS or COPY statement, type **1** beside the **Override** field. For more information, see Chapter 12, *Override Settings in Connect:Direct Processes*.
 - To disable the Secure+ Option parameter settings override feature, type **2** beside the **Override** field.
- d. To enable client authentication:
- Type **1** beside the **Client Auth** field.
 - To have the common certificate name verified during the authentication process, select **Client Auth. Compare** and when the next panel displays, type the certificate common name of the local node certificate and press **Enter**. To not have the name verified, leave this field blank by not selecting the **Client Auth. Compare** field. If the common name is not entered, the client name verification process is not performed but client authentication is.

Note: This value is case-sensitive. Make sure that you type it exactly as it appears in the certificate file.

- e. Take one of the following actions, depending on what information you want to encrypt:
- Type **1** beside the **Encrypt** field to encrypt all information sent during the handshake to set up communication sessions and the actual files being transferred.
 - Type **2** beside the **Encrypt** field to encrypt only the control block information sent during the handshake to set up communication sessions and not the actual files being transferred.
 - Type **3** beside the **Encrypt** field to default to the local node's specification of **Encrypt**.
- f. To modify the value for the **Auth Timeout** field, use the following table as a guide:

Field	Description	Valid Values
Auth Timeout	Identifies the maximum time, in seconds, that the system waits to receive Connect:Direct control blocks exchanged during the authentication protocol.	0=No timeout. 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. Set the value equal to or greater than the value set for the Connect:Direct TCP.TIMER initialization parameter. The default is 120 seconds.

5. To change the list of ciphers enabled for this remote node record:
 - a. Select the **Cipher Suites** field and press **Enter** to display the Update Cipher Suites panel.
 - b. Type **1** by the cipher you want to enable and give the highest priority.
 - c. Continue typing numbers next to the ciphers you want to enable, in order of priority.
The ciphers you enable appear in the order of priority in the **Enabled Cipher-Suites** list.
 - d. Press **F3** when you have enabled and ordered all necessary ciphers.
6. To specify the certificate label:
 - a. Select the **Certificate Label** field and press **Enter**.
 - b. Press **F8** to move to the editable portion of the panel containing the label field.
 - c. This field is case sensitive; therefore, type the label of the certificate exactly as you defined it when you generated it using one of the security applications described in Appendix B, or type an asterisk (*) to specify the same label as the local node record, and press **Enter**.

Note: The Certificate Pathname field is automatically set to '*' (Default to Local) in the Remote Node record. You are not allowed to update this field for a remote node.

7. Select **STS Parameters** and press **Enter**.

```

Secure+ Create/Update Panel - STS Parameters
Option:
Node Identification      EA Parameters      SSL/TLS Parameters
Node                    SC.OKC.PARIS1
Auth Timeout: 120
Algorithm                *
Create / Reset Auth. Prev. Keys      Expire Date MDY
Create / Reset Sig. Prev. Keys      Expire Date MDY
----- < > -----
Create / Reset Auth. Pubkey | * |
Create / Reset Sig. Pubkey  | * |
Algorithm Names | DESCBC56, TDESCBC112, IDEACBC128 |
Auth. Rmt. Key | |
Sig. Rmt. Key | |
-----
Import Remote Keys      Get Record      OK      Cancel

```

8. In the **STS Parameters** panel:
 - a. To disable the STS protocol, type **2** in the following fields: **Autoupdt**, **Enable STS**, **Signature**, and **Encrypt**.
 - b. Type an asterisk (*) in the **Algorithm** field.

9. Select **EA Parameters** and press **Enter**.

```

Secure+ Create/Update Panel - EA Parameters
Option:
Node Identification      SSL/TLS Parameters      STS Parameters
Node                    1 1. Y 2. N 3. D Override
SC.OKC.PARIS1          2 1. Y 2. N 3. D External Auth
External Auth Server Def THISISATEST
External Auth Server Address 10.20.201.2
External Auth Server Port 23234
    
```

OK Cancel

10. In the **EA Parameters** panel:

- a. Type **3** beside the **Override** field because it is not relevant to External Authentication.
- b. Take one of the following actions, depending on whether the remote node validates client certificates using the Sterling External Authentication Server application.
 - Type **1** beside the **External Auth** field if this remote node uses the External Authentication Server application.
 - Type **2** beside the **External Auth** field if the remote node does not use the External Authentication Server application.
 - Type **3** beside the External Auth field if the remote node's use of the External Authentication Server defaults to the Local Node's setting.

The remaining EA parameters are unavailable because they are valid only for the .EASERVER remote node record.

11. Select **OK** and press **Enter** to save and close this remote node record.

12. Read all warning and error messages. You can continue configuring the environment without resolving warning messages, but you must resolve all errors before you save the parameter file.

13. Take one of the following actions:

- ◆ To configure an .EASERVER remote node record, continue with *Add a Remote Node Record for the EA Server* on page 83.

- ◆ To configure a .CLIENT remote node record, continue with *Establishing Secure TCP API Connections to a Secure+ Option-Enabled Server* on page 86.
- ◆ To configure records for remote nodes that use STS, continue with *Add a Remote Node Record to the Parameter File Manually for the STS Protocol* on page 74.
- ◆ If you have no other remote node records to configure, continue with the procedures in Chapter 11, *Enable and Validate Secure+ Operation*.

Create the Parameter File Manually for the STS Protocol

If you communicate with a large group of trading partners, but only a few trading partners use Secure+ Option, you can manually create and populate the parameter file by creating a single local node record and a remote node record for each trading partner that uses Secure+ Option. This method minimizes the number of remote node records to configure in the parameter file. However, because the local node record that you create manually does not specify meaningful default settings (all settings are default to local node), you must configure all parameters.

See Chapter 12, *Override Settings in Connect:Direct Processes* for information about overriding remote node record settings when executing a particular Process.

In addition to configuring local and remote node records, you must also perform the procedures to manage keys used with the STS protocol. See Chapter 10, *Manage Keys for the STS Protocol*, for instructions.

To validate and test a Secure+ connection between two business partners, see *Validate and Test Connections by Session* on page 134.

Configuration Guidelines

When you use the manual method to populate the parameter file, you configure the local node record to define the default settings for all protocols. Review the table in *Decide How to Create the Secure+ Parameter File* on page 44 to determine the configuration approach that best suits your needs, and use the following guidelines to configure node records manually:

- ◆ Disable external authentication.
- ◆ Because you are configuring only those nodes that use Secure+ Option, disable the Secure+ Option protocols (TLS, SSL, STS) in the local node record. Then configure each remote node record with the protocol used by that trading partner.

- ◆ To enable secure connections using Secure+ Option, you must complete the procedures in *Add the Local Node Record to the Parameter File Manually for the STS Protocol* on page 68, *Add a Remote Node Record to the Parameter File Manually for the STS Protocol* on page 74, and Chapter 11, *Enable and Validate Secure+ Operation*.
- ◆ You must perform the additional tasks in Chapter 10, *Manage Keys for the STS Protocol*.
- ◆ Read all warning and error messages. You can continue configuring the environment without resolving warning messages, but you may be unable to perform secure communications. You must resolve all errors before saving the Secure+ parameter file.
- ◆ See Chapter 12, *Override Settings in Connect:Direct Processes* for information about overriding remote node record settings.
- ◆ Before you start the Secure+ Admin Tool to add your local and remote node records, verify that you have allocated the ISPF libraries in your TSO session that are required to save the Secure+ parameter file (see *Allocate Connect:Direct ISPF Libraries in TSO* on page 15 for details).

Add the Local Node Record to the Parameter File Manually for the STS Protocol

Refer to the *Local Node Security Feature Definition Worksheet* on page 180 that you completed for the local node when you perform this procedure.

To add the local node record manually:

1. Select **Edit** from the **Secure+ Admin Tool Main Screen** and press **Enter** to display the **Edit** menu.

```

File  Edit  Key Management  Help
-----
                                Secure+ Admin Tool: Main Screen
Option ==>                                Scroll CSR

                                Table Line Commands are:

E Export pub. key      H View History      D Delete node
U Update node          I Insert node

                                Secure
LC Node Name          Type 123C Override Encryption Signature ExtAuth Autoupd
-----
***** BOTTOM OF DATA *****

```

2. On the **Edit** menu, type **1** to select **Create/Update Record** and press **Enter** to display the **Secure+ Create/Update Node Identification** panel.

```
Secure+ Create/Update Panel - Node Identification
Option:

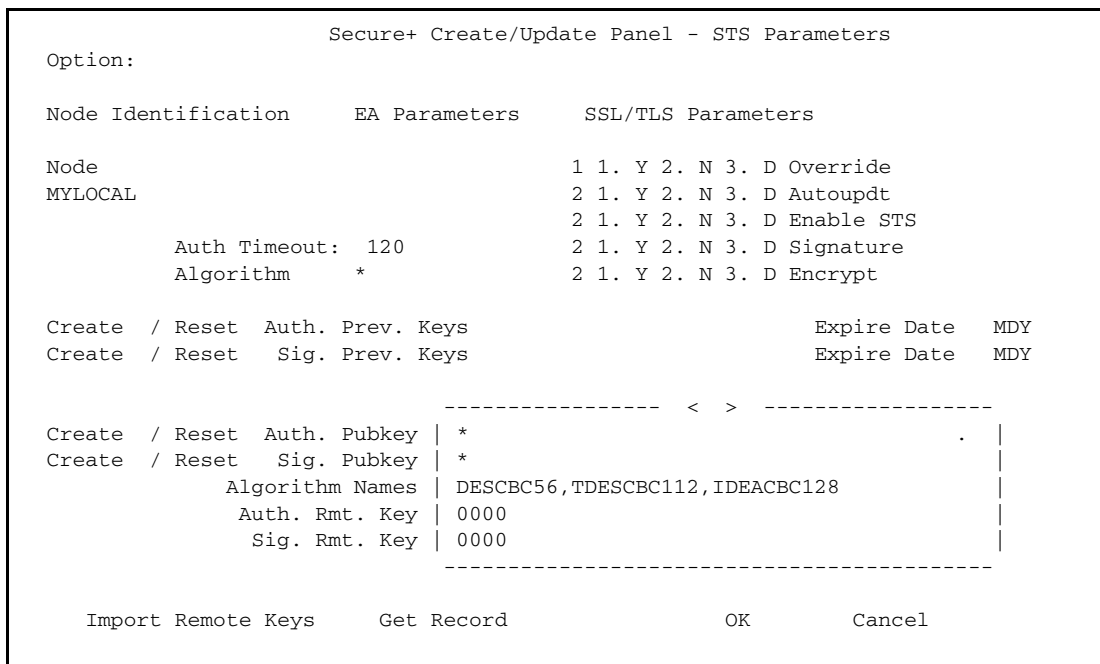
EA Parameters          SSL/TLS Parameters    STS Parameters

Node
                       1 1. Local
                       2. Remote

Alias
Names:                 TCP Information:
                       IPAddr:
                       Port:

Import Remote Keys    Get Record          OK          Cancel
```

3. In the **Node Identification** panel:
 - a. Type a name for the local node in the **Node** field.
 - b. To add the local node record, type **1** next to the **Local/Remote** field.
 - c. Leave the **TCP Information** fields (**IP address** and **Port**) blank because they are not valid for the local node record.
 - d. Leave the **Alias Names** field blank because it is not valid for the local node.
 - e. Select **STS Parameters** and press **Enter**.



4. In the **STS Parameters** panel:
 - a. Type **1** beside the **Override** field.
 - b. Disable Secure+ Option by typing **2** beside the following fields: **Autoupdt**, **Enable STS**, **Signature**, and **Encrypt**.
 - c. If necessary, change the values for the **Auth Timeout** and **Algorithm** fields using the following table as a guide:

Field Name	Description	Valid Values
Auth Timeout	Identifies the maximum time, in seconds, that the system waits to receive Connect:Direct control blocks exchanged during the authentication protocol.	0=No timeout. 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. Set the value equal to or greater than the value set for the Connect:Direct TCP.TIMER initialization parameter. The default is 120 seconds.
Algorithm	Specifies the data encryption algorithm used. Also set Encrypt to Yes.	* = Use first algorithm in list DESCBC56 TDESCBC112 IDEACBC128

5. Generate the authentication key for the STS protocol:
 - a. Select **Create/Reset Auth. Pubkey** and press **Enter** to display the Generate Seed screen.

```
Secure+ Admin Tool: Generate Seed

2 1. Specify Value          Specify the seed value by typing it
                             into the text field.
   2. Sample Value         Generate a seed by processing text
                             entered from the keyboard.

Random Number
Seed:
```

- b. Press **Enter** to accept the default value of **2-Sample Value**. The following screen is displayed.

```
File Edit Edit_Settings Menu Utilities Compilers Test Help
-----
EDIT          SYS06254.T160411.RA000.SSCHR1.R0207298          Columns 00001 00072
Command ==>                                     Scroll ==> PAGE
024: This process cannot proceed if the data in the edit file is unchanged.
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 262144
000002
000003 134217728
000004
000005 32
000006
000007 4
000008
000009 8589934592
000010
000011 131072
000012
000013 8388608
000014
000015 2097152
000016
000017 1073741824
```

- c. Edit or add data on any line and press **PF3** to save the information. Changing data creates a unique key value.
 - d. If the message *This is Loop 2 of 10* is displayed, type over data on any line and press **PF3** up to 10 times. This step is only necessary the first time you generate keys within the Secure+ Option Admin Tool for each parameter file. Repeating the process increases the randomness of keys.
 - e. When the **Pass Phrase Generation** panel is displayed, type a string at least 32 characters long containing at least one uppercase character, one lowercase character, and one numeric value and press **Enter**.

Note: You do not need to remember this number.

```

Secure+ Create/Update Panel - STS Parameters
Option:

Node Identification      EA Parameters          SSL/TLS Parameters

Node                    1 1. Y 2. N 3. D Override
MYLOCAL                 2 1. Y 2. N 3. D Autoupdt
                        2 1. Y 2. N 3. D Enable STS
                        2 1. Y 2. N 3. D Signature
                        2 1. Y 2. N 3. D Encrypt

                        Auth Timeout: 120
                        Algorithm      *

Create / Reset Auth. Prev. Keys          Expire Date MDY
Create / Reset Sig. Prev. Keys          Expire Date MDY

----- < > -----
Create / Reset Auth. Pubkey | 0203.093F.5D89.9024.5080.FE6D.7574.F55B. |
Create / Reset Sig. Pubkey | * |
                        Algorithm Names | DESCBC56,TDESCBC112,IDEACBC128 |
                        Auth. Rmt. Key | 0000 |
                        Sig. Rmt. Key | 0000 |
-----

Import Remote Keys      Get Record          OK          Cancel

```

When the **Secure+ Option Create/Update Panel** displays the message *Seed generation complete*, your public key for strong authentication has been generated and the **Create/Reset Auth. Pubkey** field is populated, as shown in the preceding illustration.

6. To generate the signature key:
 - a. Select **Create/Reset Sig. Pubkey** and press **Enter**.
 - b. Press **Enter** to accept the default value (**2-Sample Value**).

```

Secure+ Create/Update Panel - STS Parameters
Option:

Node Identification      EA Parameters          SSL/TLS Parameters

Node                    1 1. Y 2. N 3. D Override
MYLOCAL                 2 1. Y 2. N 3. D Autoupdt
                        2 1. Y 2. N 3. D Enable STS
                        2 1. Y 2. N 3. D Signature
                        2 1. Y 2. N 3. D Encrypt

                        Auth Timeout: 120
                        Algorithm      *

Create / Reset Auth. Prev. Keys          Expire Date MDY
Create / Reset Sig. Prev. Keys          Expire Date MDY

----- < > -----
Create / Reset Auth. Pubkey | 0203.093F.5D89.9024.5080.FE6D.7574.F55B. |
Create / Reset Sig. Pubkey | 0201.C2DB.B318.1B91.3A11.7FD3.3553.37EA. |
                        Algorithm Names | DESCBC56,TDESCBC112,IDEACBC128 |
                        Auth. Rmt. Key | 0000 |
                        Sig. Rmt. Key | 0000 |
-----

Import Remote Keys      Get Record          OK          Cancel

```


When the **Secure+ Option Create/Update Panel - STS Parameters** displays the message *Seed generation complete*, your public key for digital signature has been generated and the **Create/Reset Sig. Pubkey** field is populated, as shown in the preceding illustration.

7. Select **EA Parameters** and press **Enter**.
8. In the **EA Parameters** panel:
 - a. Type **2** beside the **External Auth** field to disable it. The remaining External Authentication fields are available only from the .EASERVER remote node record.

Note: Values set for parameters that are displayed in multiple panels (Override, for example) are retained in a record after you set them the first time they are displayed.

- b. Select **SSL/TLS Parameters** and press **Enter**.

Secure+ Create/Update Panel - SSL/TLS Parameters

Option:

Node Identification	EA Parameters	STS Parameters
Node	2 1. Y 2. N 3. D	Enable SSL
MYLOCAL	2 1. Y 2. N 3. D	Enable TLS

Common Parameters between SSL and TLS

1 1. Y 2. N 3. D	Override
2 1. Y 2. N 3. D	Client Auth
1 1. Y 2. N 3. D	Encrypt

Auth Timeout: 120

Certificate Label	*
Cipher Suites	FF
Certificate Pathname	*
Client Auth. Compare	

OK Cancel

9. In the **SSL/TLS Parameters** panel:
 - a. Type **2** beside the **Enable SSL** and **Enable TLS** fields to disable those protocols.
 - b. Type **1** beside the **Override** field.
 - c. Type **2** beside the **Client Auth** field because it is not valid for the STS protocol.
 - d. Type **1** beside the **Encrypt** field.
 - e. Leave the remaining fields as they are because they are not valid for the STS protocol.
10. Select **OK** and press **Enter**.
11. From the **Create/Update** panel, press **Cancel** to display the settings for the local node record.
12. Read all warning and error messages. You can continue configuring the environment without resolving warning messages, but you must resolve all errors that occur before you can save the parameter file.

13. After you configure the local node record, you can save and submit the parameter file using the procedures in Chapter 11, *Enable and Validate Secure+ Operation*, but if you have not added a remote node record, connections are not secure.

Add a Remote Node Record to the Parameter File Manually for the STS Protocol

Configure a remote node record for each trading partner that uses the STS protocol. Refer to the *Remote Node Security Feature Definition Worksheet* on page 181 that you created for a remote node that uses the STS protocol when you complete this procedure.

To add a remote node record manually for the STS protocol:

1. Select **Edit** from the **Secure+ Admin Tool Main Screen** and press **Enter** to display the **Edit** menu.

```

File  Edit  Key Management  Help
-----
                                         Row 1 of 1
Secure+ Admin Tool: Main Screen
Option ==>>>                                         Scroll CSR

Table Line Commands are:
E Export pub. key      H View History      D Delete node
U Update node         I Insert node

Secure
LC Node Name          Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
MYLOCAL               L     NNNN   Y           N           N           N           N
*****
***** BOTTOM OF DATA *****
    
```

2. On the **Edit** menu, type **1** to select **Create/Update Record** and press **Enter** to display the **Secure+ Create/Update Node Identification** panel.

```

Secure+ Create/Update Panel - Node Identification
Option:

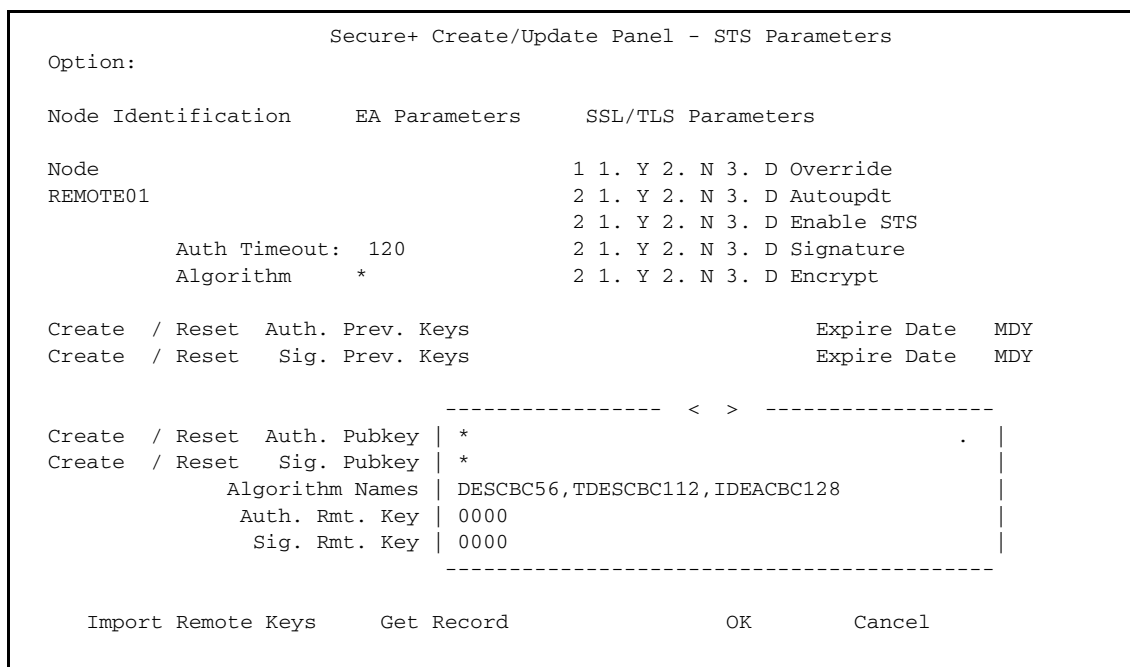
EA Parameters          SSL/TLS Parameters      STS Parameters

Node
                        2 1. Local
                        2. Remote

Alias
Names:                 TCP Information:
                        IPaddr:
                        Port:

Import Remote Keys    Get Record            OK            Cancel
    
```

3. In the **Node Identification** panel:
 - a. Type a name for the node in the **Node** field.
 - b. Type **2** next to the **Local/Remote** field.
 - c. Leave the **TCP Information** fields (**IP addr** and **Port**) blank because Connect:Direct always obtains the IP address and port for a remote node from the network map.
 - d. In the **Alias Names** field, type any alternative name for this remote node that uses the same Secure+ Option parameters. This alias name must also exist as a valid remote node entry in the network map.
 - e. Select **STS Parameters** and press **Enter**.



4. Because the **Override**, **Encrypt**, and **Signature** parameters work together, use the following table to determine the values to set in this remote node record:

Scenario	Setting for Override Parameter	Setting for Encrypt and Signature
All files must be encrypted and use signature.	Disable Override by setting to 2.	Enable Signature and Encrypt by setting to 1.
A few files must be encrypted and use signature.	Enable Override by setting to 1.	Disable Signature and Encrypt by setting to 2. You can change these settings in the COPY statement Process so that the individual files use encryption and signature. See Chapter 12, <i>Override Settings in Connect:Direct Processes</i> . For additional information, you can also go to the Connect:Direct Processes Web site .

In the **STS Parameters** panel, set values for the fields listed in the following table to enable the STS protocol:

Field	Description	Valid Values
Enable STS	<p>Enables or disables using the STS protocol for Connect:Direct Secure+ Option.</p> <p>Note: To specify that the session for a particular Process is to be secure using STS, disable STS by specifying 2 but enable the remote node to override the default of non-secure sessions by specifying YES for the OVERRIDE field. See Chapter 12, <i>Override Settings in Connect:Direct Processes</i>.</p>	<p>1 = Enable STS</p> <p>2 = Disable STS</p> <p>3 = Default to local node</p>
Autoupdt	Allows STS keys to be automatically updated when the values change.	<p>1=Yes</p> <p>2=No</p> <p>3=Default to local node</p>
Override	<p>Enables or disables the following security override functions on a session-by-session basis:</p> <ul style="list-style-type: none"> ◆ Allow values in the COPY statement to override values in the remote node record. ◆ Allow the STS protocol to be turned on or off for a particular session overriding the security default in the remote node record. 	<p>1=Yes</p> <p>2=No</p> <p>3=Default to local node</p>
Signature	Enables digital signatures for use with the STS protocol.	<p>1=Yes</p> <p>2=No</p> <p>3=Default to local node</p>
Encrypt	<p>Enables data encryption with the STS protocol, during the COPY operation. If you activate this feature, you must also populate the Algorithm field.</p> <p>Note: If the SNODE enables encryption, the PNODE cannot disable it.</p>	<p>1=Yes</p> <p>2=No</p> <p>3=Default to local node. This is not a valid value for the local node definition.</p>

5. Generate the authentication key as follows:
 - a. Select **Create /Reset Auth. Pubkey** and press **Enter** to display the **Generate Seed** screen.

```

Secure+ Admin Tool: Generate Seed

2 1. Specify Value          Specify the seed value by typing it
                             into the text field.
   2. Sample Value          Generate a seed by processing text
                             entered from the keyboard.

Random Number
Seed:

```

- b. Press **Enter** to accept the default value of **2-Sample Value**.
- c. When the following screen is displayed, edit or add data on any line and press **PF3** to save the information. Changing data creates a unique key value.

```

File Edit Edit_Settings Menu Utilities Compilers Test Help
-----
EDIT          SYS06254.T160411.RA000.SSCHR1.R0207298          Columns 00001 00072
Command ==>                                     Scroll ==> PAGE
024: This process cannot proceed if the data in the edit file is unchanged.
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 262144
000002
000003 134217728
000004
000005 32
000006
000007 4
000008
000009 8589934592
000010
000011 131072
000012
000013 8388608
000014
000015 2097152
000016
000017 1073741824

```

- d. When the **Pass Phrase Generation** panel is displayed, type a string at least 32 characters long containing at least one uppercase character, one lowercase character, and one numeric value and press **Enter**.

Note: You do not need to remember the pass phrase.

When the **Secure+ Option Create/Update Panel - STS Parameters** displays the message *Seed generation complete*, your public key for strong authentication has been generated and the **Create/Reset Auth. Pubkey** field is populated, as shown in the following illustration.

6. To generate the signature key:
 - a. Select **Create/Reset Sig. Pubkey** and press **Enter**.
 - b. Press **Enter** to accept the default value (**2-Sample Value**).

```

Secure+ Create/Update Panel - STS Parameters
Option:
Node Identification      EA Parameters      SSL/TLS Parameters
Node
REMOTE01
Auth Timeout: 120
Algorithm *
Create / Reset Auth. Prev. Keys      Expire Date MDY
Create / Reset Sig. Prev. Keys      Expire Date MDY
----- < > -----
Create / Reset Auth. Pubkey | 0203.093F.5D89.9024.5080.FE6D.7574.F55B. |
Create / Reset Sig. Pubkey | 0203.C2DB.B318.1B91.3A11.7FD3.3553.37EA. |
Algorithm Names | DESCBC56, TDESCBC112, IDEACBC128 |
Auth. Rmt. Key | 0000 |
Sig. Rmt. Key | 0000 |
-----
Import Remote Keys      Get Record      OK      Cancel
    
```

When the **Secure+ Option Create/Update Panel - STS Parameters** displays the message *Seed generation complete*, your public key for digital signature has been generated and the **Create/Reset Sig. Pubkey** field is populated, as shown in the preceding illustration.

7. Set values in one or more of the following fields as required:

Field Name	Field Description	Valid Values
Algorithm	Specifies the data encryption algorithm used. Also set Encrypt to Yes.	* = Use first algorithm in the Algorithm Names list below DESCBC56 TDESCBC112 IDEACBC128

Field Name	Field Description	Valid Values
Create / Reset Auth. Prev. Keys Expire Date	Identifies the expiration date, including the the time, for previous authentication public keys used with the STS protocol. This value eliminates the need to update Secure+ parameter file across all nodes in the network simultaneously when public keys for the local node are changed. Authentication Previous Keys are intended to be the most recently active keys prior to the current keys becoming active. To create previous keys, click Create . To clear the keys and any related expiration date, click Reset . The previous key is checked if an operation using the current key fails, and any expiration date for the previous key has not expired.	Format— You must enter both a date and time. Must match the date format specified in the Connect:Direct server's DATEFORM parameter (for more information, see the DATEFORM parameter in Appendix A, <i>Global Initialization Parameters</i> , in the <i>Connect:Direct for z/OS Administration Guide</i>). If connected to the Connect:Direct server, the current date format is displayed after the Expire Date fields and will be either MDY, DMY, YMD, or YDM. If the server cannot be logged on to, the default DATEFORM of MDY is used. For examples of valid date and time formats, see the Help panel.
Create / Reset Sig. Prev. Keys Expire Date	Identifies the expiration date and time for previous digital signature public keys used with the STS protocol. This value eliminates the need to update Secure+ parameter file across all nodes in the network simultaneously when public keys for the local node are changed. Signature Previous Keys are intended to be the most recently active keys prior to the current keys becoming active. To create previous keys, click Create . To clear the keys and any related expiration date, click Reset . The previous key is checked if an operation using the current key fails, and any expiration date for the previous key has not expired.	Format— You must enter both a date and time. Must match the date format specified in the Connect:Direct server's DATEFORM parameter (for more information, see the DATEFORM parameter in Appendix A, <i>Global Initialization Parameters</i> , in the <i>Connect:Direct for z/OS Administration Guide</i>). If connected to the Connect:Direct server, the current date format is displayed after the Expire Date fields and will be either MDY, DMY, YMD, or YDM. If the server cannot be logged on to, the default DATEFORM of MDY is used. For examples of valid date and time formats, see the Help panel.
Algorithm Names	Lists the acceptable data encryption algorithms to use when copy file encryption is requested. Listed in order of preference, with the most-preferred algorithm listed first.	DESCBC56 TDESCBC112 IDEACBC128
Import Remote Keys (on the Action selection line at the bottom of the panel)	Imports keys from the remote trading partner for both the Authorization and Signature functions in the STS protocol.	The name of the key file to import.

Field Name	Field Description	Valid Values
Auth Timeout	Identifies the maximum time, in seconds, that the system waits to receive Connect:Direct control blocks exchanged during the authentication protocol.	0=No timeout. 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. Set the value equal to or greater than the value set for the Connect:Direct TCP.TIMER initialization parameter. The default is 120 seconds.
Get Record	Opens another node record.	The name of an existing node record.

8. Select **EA Parameters** and press **Enter**.
9. In the **EA Parameters** panel:
 - a. Type **3** beside the **Override** field because it is not relevant to External Authentication.
 - b. Type **2** beside the **External Auth** field to disable it. The remaining External Authentication fields are available only from the .EASERVER remote node record.

Note: Values set for parameters that are displayed in multiple panels (Override, for example) are retained in a record after you set them the first time they are displayed.

- c. Select **SSL/TLS Parameters** in the panel selection line and press **Enter**.

```

Secure+ Create/Update Panel - SSL/TLS Parameters
Option:
Node Identification      EA Parameters      STS Parameters
Node                    2 1. Y 2. N 3. D Enable SSL
REMOTE01                2 1. Y 2. N 3. D Enable TLS

Common Parameters between SSL and TLS
1 1. Y 2. N 3. D Override
2 1. Y 2. N 3. D Client Auth
1 1. Y 2. N 3. D Encrypt

Auth Timeout: 120

Certificate Label      | * |
Cipher Suites          | FF |
Certificate Pathname   | * |
Client Auth. Compare  |   |

OK                      Cancel
    
```

10. In the **SSL/TLS Parameters** panel:
 - a. Type **2** beside the **Enable SSL** and **Enable TLS** fields to disable those protocols.
 - b. Type **1** beside the **Override** field.
 - c. Type **2** beside the **Client Auth** field because it is not valid for the STS protocol.
 - d. Type **1** beside the **Encrypt** field.
 - e. Leave the remaining fields as they are because they are not valid for the STS protocol.
11. Select **OK** and press **Enter** to display the updated values.
12. Read all warning and error messages. You can continue configuring the environment without resolving warning messages, but you may be unable to perform secure communications. You must resolve all errors before saving the parameter file.
13. Take one of the following actions:
 - ◆ To configure records for remote nodes that use a protocol other than STS, continue with the procedure, *Add a Remote Node Record to the Parameter File Manually for the SSL or TLS Protocol* on page 60.
 - ◆ If you have no other remote node records to configure, continue with the procedures in Chapter 11, *Enable and Validate Secure+ Operation*.

Additional Configuration Options

You can configure special-purpose remote node records to perform the following functions:

- ◆ Validate certificates using the Sterling Authentication Server application (TLS and SSL)
- ◆ Block nonsecure TCP API connections (TLS and SSL)
- ◆ Secure passwords at rest within the Connect:Direct TCQ and AUTH files (all protocols)

With the SSL and TLS protocols, you can validate certificates using the Sterling External Authentication Server application. To use the Sterling External Authentication Server application, configure your application to connect to the host name and port where the Sterling External Authentication Server application (.EASERVER) resides. Specify a certificate validation definition. For configuration instructions, see *Add a Remote Node Record for the EA Server* on page 83.

Use only secure TCP API connections to connect to a Connect:Direct for z/OS server. To block nonsecure TCP API connections, define a .CLIENT remote node record, disable override, and identify SSL or TLS as the protocol to use for secure TCP API connections. For configuration instructions, see *Establishing Secure TCP API Connections to a Secure+ Option-Enabled Server* on page 86.

In Connect:Direct, passwords can be used in Connect:Direct when Processes are submitted, during API signons, and when the AUTH file is maintained. You can use Strong Password Encryption (SPE) to secure passwords at rest within the Connect:Direct TCQ and AUTH files. See *Implementing Strong Password Encryption* on page 90.

Add a Remote Node Record for the EA Server

To verify certificates using the Sterling External Authentication Server application, create a remote node record for the External Authentication (EA) server in the Secure+ parameter file. Before you begin, complete the *.EASERVER Node Security Feature Definition Worksheet* on page 183.

To add a remote node record for the EA server:

1. Select **Edit** from the **Secure+ Admin Tool Main Screen** and press **Enter** to display the **Edit** menu.
2. On the **Edit** menu, type **1** to select **Create/Update Record** and press **Enter**.
3. In the **Node Identification** panel:
 - a. Type **.EASERVER** in the **Node** field.
 - b. Type **2** beside the **Local/Remote** field.

Note: Leave the **TCP Information** fields (**IP addr** and **Port**) blank because you define them in the EA Parameters panel.

- c. Select **EA Parameters** and press **Enter**.

Secure+ Create/Update Panel - EA Parameters

Option:

Node Identification	SSL/TLS Parameters	STS Parameters
Node	<u>1</u> 1. Y 2. N 3. D Override	
SC.OKC.PARIS1	<u>2</u> 1. Y 2. N 3. D External Auth	
External Auth Server Def	CertValidateDef	
External Auth Server Address	10.20.201.2	
External Auth Server Port	23234	

OK Cancel

4. In the **EA Parameters** panel:
 - a. Type **3** beside the **Override** field because it is not relevant to External Authentication.
 - b. Type **2** beside the **External Auth** field.
 - c. Type the information from the worksheet for the .EASERVER record in the following fields:

Field	Description
External Auth Server Def	Name of the certificate validation definition configured on the EA server that defines how to validate certificates. This field is case sensitive.
External Auth Server Address	IP address of server for the Sterling External Authentication Server application.

Field	Description
External Auth Server Port	Number of the port to use to connect to the EA server.

Note: After you create the .EASERVER remote node record, the **External Auth Server Def**, **External Auth Server Address**, and **External Auth Server Port** fields are populated in the EA Parameters panel of all Secure+ parameter file records, but the only field that can be modified from a record other than the .EASERVER record is the **Enable External Auth** field.

- d. Select **SSL/TLS Parameters** in the panel selection bar and press **Enter**.
5. In the **SSL/TLS Parameters** panel, enable Secure+ Option by typing **1** beside the **Enable TLS** or **Enable SSL** field. Select the other protocol you are not using for EA server connections and type **2** beside the **Enable TLS** or **Enable SSL** field. and press **Enter**.
6. To enable client authentication:
 - a. Type **1** beside the **Client Auth** field.
 - b. To have the common certificate name verified during the authentication process, select **Client Auth. Compare** and when the next panel displays, type the certificate common name of the local node certificate and press **Enter**. To not have the name verified, leave this field blank by not selecting the **Client Auth. Compare** field. If the common name is not entered, the client name verification process is not performed but client authentication is.

Note: This value is case-sensitive. Make sure that you type it exactly as it appears in the certificate file.

7. To enable and define the priority of ciphers:
 - a. Select the **Cipher Suites** field and press **Enter** to display the Update Cipher Suites Panel.
 - b. Type **1** by the cipher you want to enable and give the highest priority.
 - c. Continue typing numbers next to the ciphers you want to enable, in order of priority.
The ciphers you enable appear in the order of priority in the **Enabled Cipher-Suites** list.
 - d. Press **F3** when you have enabled and ordered all necessary ciphers.
8. To specify the certificate label:
 - a. Select the **Certificate Label** field and press **Enter**.
 - b. Press **F8** to move to the editable portion of the label field.
 - c. This field is case-sensitive, therefore, type the label of the certificate exactly as you defined it when you generated it using one of the security applications described in Appendix B, or type an asterisk (*) to specify the same label as the local node, and press **Enter**.

Note: The Certificate Pathname field is automatically set to '*' (Default to Local) in the Remote Node record. You are not allowed to update this field for a remote node.

9. Select **OK** and press **Enter** to save and close this remote node record.
10. Read all warning and error messages. You can continue configuring the environment without resolving warning messages, but you must resolve all errors before you save the parameter file.
11. Press **Cancel** to display current settings for the the EA node.

```

File  Edit  Key Management  Help
-----
                                         Row 1 of 1
                Secure+ Admin Tool: Main Screen
Option ==>                                         Scroll CSR

                Table Line Commands are:

E Export pub. key          H View History          D Delete node
U Update node              I Insert node

                Secure
LC Node Name              Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
.EASERVER                 R    NYNY   N           N           N           Y           N
***** BOTTOM OF DATA *****

```

12. Save the parameter file using the procedure in Chapter 11, *Enable and Validate Secure+ Operation*.

Establishing Secure TCP API Connections to a Secure+ Option-Enabled Server

Connect:Direct servers that use Secure+ Option allow you to allow secure TCP API connections or block nonsecure TCP API connections. Nonsecure API applications include Connect:Direct CICS Option, DMBATCH, ISPF IUI, z/OS Console interface, and Interconnect Option (ICO). Secure API applications can include Sterling Control Center and Connect:Direct Browser User Interface.

Note: It is not recommended but if you want to run Secure+ using a nonsecure API connection, you must set the S+.CMD.ENFORCE.SECURE.CONNECTION parameter to NO. See Appendix A, Global Initialization Parameters, in the Connect:Direct for z/OS Administration Guide for more information. In addition, you must specify OVERRIDE=YES in step 4 on page 88.

To prevent nonsecure TCP API connections, define a remote node record called .CLIENT and disable override. Additionally, identify the protocol to use for secure API connections. Defining a remote node called .CLIENT and disabling override prevents nonsecure connections to the Connect:Direct server without disabling override settings in the local node record.

If you define a .CLIENT record and disable override, also configure DMBATCH and ISPF IUI programs in Connect:Direct to use the SNA protocol. These programs are nonsecure TCP API connections.

An API configuration follows the same rules as other remote node connections with the following exceptions:

- ◆ API connections use either the SSL or the TLS security protocol.
- ◆ The Connect:Direct server supports TCP and defines a TCP API port for these connections. Refer to *Connect:Direct for z/OS Administration Guide* for instructions on setting up TCP API support on the server.
- ◆ Settings in the .CLIENT node definition automatically override the local node.

To configure a .CLIENT remote node record when Secure+ Option is enabled:

1. From the **Secure+ Admin Tool Main Screen**, select **Edit** and press **Enter** to display the **Edit** menu.
2. On the **Edit** menu, type **1** to select **Create/Update Record** and press **Enter**.

Secure+ Create/Update Panel - Node Identification

Option:

EA Parameters	SSL/TLS Parameters	STS Parameters
Node	1. Local 2. Remote	
Alias		
Names:	TCP Information: IPaddr: Port:	
Import Remote Keys Get Record OK Cancel		

3. On the **Node Identification** panel:
 - a. Type **.CLIENT** in the **Node** field.

Note: You must name this node .CLIENT in order for Connect:Direct to read this node and allow secure TCP API connections.

- b. Type **2** next to the **Local/Remote** field.

Note: Leave the **TCP Information** fields (**IP addr** and **Port**) blank because Connect:Direct always obtains the IP address and port for a remote node from the network map. Also, leave the **Alias Names** field blank.

- c. Select **EA Parameters** in the panel selection bar and press **Enter**.

```

Secure+ Create/Update Panel - EA Parameters
Option:
Node Identification      SSL/TLS Parameters      STS Parameters
Node                    2 1. Y 2. N 3. D Override
,CLIENT                 2 1. Y 2. N 3. D External Auth
External Auth Server Def
External Auth Server Address
External Auth Server Port
OK                        Cancel

```

4. In the **EA Parameters** panel:
- Type **2** beside the **Override** field to disable override for the `.CLIENT` remote node record.

Note: To allow nonsecure API connections, type **1** to enable overrides.

- Type **2** beside the **External Auth** field to disable it. The remaining EA parameters are unavailable because they are valid only for the `.EASERVER` remote node record.
 - Select **SSL/TLS Parameters** from the panel selection bar and press **Enter**.
5. Take one of the following actions, depending on whether you want to use the Secure+ Option parameter settings override feature:
- Type **1** beside the **Enable TLS** (or **Enable SSL**) field to enable the desired protocol for this remote node.
 - Type **3** beside the **Enable TLS** (or **Enable SSL**) field to default to the local node setting.
- Type **2** beside the **Client Auth** field to disable it.
 - Change the value in the **Auth Timeout** field, if necessary, using the following table as a guide:

Field	Description	Valid Values
Auth Timeout	Identifies the maximum time, in seconds, that the system waits to receive Connect:Direct control blocks exchanged during the authentication protocol.	0=No timeout. Connect:Direct waits indefinitely to receive the next message. 120 = Default. Specify a time to prevent malicious entry from taking as much time as necessary to attack the authentication protocol. Set the value equal to or greater than the value set for the Connect:Direct TCP.TIMER initialization parameter.

6. Select **STS Parameters** in the panel selection bar and press **Enter**.

```

Secure+ Create/Update Panel - STS Parameters
Option:
Node Identification      EA Parameters      SSL/TLS Parameters
Node
.CLIENT                 1 1. Y 2. N 3. D Override
                        2 1. Y 2. N 3. D Autoupdt
                        2 1. Y 2. N 3. D Enable STS
                        2 1. Y 2. N 3. D Signature
Auth Timeout: 120      2 1. Y 2. N 3. D Encrypt
Algorithm *            2 1. Y 2. N 3. D Encrypt

Create / Reset Auth. Prev. Keys      Expire Date MDY
Create / Reset Sig. Prev. Keys      Expire Date MDY

Create / Reset Auth. Pubkey | *
Create / Reset Sig. Pubkey | *
Algorithm Names | DESCBC56, TDESCBC112, IDEACBC128
Auth. Rmt. Key | 0000
Sig. Rmt. Key | 0000

----- < > -----
Import Remote Keys      Get Record      OK      Cancel
    
```

7. In the **STS Parameters** panel:
 - a. Disable the STS protocol by typing **2** beside the following fields: **Autoupdt**, **Enable STS**, **Signature**, and **Encrypt**.
 - b. Type an asterisk (*) beside the **Algorithm** field.
The remaining fields are not valid for the .CLIENT record.
8. Click **OK** and press **Enter** to save and close the .CLIENT node record.
9. Save the parameter file using the procedure in Chapter 11, *Enable and Validate Secure+ Operation*.
10. Ensure that the ISPF IUI and DMBATCH connections define SNA as the connection protocol.

Note: If the .CLIENT node record disables the Override function, ISPF IUI and DMBATCH must use the SNA protocol.

Implementing Strong Password Encryption

To implement the Strong Password Encryption (SPE), you add an SPE record to the Secure+ parameter file in the same way you would any remote node record. After you go through the following procedure and restart Connect:Direct Secure+ Option, the SPE feature will be in effect.

To add an SPE record to the Secure+ parameter file and enable the SPE feature:

1. Select **Edit** from the **Secure+ Admin Tool Main Screen** and press **Enter** to display the **Edit** menu.
2. On the **Edit** menu, select **1** to select **Create/Update Record** and press **Enter** to display the **Secure+ Create/Update Node Identification** panel.
3. On the Node Identification panel:
 - a. Type **.password** in the **Node** field.
 - b. Type **2** next to the **Local** field.

```

Secure+ Create/Update Panel - Node Identification
Option:

EA Parameters          SSL/TLS Parameters    STS Parameters

Node
.password              2 1. Local
                       2. Remote

Alias
Names:                 TCP Information:
                       IPAddr:
                       Port:

Import Remote Keys    Get Record           OK           Cancel

```

- c. Press **Enter** to display the **Secure+ Create/Update Panel - SPE Parameters** screen.
4. On the **SPE Parameters** panel, type **1** next to the **Enable SPE** field.

```

Secure+ Create/Update Panel - SPE Parameters

Option:

Node

.PASSWORD          1 1. Y 2. N  Enable SPE
                   (SPE not currently in use)

Password Public Key | * |
Algorithm Names     | TDESCBC112 |
                   ----- < > -----

                                   OK      Cancel
    
```

Press **Enter** to enable SPE and finish creating the SPE record by clicking **OK**.

5. Save the parameter file using the procedure in Chapter 12, *Enable and Validate Secure+ Operation*.
6. Restart Connect:Direct Secure+ Option.
7. To verify that Connect:Direct Secure+ Option Secure+ Option initialization is complete along with the SPE feature, after you restart Connect:Direct Secure+ Option, review the started task output for the following messages interspersed with the other initialization messages:

```

SITA460I Strong Password Encryption Initiated; CONNECT.CD.AUTH
SITA462I Strong Password Encryption Completed; CONNECT.CD.AUTH
SITA460I Strong Password Encryption Initiated; CONNECT.CD.TCQ
SITA462I Strong Password Encryption Completed; CONNECT.CD.TCQ
    
```

Note: These messages display even if no .PASSWORD record exists and no encryption is possible. If you return to the Secure+ Create/Update Panel - SPE Parameters screen where you enabled SPE, you should see (SPE currently in use) displayed to confirm that SPE has indeed been implemented.

Disabling Strong Password Encryption

If the Strong Password Encryption feature was backed out inappropriately by deleting the .PASSWORD record while at the same time passwords existed in the TCQ and AUTH files in the SPE format, you will see one of the messages listed in the following section, *Troubleshooting Possible SPE Problems*. Follow the procedure in this section, restart Connect:Direct Secure+ Option, and then enable the SPE feature again.

To disable the SPE feature:

1. Start the Secure+ Admin Tool to display the **Secure+ Admin Tool: Main Screen**, which displays the nodes populated from the Connect:Direct Secure+ Option network map along with other records in the Secure+ parameter file.

```

File Edit Key Management Help
-----
SC.DUB.DOVER1          Secure+ Admin Tool: Main Screen          Row 1 of 7
Option ==>                                                    Scroll CSR

                        Table Line Commands are:

E Export pub. key      H View History          D Delete node
U Update node          I Insert node

                        Secure
LC Node Name          Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
. CLIENT              R    NNNN   Y      N      N      N      N
. EASERVER            R    N*YN   N      N      N      N      N
. PASSWORD            R    Y***   *      *      *      *      *
SC.DUB.DOVER1        L    NNNN   Y      N      N      N      N
SC.DUB.DOVER2        R    NNYN   Y      N      N      N      N
SC.DUB.DOVER3        R    NNYN   Y      N      N      N      N
SC.DUB.DOVER4        R    NYNY   Y      N      N      N      N
***** BOTTOM OF DATA *****

```

2. Type **U** next to the **.PASSWORD** record and press **Enter** to display the **Secure+ Create/Update Panel - SPE Parameters** screen.
3. On the **SPE Parameters** panel, type **2** next to the **Enable SPE** field and press **Enter**.

```

Secure+ Create/Update Panel - SPE Parameters

Option:

Node

. PASSWORD                2 1. Y 2. N  Enable SPE
                        (SPE not currently in use)

                        < >
Password Public Key | * |
Algorithm Names    | TDESCBC112 |
                        -----
                        OK      Cancel

```

4. Save the parameter file using the procedure in Chapter 11, *Enable and Validate Secure+ Operation*.
5. Restart Connect:Direct Secure+ Option.

Troubleshooting Possible SPE Problems

If the Strong Password Encryption key stored in the **.PASSWORD** record is out of sync with the SPE key used to encrypt the passwords, errors can occur and you must reset all SPE passwords and reimplement the SPE feature.

The .PASSWORD record can get out of sync if one of the following occurs:

- ◆ You restore the .PASSWORD record from a backup of the Secure+ parameter file—The .PASSWORD record is updated and a new encryption key generated each time the Connect:Direct Secure+ Option server is restarted, so the backup will probably not contain the current parameters.
- ◆ The .PASSWORD record is deleted outside of Connect:Direct Secure+ Option and Secure+ Option—The .PASSWORD record is recreated as needed, so the SPE key used to encrypt the passwords no longer exists.
- ◆ The .PASSWORD record is corrupt—The SPE encryption key used to encrypt the passwords is not accessible.

The following tables identify errors you may experience when using the SPE feature, along with solutions to fix each issue.

Condition: Because of SPE errors, Connect:Direct Secure+ Option either initializes with a SITA461I message or does not initialize at all with a SITA463E message.

Error	Cause	Action
SITA461I SITA463E	SPE-formatted passwords exist In the TCQ and/or AUTH files, but Secure+ Option has not been enabled.	Connect:Direct Secure+ Option has not been set up to run with Secure+ Option. Add the SECURE.DSN= <i>filename</i> parameter to the initialization parameters, where <i>filename</i> is the name of the Secure+ parameter file. Restart Connect:Direct Secure+ Option. To see more detail information about individual errors related to the general failure, see the ESTAE trace output
	<ul style="list-style-type: none"> ◆ SPE-formatted passwords exist In the TCQ and/or AUTH files, but there is no .PASSWORD record in the Secure+ parameter file. ◆ SPE-formatted passwords exist in the TCQ and/or AUTH files, but the .PASSWORD record in the Secure+ parameter file has OLD encryption keys. This can only occur if an old Secure+ parmfile is restored with a backup that contains old keys. 	<p>Reset all passwords in the TCQ and AUTH files by performing these actions:</p> <ul style="list-style-type: none"> ◆ Select the AUTH file record in the AUTH file. Provide a new password and blank out all unusable data. ◆ In the TCQ file, delete all Processes and resubmit. <p>To see more detail information about individual errors related to the general failure, see the ESTAE trace output</p>

Condition: You encounter errors while trying to maintain the AUTH file.

Error	Cause	Action
SAFB023W SAFF016W SAFC016W SAFE016W	While inserting and updating users through the IUI (INSERT/UPDATE/SELECT/DELETE USER RECORD screen), Connect:Direct Secure+ Option could not read or record passwords. The .PASSWORD record does not contain the correct encryption key pair. The Secure+ parameter file may have been restored with an old copy of the .PASSWORD record.	<ol style="list-style-type: none"> 1. Disable the SPE feature. 2. Restart Connect:Direct Secure+ Option. 3. Enable the SPE feature again. 4. Restart Connect:Direct Secure+ Option. <p>To see more detail information about individual errors related to the general failure, see the ESTAE trace output</p>

Configure the Local Node Record Imported from the Network Map

The following procedures assume that you populated the parameter file by importing the network map. The Quick Start method creates a remote node record in the parameter file for each remote node record in the network map and a local node record. Using the Quick Start method to populate the parameter file is most efficient if you have a large number of trading partners that use the same protocol. You can enable that protocol in the local node record and because the remote node records are set automatically to Default to Local Node, they inherit the settings of the local node.

Depending on how you configure the local node record, you may or may not need to modify the remote node records. You must disable the Secure+ Option protocols in the records for all remote nodes that do not use Secure+ Option, and update all remote node records that use a protocol that is different from the protocol defined in the local node record.

Use the following procedures to configure the local node record imported from the network map for the SSL, TLS, and STS protocols:

- ◆ *Configure the Local Node Record for the SSL or TLS Protocol on page 96*
- ◆ *Configure the Local Node Record for the STS Protocol on page 103*

Configuration Guidelines

Observe the following guidelines when you configure node records imported from the network map:

- ◆ Secure+ Option protocols are disabled initially for all records created from the network map when you use Quick Start to populate the parameter file.
- ◆ You must create keys for the STS protocol because this action also creates the key that encrypts the Secure+ parameter file.
- ◆ Before you start the Secure+ Admin Tool to configure the local node record, you may want to save and submit the parameter file to verify that you can generate the SAVE AS JCL. If you

- are unable to generate the JCL for the SAVE AS job, verify that you have allocated the ISPF libraries in your TSO session that are required to save the Secure+ parameter file (see *Save and Submit the Secure+ Parameter File* on page 131).
- ◆ To enable secure connections using Connect:Direct Secure+ Option, you must complete the following procedures:
 - ◆ *Configure the Local Node Record for the SSL or TLS Protocol* on page 96 or *Configure the Local Node Record for the STS Protocol* on page 103
 - ◆ Relevant procedures in Chapter 9, *Configure Remote Node Records Imported from the Network Map*
 - ◆ Procedures in Chapter 11, *Enable and Validate Secure+ Operation*
 - ◆ Read all warning and error messages. You can continue configuring the environment without resolving warning messages, but you may be unable to perform secure communications. You must resolve all errors before saving the parameter file.

Configure the Local Node Record for the SSL or TLS Protocol

All Secure+ Option protocols are disabled when you import the network map. This procedure updates the local node record for the SSL (or TLS) protocol, enables the **Override** parameter, and verifies that the selected and STS protocols are disabled. Remember that all options set for the local node are inherited by all remote node records.

To update the local node record for the SSL (or TLS) protocol:

1. From the Secure+ Admin Main screen, type **U** next to the local node record and press **Enter** to display the **Secure+ Create/Update Panel** and the current values for the selected node.

Note: When you import the network map, the system enables **Override** in the local node record automatically, as shown in the following illustration.

Configure the Local Node Record for the SSL or TLS Protocol

```

File  Edit  Key Management  Help
-----
Row 9 to 13 of 13
Q2A.ZOS.PARIS      Secure+ Admin Tool: Main Screen
Option ===>                               Scroll CSR

Table Line Commands are:

E Export pub. key      H View History        D Delete node
U Update node          I Insert node

Secure
LC Node Name          Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
u Q2A.ZOS.PARIS      L    NNNN   Y      N      N      N      N
  Q3A.ZOS.PARIS      R    ***N   N      *      *      *      *
  Q3B.ZOS.PARIS      R    ***N   N      *      *      *      *
  SOL36SP             R    ***N   N      *      *      *      *
  W2S.4200.CDWOPS8   R    ***N   N      *      *      *      *
***** BOTTOM OF DATA *****

```

2. In the **STS Parameters** panel, verify that the following STS protocol parameters are disabled: **Autoupdt**, **Enable STS**, **Signature**, and **Encrypt**.

```

Secure+ Create/Update Panel - STS Parameters

Option:

Node Identification      EA Parameters      SSL/TLS Parameters

Node
Q2A.ZOS.PARIS           1 1. Y 2. N 3. D Override
                        2 1. Y 2. N 3. D Autoupdt
                        2 1. Y 2. N 3. D Enable STS
Auth Timeout: 120       2 1. Y 2. N 3. D Signature
Algorithm *             2 1. Y 2. N 3. D Encrypt

Create / Reset Auth. Prev. Keys      Expire Date
Create / Reset Sig. Prev. Keys      Expire Date

----- < > -----
Create / Reset Auth. Pubkey | * |
Create / Reset Sig. Pubkey | * |
Algorithm Names | DESCBC56, TDESCBC112, IDEACBC128 |
Auth. Rmt. Key | |
Sig. Rmt. Key | |
-----

Import Remote Keys      Get Record      OK      Cancel

```

3. Generate the STS protocol authentication key, which is part of the key pair that is used to encrypt the parameter file:
 - a. Select **Create/Reset Auth. Pubkey** and press **Enter** to display the **Generate Seed** screen.

```

Secure+ Admin Tool: Generate Seed

2 1. Specify Value          Specify the seed value by typing it
                             into the text field.
   2. Sample Value         Generate a seed by processing text
                             entered from the keyboard.

Random Number
Seed:

```

- b. Press **Enter** to accept the default value of **2-Sample Value**.
- c. When the following screen is displayed, edit or add data on any line and press **PF3** to save the information. Changing data creates a unique key value.

```

File  Edit  Edit_Settings  Menu  Utilities  Compilers  Test  Help
-----
EDIT          SYS06254.T160411.RA000.SSCHR1.R0207298          Columns 00001 00072
Command ==>                                         Scroll ==> PAGE
024: This process cannot proceed if the data in the edit file is unchanged.
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 262144
000002
000003 134217728
000004
000005 32
000006
000007 4
000008
000009 8589934592
000010
000011 131072
000012
000013 8388608
000014
000015 2097152
000016
000017 1073741824

```

- d. When the message *This is Loop 2 of 10* is displayed, type over data on any line and press **PF3** up to 10 times. This step is only necessary the first time you generate keys within the Secure+ Option Admin Tool for each parameter file. Repeating the process increases the randomness of keys.
- e. When the **Pass Phrase Generation** panel is displayed, type a string at least 32 characters long containing at least one uppercase character, one lowercase character, and one numeric value and press **Enter**.

Note: You do not need to remember the pass phrase.

Configure the Local Node Record for the SSL or TLS Protocol

```

Secure+ Create/Update Panel - STS Parameters
Option:

Node Identification      EA Parameters      SSL/TLS Parameters

Node
Q2A.ZOS.PARIS          1 1. Y 2. N 3. D Override
                        2 1. Y 2. N 3. D Autoupdt
                        2 1. Y 2. N 3. D Enable STS
                        2 1. Y 2. N 3. D Signature
                        2 1. Y 2. N 3. D Encrypt

                        Auth Timeout: 120
                        Algorithm      *

Create / Reset Auth. Pubkey | 0203.093F.5D89.9024.5080.FE6D.7574.F55B. |
Create / Reset Sig. Pubkey  | |
                        Algorithm Names | DESCBC56,TDESCBC112,IDEACBC128 |
                        Auth. Rmt. Key | |
                        Sig. Rmt. Key  | |

----- < > -----
Import Remote Keys      Get Record      OK      Cancel
    
```

When the **Secure+ Option Create/Update Panel - STS Parameters** displays the message *Seed generation complete*, your public key for strong authentication has been generated and the **Create/Reset Auth. Pubkey** field is populated, as shown in the preceding illustration.

4. Generate the signature key, which is part of the key pair used to encrypt the parameter file:
 - a. Select **Create/Reset Sig. Pubkey** and press **Enter**.
 - b. Press **Enter** to accept the default value (**2-Sample Value**).

```

Secure+ Create/Update Panel - STS Parameters
Option:

Node Identification      EA Parameters      SSL/TLS Parameters

Node
Q2A.ZOS.PARIS          1 1. Y 2. N 3. D Override
                        2 1. Y 2. N 3. D Autoupdt
                        2 1. Y 2. N 3. D Enable STS
                        2 1. Y 2. N 3. D Signature
                        2 1. Y 2. N 3. D Encrypt

                        Auth Timeout: 120
                        Algorithm      *

Create / Reset Auth. Prev. Keys      Expire Date
Create / Reset Sig. Prev. Keys      Expire Date

Create / Reset Auth. Pubkey | 0203.093F.5D89.9024.5080.FE6D.7574.F55B. |
Create / Reset Sig. Pubkey  | 0201.C2DB.B318.1B91.3A11.7FD3.3553.37EA. |
                        Algorithm Names | DESCBC56,TDESCBC112,IDEACBC128 |
                        Auth. Rmt. Key | 0000 |
                        Sig. Rmt. Key  | 0000 |

----- < > -----
Import Remote Keys      Get Record      OK      Cancel
    
```

When the **Secure+ Option Create/Update Panel - STS Parameters** displays the message *Seed generation complete*, your public key for digital signature has been generated and the **Create/Reset Sig. Pubkey** field is populated, as shown in the preceding illustration.

5. Select **SSL/TLS Parameters** in the panel selection line and press **Enter** to display the **SSL/TLS Parameters** panel.

```

Secure+ Create/Update Panel - SSL/TLS Parameters
Option:
Node Identification      EA Parameters      STS Parameters
Node
Q2A.ZOS.PARIS          1 1. Y 2. N 3. D Enable SSL
                       2 1. Y 2. N 3. D Enable TLS

Common Parameters between SSL and TLS

                       2 1. Y 2. N 3. D Override
                       1 1. Y 2. N 3. D Client Auth
                       1 1. Y 2. N 3. D Encrypt

Auth Timeout: 120

Certificate Label      | *
Cipher Suites         | FF
Certificate Pathname   | *
Client Auth. Compare  |
-----
OK                      Cancel
    
```

6. To select the protocol you want to enable, type **1** beside the **Enable SSL** or **Enable TLS** field. Type **2** beside the other field to disable the other protocol.

Note: If you attempt to set both SSL and TLS to 1 (or Y for Enabled), a warning displays indicating that the Enable SSL setting was changed to 2 (disabled) because you can only enable one protocol at a time for a particular node.

7. Type **1** in the **Override** field and **2** in the **Client Auth** field.
8. Take one of the following actions depending on the Encrypt option you want to implement:
 - a. To encrypt both the control block information and the files being transferred, type **1** beside the **Encrypt** field.
 - b. To encrypt only the control block information used to establish the session, type **2** beside the **Encrypt** field.
 - c. To default to the local node record, type **3** beside the **Encrypt** field.
9. Update the value for the **Auth Timeout** field, if necessary, using the following table as a guide:

Field	Description	Valid Values
Auth Timeout	Identifies the maximum time, in seconds, that the system waits to receive Connect:Direct control blocks exchanged during the authentication protocol.	0=No timeout. 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. Set the value equal to or greater than the value set for the Connect:Direct TCP.TIMER initialization parameter. The default is 120 seconds.

10. If necessary, update the certificate label:
 - a. Select the **Certificate Label** field and press **Enter**.
 - b. Press **F8** to move to the editable portion of the panel containing the label field.
 - c. This field is case sensitive; therefore, type the certificate label exactly as you defined it when you generated it using one of the security applications described in Appendix B and press **Enter**.
11. If necessary, update the location where the certificate information is stored:
 - a. Select the **Certificate Pathname** field and press **Enter** to display the **Certificate Pathname** panel.
 - b. Press **F8** to scroll to the editable portion of the panel containing the **Certificate Path Name** field.
 - c. Type the UNIX path name of the key database (.kdb) or the security system key ring name that contains all the certificates referred to in the parameter file.

Note: This value is case sensitive. Ensure that you type it exactly as it appears in the certificate file. Refer to the information you recorded in *Local Node Security Feature Definition Worksheet* on page 180.

- d. If you are using a key database:
 - a. Press **F8** to scroll to the password field.
 - b. Type the password used when the key database was created and press **Enter**.

Note: This value is case sensitive. Ensure that you type it exactly as it appears in the certificate file. Refer to the information you recorded in *Local Node Security Feature Definition Worksheet* on page 180.

Note: If you are using a key ring, leave the password field blank.

12. To update the enabled cipher suites:
 - a. Select the **Cipher Suites** field and press **Enter** to display the Update Cipher Suites panel.

```

More:      +
More:      +

Update the order field below to enable and order cipher suites.

O
r
d
e
r  All Available Cipher-Suites      Enabled Cipher-Suites

== =====
1  SSL_RSA_AES_128_SHA                SSL_RSA_AES_128_SHA
2  SSL_RSA_AES_256_SHA                SSL_RSA_AES_256_SHA
3  SSL_RSA_WITH_3DES_EDE_CBC_SHA     SSL_RSA_WITH_3DES_EDE_CBC_SHA
4  SSL_RSA_WITH_DES_CBC_SHA          SSL_RSA_WITH_DES_CBC_SHA
5  SSL_RSA_EXPORT_WITH_RC2_CBC_40_MD5 SSL_RSA_EXPORT_WITH_RC2_CBC_40_MD5
6  SSL_RSA_WITH_RC4_128_SHA          SSL_RSA_WITH_RC4_128_SHA
7  SSL_RSA_WITH_RC4_128_MD5          SSL_RSA_WITH_RC4_128_MD5
8  SSL_RSA_EXPORT_WITH_RC4_40_MD5    SSL_RSA_EXPORT_WITH_RC4_40_MD5
9  SSL_RSA_WITH_NULL_SHA              SSL_RSA_WITH_NULL_SHA
10 SSL_RSA_WITH_NULL_MD5              SSL_RSA_WITH_NULL_MD5
11 DEFAULT_TO_LOCAL_NODE              DEFAULT_TO_LOCAL_NODE

```

- b. Type **1** by the cipher suite you want to enable and give the highest priority.
- c. Type **2** by the cipher suite you want to enable and place second in priority.
- d. Continue typing numbers next to the cipher suites you want to enable, in order of priority.

The cipher suites you enable appear in the order of priority in the **Enabled Cipher-Suites** list.
- e. Press **F3** when you have enabled and ordered all necessary cipher suites.

Note: If you do not know what cipher suites are available, run a trace on the Connect:Direct system. Setting **debug=8C0000AE** in the initialization parameter file dynamically allocates DD R00000001. Available cipher suites are listed in the trace DD. Turn global tracing off before you continue.

13. Select **EA Parameters** and press **Enter**.
14. Verify that External Authentication (**External Auth**) is disabled (set to **2**). The remaining external authentication fields are unavailable because they are valid only for the .EASERVER remote node record.

15. Read all warning and error messages. You can continue configuring the environment without resolving warning messages, but you may be unable to perform secure communications. You must resolve all errors before you can save the parameter file.
16. After you configure the local node record, you can save and submit the parameter file using the procedures in Chapter 11, *Enable and Validate Secure+ Operation*, but if you have not added a remote node record, connections are not secure.

Configure the Local Node Record for the STS Protocol

All Secure+ Option protocols are disabled when you import the network map. This procedure updates the local node record for the STS protocol, enables the **Override** parameter, and verifies that the TLS and SSL protocols are disabled. Remember that all options set for the local node are inherited by all remote node records.

To update the local node record imported from the network map:

1. Type **U** next to the local node record and press **Enter**. The **Secure+ Create/Update Panel** displays the current values for the selected node.

Note: When you import the network map, the system enables **Override** in the local node record automatically, as shown in the following illustration.

```

File  Edit  Key Management  Help
-----
                                     Row 9 to 13 of 13
Q2A.ZOS.PARIS      Secure+ Admin Tool: Main Screen
Option ===>                                             Scroll CSR

                Table Line Commands are:

E Export pub. key      H View History          D Delete node
U Update node          I Insert node

                Secure
LC Node Name          Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
u Q2A.ZOS.PARIS      L    NNNN   Y      N      N      N      N
Q3A.ZOS.PARIS      R    ***N   N      *      *      *      *
Q3B.ZOS.PARIS      R    ***N   N      *      *      *      *
SOL36SP            R    ***N   N      *      *      *      *
W2S.4200.CDWOPS8   R    ***N   N      *      *      *      *
***** BOTTOM OF DATA *****
    
```

2. In the **STS Parameters** panel:
 - a. Type **1** beside the **Enable STS** field to enable the STS protocol.
 - b. Set values in one or more of the following fields as required to configure the STS protocol parameters:

Field Name	Field Description	Valid Values
Autoupdt	Allows STS keys to be automatically update when the values change.	1=Yes 2=No 3=Default to local node.
Override	Allows settings in a remote node record to override settings in the local node record.	1=Yes 2=No 3=Default to local node.
Signature	Enables digital signatures for use with the STS protocol.	1=Yes 2=No 3=Default to local node. This is not a valid value for the local node definition.
Encrypt	Enables data encryption with the STS protocol, during the copy operation. If you activate this feature, you must also populate Algorithm . If the SNODE enables encryption, the PNODE cannot disable it.	1=Yes 2=No 3=Default to local node. This is not a valid value for the local node definition.
Algorithm	Specifies the data encryption algorithm used. Also set Encrypt to Yes.	* = Use first algorithm in list DESCBC56 TDESCBC112 IDEACBC128
Sig. Prev. Keys Expire Date	Identifies the expiration date for previous digital signature public keys used with the STS protocol. This value eliminates the need to update Secure+ parameter files across all nodes in the network simultaneously when public keys for the local node are changed.	Format YYYY/MM/DD HH:MM:SS If time is not specified, 00:00:01 is used.
Auth. Pubkey	Generates the public key used for strong authentication with the STS protocol.	Generated by Secure+ Option. See step 3 on page 98.
Sig. Pubkey	Generates the public key used for digital signature with the STS protocol.	Generated by Secure+ Option. See step 4 on page 99.
Algorithm Names	Lists the acceptable data encryption algorithms to use when copy file encryption is requested. Listed in order of preference, with the most-preferred algorithm listed first.	DESCBC56 TDESCBC112 IDEACBC128 Not used with the TLS or SSL protocol.
Import Remote Keys	Imports keys from the remote trading partner for both the Authorization and Signature functions in the STS protocol.	The name of the key file to import.

Configure the Local Node Record for the STS Protocol

Field Name	Field Description	Valid Values
Auth. Prev. Keys Expire Date	Identifies the expiration date for previous authentication public keys used with the STS protocol. This value eliminates the need to update Secure+ parameter files across all nodes in the network simultaneously when public keys for the local node are changed.	Format YYYY/MM/DD HH:MM:SS If time is not specified, 00:00:01 is used.
Auth Timeout	Identifies the maximum time, in seconds, that the system waits to receive Connect:Direct control blocks exchanged during the authentication protocol.	0=No timeout. 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. Set the value equal to or greater than the value set for the Connect:Direct TCP.TIMER initialization parameter. The default is 120 seconds.

```

Secure+ Create/Update Panel - STS Parameters
Option:
Node Identification      EA Parameters      SSL/TLS Parameters
Node                    1 1. Y 2. N 3. D Override
Q2A.ZOS.PARIS          2 1. Y 2. N 3. D Autoupdt
                        2 1. Y 2. N 3. D Enable STS
                        Auth Timeout: 120
                        Algorithm      *      2 1. Y 2. N 3. D Signature
                        2 1. Y 2. N 3. D Encrypt

Create / Reset Auth. Prev. Keys      Expire Date
Create / Reset Sig. Prev. Keys      Expire Date

Create / Reset Auth. Pubkey |-----< >-----|
Create / Reset Sig. Pubkey |-----|
Algorithm Names | DESCBC56, TDESCBC112, IDEACBC128 |
Auth. Rmt. Key | 0000 |
Sig. Rmt. Key | 0000 |
-----|-----

Import Remote Keys      Get Record      OK      Cancel
  
```

3. Generate the authentication key for use with the STS protocol:
 - a. Select **Create/Reset Auth. Pubkey** and press **Enter** to display the **Generate Seed** screen.

```
Secure+ Admin Tool: Generate Seed

2 1. Specify Value          Specify the seed value by typing it
                             into the text field.
   2. Sample Value         Generate a seed by processing text
                             entered from the keyboard.

Random Number
Seed:
```

- b. Press **Enter** to accept the default value of **2-Sample Value**. The following screen is displayed.

```
File Edit Edit_Settings Menu Utilities Compilers Test Help
-----
EDIT          SYS06254.T160411.RA000.SSCHR1.R0207298          Columns 00001 00072
Command ===>                                     Scroll ===> PAGE
024: This process cannot proceed if the data in the edit file is unchanged.
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 262144
000002
000003 134217728
000004
000005 32
000006
000007 4
000008
000009 8589934592
000010
000011 131072
000012
000013 8388608
000014
000015 2097152
000016
000017 1073741824
```

- c. Edit or add data on any line and press **PF3** to save the information. Changing data creates a unique key value.
 - d. If the message *This is Loop 2 of 10* is displayed, type over data on any line and press **PF3** up to 10 times. This step is only necessary the first time you generate keys within the Secure+ Option Admin Tool for each parameter file. Repeating the process increases the randomness of keys.
 - e. When the **Pass Phrase Generation** panel is displayed, type a string at least 32 characters long containing at least one uppercase character, one lowercase character, and one numeric value and press **Enter**.

Note: You do not need to remember the pass phrase.

```

Secure+ Create/Update Panel - STS Parameters
Option:
Node Identification      EA Parameters      SSL/TLS Parameters
Node
Q2A.ZOS.PARIS          1 1. Y 2. N 3. D Override
                        2 1. Y 2. N 3. D Autoupdt
                        2 1. Y 2. N 3. D Enable STS
                        2 1. Y 2. N 3. D Signature
                        2 1. Y 2. N 3. D Encrypt
Auth Timeout: 120
Algorithm *
Create / Reset Auth. Prev. Keys      Expire Date
Create / Reset Sig. Prev. Keys      Expire Date
----- < > -----
Create / Reset Auth. Pubkey | 0303.ADB4.2924.EADD.FF27.4B7F.B248.E1CA |
Create / Reset Sig. Pubkey | |
Algorithm Names | DESCBC56,TDESCBC112,IDEACBC128 |
Auth. Rmt. Key | 0000 |
Sig. Rmt. Key | 0000 |
-----
Import Remote Keys      Get Record      OK      Cancel
    
```

When the **Secure+ Option Create/Update Panel** displays the message *Seed generation complete*, your public key for strong authentication has been generated and the **Create/Reset Auth. Pubkey** field is populated.

4. To generate the signature key:
 - a. Select **Create/Reset Sig. Pubkey** and press **Enter**.
 - b. Press **Enter** to accept the default value (**2-Sample Value**).

```

Secure+ Create/Update Panel - STS Parameters

Option:

Node Identification      EA Parameters          SSL/TLS Parameters

Node
Q2A.ZOS.PARIS          1 1. Y 2. N 3. D Override
                       2 1. Y 2. N 3. D Autoupdt
                       2 1. Y 2. N 3. D Enable STS
Auth Timeout: 120      2 1. Y 2. N 3. D Signature
Algorithm *            2 1. Y 2. N 3. D Encrypt

Create / Reset Auth. Prev. Keys          Expire Date
Create / Reset Sig. Prev. Keys          Expire Date

Create / Reset Auth. Pubkey | ----- < > ----- |
Create / Reset Sig. Pubkey | 0203.093F.5D89.9024.5080.FE6D.7574.F55B. |
                          | 0201.C2DB.B318.1B91.3A11.7FD3.3553.37EA. |
Algorithm Names | DESCBC56,TDESCBC112,IDEACBC128 |
Auth. Rmt. Key | 0000 |
Sig. Rmt. Key | 0000 |
                          | ----- |

Import Remote Keys      Get Record          OK          Cancel
    
```

When the **Secure+ Option Create/Update Panel** displays the message *Seed generation complete*, your public key for digital signature has been generated and the **Create/Reset Sig. Pubkey** field is populated, as shown in the preceding illustration.

5. Select **EA Parameters** and press **Enter**.
6. In the **EA Parameters** panel, type **2** beside the **Enable External Auth** field. The remaining external authentication fields are unavailable because they are valid only for the .EASERVER remote node record.
7. Select **OK** and press **Enter** to display the updated values.
8. Read all warning and error messages. You can continue configuring the environment without resolving warning messages, but you must resolve all errors that occur before you can save the parameter file.
9. After you configure the local node record, you can save and submit the parameter file using the procedures in Chapter 11, *Enable and Validate Secure+ Operation*, but if you have not added a remote node record, connections are not secure.

Configure Remote Node Records Imported from the Network Map

The following procedures assume that you populate the parameter file by importing the network map. The Quick Start method creates a remote node record in the parameter file for each remote node record in the network map and a local node record. Using the Quick Start method to populate the parameter file is most efficient if you have a large number of trading partners that use the same protocol. You can enable that protocol in the local node record and because the remote node records are set automatically to Default to Local Node, they inherit the settings of the local node.

Depending on how you configured the local node record, you may or may not need to modify the remote node records. You must disable the Secure+ Option protocols in the records for all remote nodes that do not use Secure+ Option, and update all remote node records that use a protocol that is different from the protocol defined in the local node record.

Use the following procedures to modify remote node records imported from the network map for the SSL, TLS, and STS protocols, and to disable all protocols for a remote node:

- ◆ *Configure a Remote Node Record for the SSL or TLS Protocol* on page 110
- ◆ *Configure a Remote Node Record for the STS Protocol* on page 114
- ◆ *Disable Secure+ Option in a Remote Node Record* on page 119

Your environment may have one or both of the following requirements:

- ◆ Blocking nonsecure TCP API connections
- ◆ Verifying certificates using the Sterling External Authentication Server application

For instructions on configuring these special-purpose remote node records for the TLS and SSL protocol, see the following procedures:

- ◆ *Establishing Secure TCP API Connections to a Secure+ Option-Enabled Server* on page 86
- ◆ *Add a Remote Node Record for the EA Server* on page 83

Configuration Guidelines

Observe the following guidelines when you configure node records imported from the network map:

- ◆ Secure+ Option protocols are disabled for all records created from the network map when you use Quick Start to populate the parameter file.
- ◆ For all environments, you must define required settings in the local node record. If desired, you can define optional settings in the local node record and use them in all remote node records.
- ◆ You must create keys for remote nodes that use the the STS protocol.
- ◆ To enable secure connections using Connect:Direct Secure+ Option, you must complete the relevant procedure for configuring the local node record in Chapter 8, *Configure the Local Node Record Imported from the Network Map*, the relevant procedures in this chapter, and the procedures in Chapter 11, *Enable and Validate Secure+ Operation*.
- ◆ Read all warning and error messages. You can continue configuring the environment without resolving warning messages, but you may be unable to perform secure communications. You must resolve all errors before saving the parameter file.

Configure a Remote Node Record for the SSL or TLS Protocol

After you configure the local node, you can configure remote node records. When you import the network map file, you create a remote node record in the parameter file for each remote node record in the network map. Depending on how you configured the local node record, you may or may not need to update the remote node records.

- ◆ If you disabled the Secure+ Option protocols in the local node record, Secure+ Option is disabled for all remote node records. You must update all remote node records that use Secure+ Option to identify which protocol is used by the trading partner.
- ◆ If you enabled a protocol in the local node record, that protocol is enabled in all remote node records. You must disable the Secure+ Option protocols in the records for all remote nodes that do not use Secure+ Option, and update all remote node records that use a protocol that is different from the protocol defined in the local node record.

Note: To override security functions for a particular session, you can use the SECURE parameter in the PROCESS statement. For more information, see Chapter 12, *Override Settings in Connect:Direct Processes*.

The following procedure assumes that you enabled the SSL (or TLS) protocol in the local node record, this remote node uses the SSL (or TLS) protocol, and that you need to modify some SSL (or TLS) parameters for this remote node record.

To update a remote node record for the SSL (or TLS) protocol:

1. Type **U** next to the remote node record to update and press **Enter** to display the current values for the selected node in the **Secure+ Create/Update Panel - STS Parameters** panel.

Note: An asterisk in a field on the Secure+ Admin Main Screen indicates the value Default to Local Node. If the TLS protocol is enabled in the Local Node record, Y appears in the third position instead of the second position in the Secure 123C column below.

```

File  Edit  Key Management  Help
-----
                                     Row 9 to 13 of 13
Q3B.ZOS.PARIS          Secure+ Admin Tool: Main Screen
Option ===>                                     Scroll CSR

                                     Table Line Commands are:

E Export pub. key      H View History        D Delete node
U Update node          I Insert node

                                     Secure
LC Node Name          Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
  Q2A.ZOS.PARIS       L    NYNN   Y          N          N          N          N
  Q3A.ZOS.PARIS       R    ***N   N          *          *          *          *
  u Q3B.ZOS.PARIS     R    ***N   N          *          *          *          *
  SOL36SP              R    ***N   N          *          *          *          *
***** BOTTOM OF DATA *****
    
```

2. In the **STS Parameters** panel:
 - a. Verify that the following parameters are disabled (set to **2**): **Override**, **Autoupdt**, **Enable STS**, **Signature**, and **Encrypt**, or set to Default to Local Node (**3**).
 - b. Verify that the following fields are set to Default to Local Node (**3**): **Create/Reset Auth. Pubkey**, and **Create/Reset Sig. Pubkey**.

```

Secure+ Create/Update Panel - STS Parameters
Option:
Node Identification      EA Parameters      SSL/TLS Parameters
Node
Q3B.ZOS.PARIS          1 1. Y 2. N 3. D Override
                        2 1. Y 2. N 3. D Autoupdt
                        2 1. Y 2. N 3. D Enable STS
                        2 1. Y 2. N 3. D Signature
                        2 1. Y 2. N 3. D Encrypt
Auth Timeout: 120
Algorithm *
Create / Reset Auth. Prev. Keys      Expire Date
Create / Reset Sig. Prev. Keys      Expire Date
----- < > -----
Create / Reset Auth. Pubkey | *
Create / Reset Sig. Pubkey | *
Algorithm Names | DESCBC56,TDESCBC112,IDEACBC128
Auth. Rmt. Key | 0000
Sig. Rmt. Key | 0000
-----
Import Remote Keys      Get Record      OK      Cancel

```

- c. Select **EA Parameters** and press **Enter**.
3. In the **EA Parameters** panel:
 - a. Specify a value for the External Authentication parameter, if required by your environment, using the following table as a guide:

Field	Description	Valid Values
External Auth	Allows validating certificates for secure sessions using the Sterling External Authentication Server application.	1=Yes 2=No 3=Default to local node

- b. Select **SSL/TLS Parameters** in the panel selection line and press **Enter**.
4. Take one of the following actions depending on the protocol you are implementing:
 - ◆ If you defined default SSL settings in the local node record that this remote node record uses, verify that the Enable TLS field is disabled (set to 2) or set to Default to Local Node (3). If you do not need to change any other settings, continue with step 10 on page 114.
 - If you defined default TLS settings in the local node record that this remote node record uses, verify that the Enable SSL field is disabled (set to 2) or set to Default to Local Node (3). If you do not need to change any other settings, continue with step 10 on page 114.

Note: If you attempt to set both SSL and TLS to 1 (or Y for Enabled), a warning displays indicating that the Enable SSL setting was changed to 2 (disabled) because you can only enable one protocol at a time for a particular node.

- ◆ To modify SSL (or TLS) protocol settings in a remote node record, continue with step 7.
5. Take one of the following actions, depending on what information you want to encrypt:
 - Type **1** beside the **Encrypt** field to encrypt all information sent during the handshake to set up communication sessions and the actual files being transferred.
 - Type **2** beside the **Encrypt** field to encrypt only the control block information sent during the handshake to set up communication sessions and not the actual files being transferred.
 6. Take one of the following actions, depending on whether you want to use the Secure+ Option parameter settings override feature.
 - To enable the Secure+ Option parameter settings override feature in the PROCESS or COPY statement, type **1** beside the **Override** field. For more information, see Chapter 12, *Override Settings in Connect:Direct Processes*.
 - To disable the Secure+ Option parameter settings override feature, type **2** beside the **Override** field.
 7. To change the list of cipher suites enabled for a remote node record:
 - a. Select the **Cipher Suites** field and press **Enter**. The Update Cipher Suites panel is displayed.
 - b. Type **1** by the cipher suite you want to enable and give the highest priority.
 - c. Continue typing numbers next to the cipher suites you want to enable, in order of priority. The cipher suites you enable appear in the order of priority in the **Enabled Cipher-Suites** list.
 - d. Press **F3** when you have enabled and ordered all necessary cipher suites.
 8. To enable client authentication:
 - a. Type **1** beside the **Client Auth** field.
 - b. To have the common certificate name verified during the authentication process, select **Client Auth. Compare** and when the next panel displays, type the certificate common name of the local node certificate and press **Enter**. To not have the name verified, leave this field blank by not selecting the **Client Auth. Compare** field. If the common name is not entered, the client name verification process is not performed but client authentication is.

Note: This value is case-sensitive. Make sure that you type it exactly as it appears in the certificate file.

 9. To specify the certificate label:
 - a. Select the **Certificate Label** field and press **Enter**.
 - b. Press **F8** to move to the editable portion of the panel containing the label field.

- c. This field is case sensitive; therefore, type the label of the certificate exactly as you defined it when you generated it using one of the security applications described in Appendix B, or type an asterisk (*) to specify the same label as the local node record, and press **Enter**.

Note: The Certificate Pathname field is automatically set to '*' (Default to Local) in the Remote Node record. You are not allowed to update this field for a remote node.

10. Select **OK** and press **Enter** to display the updated values.
11. Read all warning and error messages. You can continue configuring the environment without resolving warning messages, but you must resolve all errors that occur before you can save the parameter file.
12. Save the parameter file using the instructions in Chapter 11, *Enable and Validate Secure+ Operation*.

Configure a Remote Node Record for the STS Protocol

After you configure the local node, you can configure remote node records. When you imported the network map file, you created a remote node record in the parameter file for each remote node record in the network map. Depending on how you configured the local node record, you may or may not need to update the remote node record.

- ◆ If you disabled Secure+ Option in the local node record, Secure+ Option is disabled for all remote node records. You must update all remote node records that use Secure+ Option to identify which protocol is used by the trading partner.
- ◆ If you enabled a protocol in the local node record, that protocol is enabled in all remote node records. You must disable the Secure+ Option protocols in the records for all remote nodes that do not use Secure+ Option, and update all remote node records that use a protocol that is different from the protocol defined in the local node record.

Note: To override security functions for a particular session, you can use the SECURE parameter in the COPY or PROCESS statement. For more information, see Chapter 12, *Override Settings in Connect:Direct Processes*.

- ◆ You must perform additional tasks for those nodes that use the STS protocol. Refer to Chapter 10, *Manage Keys for the STS Protocol*.

The following procedure assumes that you enabled the STS protocol in the local node record, this remote node uses the STS protocol, and that you need to modify some STS parameters for this remote node record.

To update a remote node record for the STS protocol:

1. Type **U** next to the remote node record to update and press **Enter**. The **Secure+ Create/Update Panel** displays the current values for the selected node.

Configure a Remote Node Record for the STS Protocol

```

File  Edit  Key Management  Help
-----
                                                    Row 9 to 13 of 13
Q3B.ZOS.PARIS      Secure+ Admin Tool: Main Screen
Option ===>                                              Scroll CSR

                Table Line Commands are:

E Export pub. key      H View History      D Delete node
U Update node         I Insert node

                Secure
LC Node Name          Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
  Q2A.ZOS.PARIS      L   YNNN   Y      Y      Y      N      Y
  Q3A.ZOS.PARIS      R   ***N   N      *      *      *      *
u  Q3B.ZOS.PARIS      R   ***N   N      *      *      *      *
  SOL36SP            R   ***N   N      *      *      *      *
  W2S.4200.CDWOPS8   R   ***N   N      *      *      *      *
***** BOTTOM OF DATA *****

```

2. In the **STS Parameters** panel:

- a. Review the following table to determine the values to set for the **Override**, **Encrypt**, and **Signature** parameters because these parameters work together.

Scenario	Setting for Override Parameter	Setting for Encrypt and Signature
All files must be encrypted and use signature.	Disable Override by setting to 2.	Enable Signature and Encrypt by setting to 1. Note: If you disable Override, you cannot disable security in the PROCESS statement.
A few files must be encrypted and use signature.	Enable Override by setting to 1.	Disable Signature and Encrypt by setting to 2. You can override these settings in a COPY statement so that the individual files use encryption and signature. See Chapter 12, <i>Override Settings in Connect:Direct Processes</i> . For a complete description of the SECURE parameter, go to the Connect:Direct Processes Web site .

```

Secure+ Create/Update Panel - STS Parameters
Option:
Node Identification      EA Parameters      SSL/TLS Parameters
Node
Q3B.ZOS.PARIS          1 1. Y 2. N 3. D Override
                       2 1. Y 2. N 3. D Autoupdt
                       2 1. Y 2. N 3. D Enable STS
Auth Timeout: 120      2 1. Y 2. N 3. D Signature
Algorithm *            2 1. Y 2. N 3. D Encrypt

Create / Reset Auth. Prev. Keys      Expire Date
Create / Reset Sig. Prev. Keys      Expire Date

Create / Reset Auth. Pubkey |
Create / Reset Sig. Pubkey |
Algorithm Names | DESCBC56,TDESCBC112,IDEACBC128
Auth. Rmt. Key | 0000
Sig. Rmt. Key | 0000

Import Remote Keys      Get Record      OK      Cancel
    
```

b. Set values for the parameters listed in the following table to enable the STS protocol:

Field	Description	Valid Values
Enable STS	Enables or disables using the STS protocol for Connect:Direct Secure+ Option.	1 = Enable STS 2 = Disable STS 3 = Default to local node
Autoupdt	Allows STS keys to be automatically updated when the values change.	1=Yes 2=No 3=Default to local node
Override	Activating override in a remote node record that uses the STS protocol enables the values in the COPY statement to override values in the remote node record.	1=Yes 2=No 3=Default to local node
Signature	Enables digital signatures for use with the STS protocol.	1=Yes 2=No 3=Default to local node
Encrypt	Enables data encryption with the STS protocol during the COPY operation. If you activate this feature, you must also populate the Algorithm field. If the SNODE enables encryption, the PNODE cannot disable it.	1=Yes 2=No 3=Default to local node. This is not a valid value for the local node definition.
Algorithm	Specifies the data encryption algorithm used. Also set Encrypt to Yes.	* = Use first algorithm in list DESCBC56 TDESCBC112 IDEACBC128

Field	Description	Valid Values
Auth Timeout	Identifies the maximum time, in seconds, that the system waits to receive Connect:Direct control blocks exchanged during the authentication protocol.	0=No timeout. 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. Set the value equal to or greater than the value set for the Connect:Direct TCP.TIMER initialization parameter. The default is 120 seconds.

3. Generate the authentication key for use with the STS protocol:
 - a. Select **Create/Reset Auth. Pubkey** and press **Enter**. to display the **Generate Seed** screen.

```
Secure+ Admin Tool: Generate Seed

2 1. Specify Value          Specify the seed value by typing it
                             into the text field.
   2. Sample Value         Generate a seed by processing text
                             entered from the keyboard.

Random Number
Seed:
```

- b. Press **Enter** to accept the default value of **2-Sample Value**.
 - c. When the following screen is displayed, edit or add data on any line and press **PF3** to save the information. Changing data creates a unique key value.

```

File Edit Edit_Settings Menu Utilities Compilers Test Help
-----
EDIT          SYS06254.T160411.RA000.SSCHR1.R0207298          Columns 00001 00072
Command ==>                                         Scroll ==> PAGE
024: This process cannot proceed if the data in the edit file is unchanged.
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 262144
000002
000003 134217728
000004
000005 32
000006
000007 4
000008
000009 8589934592
000010
000011 131072
000012
000013 8388608
000014
000015 2097152
000016
000017 1073741824

```

4. Generate the signature key:
 - a. Select **Create/Reset Sig. Pubkey** and press **Enter**.
 - b. Press **Enter** to accept the default value (**2-Sample Value**).
5. Set values for one or more of the following parameters as required:

Field Name	Field Description	Valid Values
Sig. Prev. Keys Expire Date	Identifies the expiration date for previous digital signature public keys used with the STS protocol. This value eliminates the need to update Secure+ parameter files across all nodes in the network simultaneously when public keys for the local node are changed.	Format YYYY/MM/DD HH:MM:SS If time is not specified, 00:00:01 is used.
Algorithm Names	Lists the acceptable data encryption algorithms to use when copy file encryption is requested. Listed in order of preference, with the most-preferred algorithm listed first.	DESCBC56 TDESCBC112 IDEACBC128
Import Remote Keys	Imports keys from the remote trading partner for both the Authorization and Signature functions in the STS protocol.	The name of the key file to import.
Auth. Prev. Keys Expire Date	Identifies the expiration date for previous authentication public keys used with the STS protocol. This value eliminates the need to update Secure+ parameter files across all nodes in the network simultaneously when public keys for the local node are changed.	Format YYYY/MM/DD HH:MM:SS If time is not specified, 00:00:01 is used.

Field Name	Field Description	Valid Values
Get Record	Opens another node record.	The name of an existing node record.

6. Select **EA Parameters** and press **Enter**.
7. Verify that the **External Auth** field is disabled (set to **2**) or set to Default to Local Node (**3**).
8. Select **SSL/TLS Parameters** and press **Enter**.
9. Verify that the **Enable SSL**, **Enable TLS**, and the **Client Auth** fields are set to Default to Local Node (**3**) or disabled (**2**).
10. Select **OK** and press **Enter** to display the updated values.
11. Read all warning and error messages. You can continue configuring the environment without resolving warning messages, but you may be unable to perform secure communications. You must resolve all errors before saving the parameter file.
12. Save the parameter file using the procedure in Chapter 11, *Enable and Validate Secure+ Operation*.

Disable Secure+ Option in a Remote Node Record

If you have remote nodes that do not use Secure+ Option, then you must disable all protocols for those node.

To disable all protocols in a remote node record imported from the network map:

1. Type **U** next to the remote node record to update and press **Enter** to display the current values for the selected node in the **Secure+ Create/Update Panel - STS Parameters** panel.

Note: An asterisk in a field on the Secure+ Admin Main Screen indicates the value Default to Local Node.

```

File Edit Key Management Help
-----
Row 9 to 13 of 13
Q3B.ZOS.PARIS      Secure+ Admin Tool: Main Screen
Option ==>>>                                           Scroll CSR

Table Line Commands are:

E Export pub. key      H View History      D Delete node
U Update node         I Insert node

Secure
LC Node Name      Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
Q2A.ZOS.PARIS    L   NYNN   Y      N      N      N      N
Q3A.ZOS.PARIS    R   ***N   N      *      *      *      *
u Q3B.ZOS.PARIS  R   ***N   N      *      *      *      *
SOL36SP          R   ***N   N      *      *      *      *
W2S.4200.CDWOPS8 R   ***N   N      *      *      *      *
***** BOTTOM OF DATA *****

```

2. In the **STS Parameters** panel:
 - a. Disable (set to **2**) the following parameters: **Override**, **Autoupdt**, **Enable STS**, **Signature**, and **Encrypt**, if necessary.
 - b. Verify that the following fields are set to Default to Local Node (*): **Create/Reset Auth. Pubkey**, and **Create/Reset Sig. Pubkey**.

```

Secure+ Create/Update Panel - STS Parameters
Option:
Node Identification      EA Parameters      SSL/TLS Parameters
Node
Q3B.ZOS.PARIS           1 1. Y 2. N 3. D Override
                        2 1. Y 2. N 3. D Autoupdt
                        2 1. Y 2. N 3. D Enable STS
Auth Timeout: 120      2 1. Y 2. N 3. D Signature
Algorithm *            2 1. Y 2. N 3. D Encrypt

Create / Reset Auth. Prev. Keys      Expire Date
Create / Reset Sig. Prev. Keys      Expire Date

Create / Reset Auth. Pubkey | *
Create / Reset Sig. Pubkey | *
Algorithm Names | DESCBC56, TDESCBC112, IDEACBC128
Auth. Rmt. Key | 0000
Sig. Rmt. Key | 0000
----- < > -----
Import Remote Keys      Get Record      OK      Cancel

```

3. Select **EA Parameters** and press **Enter**.

4. In the **EA Parameters** panel, disable the External Authentication parameter by typing **2** beside the **External Auth** field, if necessary. The remaining external authentication parameters are unavailable because they are valid only for the .EASERVER remote node record.
5. Select **SSL/TSL Parameters** in the panel selection line and press **Enter**.
6. If necessary, disable the SSL and TLS protocols by typing **2** beside the **Enable SSL** and **Enable TLS** fields, if necessary.
7. Select **OK** and press **Enter** to display the updated values.
8. Read all warning and error messages. You can continue configuring the environment without resolving warning messages, but you may be unable to perform secure communications. You must resolve all errors before saving the parameter file.
9. Save the parameter file using the instructions in Chapter 11, *Enable and Validate Secure+ Operation*.

Manage Keys for the STS Protocol

For nodes that use the STS protocol, you are responsible for managing the keys that you create. 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 to establish a secure connection with that node. The first time you use the STS protocol, you manually exchange keys with each trading partner. After you exchange keys for the first communications session with Secure+ Option, if you have enabled the Autoupdate parameter, then the STS public keys are updated automatically for subsequent communications sessions, which simplifies key management for ongoing communications.

To manage the keys for the STS protocol, perform the following tasks:

- ◆ Export keys to your trading partner
- ◆ Import keys from your trading partner
- ◆ Validate the Secure+ parameter file

Initial Exchange of STS Keys

The first time you exchange STS keys with your trading partners that use the STS protocol, you must perform the following steps on both systems where Secure+ Option is installed.

To exchange STS keys the first time:

1. Open the remote node records for the trading partners that use the STS protocol and ensure that the Enable STS parameter is set to **1** on both systems where Secure+ Option is installed.

Note: The STS protocol must be enabled in the remote node record in order to export the STS keys to a file.

2. Export your STS keys using the procedure *Export STS Keys* on page 124.
3. Open the remote record again and disable Secure+ by typing **2** beside the **Enable STS** field in the STS Parameters panel.

Note: You must disable Secure+ until you have transferred your keys to your trading partner, imported keys from the remote trading partner, and enabled Secure+ because your sessions will fail if you have enabled the STS protocol but you have not exchanged keys.

4. Save the parameter file.
5. To ensure the integrity of the export file that contains your STS keys, create a Connect:Direct Process to transfer the export key file to your trading partner.

Note: If you are sending the export key file to a non-mainframe node, you must send it in binary format.

6. Import the STS keys from your trading partner using the procedure *Import STS Keys from a File* on page 126.
7. In the STS Parameters panel for the remote node records of the trading partners that use the STS protocol, set the **Enable STS** parameter to **1** to enable Secure+ Option on both systems where Secure+ Option is installed.
8. Save the parameter files on both systems where Secure+ Option is installed.

After you complete this procedure, all connections that use the STS protocol are made using Secure+ Option unless you have enabled the Override parameter in the remote node record and override the Secure+ Option parameter settings from the COPY statement.

After you exchange keys for the first communications session with Secure+ Option, if you enabled the Autoupdt parameter, then the STS public keys are updated automatically for subsequent communications sessions, which simplifies key management for ongoing communications.

Export STS Keys

After you create signature and authentication keys for a node record, you must send this information to the trading partner. Export the information to a file that you can send to the trading partner.

To export the authentication and digital signature public key values:

1. From the **Secure+ Admin Tool: Main Screen**, select **Key Management** and press **Enter**.

Import STS Keys from a File

When you receive the STS key file from the administrator of the remote node (trading partner), you must import it to the Secure+ parameter file. This example illustrates how to import the remote node key file named *USERID.SECURE.EXPQ3A.#CSGPRO.#D390*.

To import the authentication and digital signature public key values:

1. From the **Secure+ Admin Tool: Main Screen**, select **Key Management** and press **Enter**.
2. Perform one of the following actions:
 - ◆ Type **1** to select **Import Public Keys** and press **Enter**. This option imports keys from your trading partner for all nodes in the parameter file.
 - ◆ Type **U** next to the node for which you want to import keys.

```

File  Edit  Key Management  Help
-----+-----+-----+-----+-----+-----+-----+-----+-----+
| 1  1. Import Public keys      |                               | Row 1 of 10
CSG.PROD39 | 2. Export Public Keys      | n Screen
Option === | *. Distribute Public Keys   |                               | Scroll CSR
-----+-----+-----+-----+-----+-----+
                               |
                               | Table Line Commands are:
                               |
E Export pub. key      H View History      D Delete node
U Update node          I Insert node
                               |
                               | Secure
LC Node Name          Type  123C  Override Encryption Signature ExtAuth Autoupd
-----+-----+-----+-----+-----+-----+-----+
AIX3601SP             R    ***N   N          *          *          *          *
CSG.PROD390           L    NNNN   Y          N          N          N          N
Q1A.ZOS.PARIS         R    ***N   N          *          *          *          *
Q1E.ZOS.PARIS         R    ***N   N          *          *          *          *
Q1G.ZOS.PARIS         R    ***N   N          *          *          *          *
Q2A.ZOS.PARIS         R    ***N   N          *          *          *          *
Q3A.ZOS.PARIS         R    Y***   N          N          N          *          N
W2S.4200.CDWOPS8     R    ***N   N          *          *          *          *

```

3. Type the file name prefix or partial prefix followed by an asterisk (*), select **Browse**, and press **Enter**, or type the complete file name that you received from your trading partner, as shown in the following illustration.

```

Secure+ Admin Tool: File Selection

Enter file name for: Secure Import File

File
Name:      USERID.SECURE.EXPQ3A.#CSGPRO.#D390      Browse

File System Type:
1 1. MVS  2. HFS      Cancel
    
```

4. Type **S** next the file to import, (ensure that the file extension of the import file includes the node name) and press **Enter**.

```

Secure+ Admin Tool: File Selection      Row 1 of 1

Option: _____ Scroll CSR

Enter "S" on the line of the file for  for MVS.

LC Filename or Directory
S USERID.SECURE.EXPQ3A.#CSGPRO.#D390
***** Bottom of data *****
    
```

The message *2 entries imported from othernode* is displayed on the **Secure+ Admin Tool: Main Screen**, indicating that both the authentication and the digital signature public keys have been imported.

```

File  Edit  Key Management  Help
-----
                                Row 7 of 10

CSG.PROD390      Secure+ Admin Tool: Main Screen
024: 2 entries imported from Q3A.ZOS.PARIS

Table Line Commands are:

E Export pub. key      H View History      D Delete node
U Update node          I Insert node

Secure
LC Node Name          Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
Q3A.ZOS.PARIS         R     Y***  N           N           N           *           N
W2S.4200.CDWOPS8      R     ***N  N           *           *           *           *
135.71.104.3          R     ***N  N           *           *           *           *
199.0.91.45           R     ***N  N           *           *           *           *
***** BOTTOM OF DATA *****
    
```

- Type **U** next to the remote node record name and press **Enter** to verify that the remote public keys are imported. If the keys have been imported, the keys are displayed in the **Auth. Rmt. Key** and the **Sig. Rmt. Key** fields.

```

Secure+ Create/Update Panel - STS Parameters
Option:
Node Identification      EA Parameters          SSL/TLS Parameters
Node                   1 1. Y 2. N 3. D Override
Q3A.ZOS.PARIS         2 1. Y 2. N 3. D Autoupdt
                      2 1. Y 2. N 3. D Enable STS
                      Auth Timeout: 120    2 1. Y 2. N 3. D Signature
                      Algorithm *          2 1. Y 2. N 3. D Encrypt

Create / Reset Auth. Prev. Keys          Expire Date
Create / Reset Sig. Prev. Keys          Expire Date

Create / Reset Auth. Pubkey | ----- < > ----- |
Create / Reset Sig. Pubkey | 0303.1C2E.630C.E557.4F27.B1BB.358F.BB3A. |
Algorithm Names | 0206.4C8C.575F.5956.35D8.C91F.EB0E.CAE8. |
Auth. Rmt. Key | DESCBC56, TDESCBC112, IDEACBC128 |
Sig. Rmt. Key | 0200.0A7E.686B.49FC.B1E1.1BC3.E844.BA5B. |
                   | 0301.46C6.20A2.0F09.70F0.9C7D.E401.DE1A. |

Import Remote Keys      Get Record          OK          Cancel
    
```

After you and your trading partner have imported each other's keys, you can verify that you have imported the keys correctly by displaying the STS Parameters panel for the remote node record on each system and validating that the key values have the correspondence illustrated in the following table. See the STS Parameters panels preceding and following this table for the data used to illustrate the correspondence.

Node Record	Parameter and Value	Node Record	Parameter and Value
Q3A.ZOS.PARIS	Create / Reset Auth. Pubkey= 0303.1C2E.630C.E557.4F27.B1B B.358F.BB3A.	CSG.PROD390	Auth. Rmt. Key= 0303.1C2E.630C.E557.4F27.B1BB.3 58F.BB3A.
Q3A.ZOS.PARIS	Create / Reset Sig. Pubkey= 0206.4C8C.575F.5956.35D8.C91F .EB0E.CAE8.	CSG..PROD390	Sig. Rmt. Key= 0206.4C8C.575F.5956.35D8.C91F.E B0E.CAE8.
CSG.PROD390	Create / Reset Auth. Pubkey 0303.1C2E.630C.E557.4F27.B1B B.358F.BB3A.	Q3A.ZOS.PARIS	Auth.Rmt. Key= 0303.1C2E.630C.E557.4F27.B1BB.3 58F.BB3A.
CSG.PROD390	Create / Reset Sig. Pubkey= 0206.4C8C.575F.5956.35D8.C91F .EB0E.CAE	Q3A.ZOS.PARIS	Sig. Rmt. Key= 0206.4C8C.575F.5956.35D8.C91F.E B0E.CAE


```

Secure+ Create/Update Panel - STS Parameters
Option:
Node Identification      EA Parameters      SSL/TLS Parameters
Node
CSG.PROD390
Auth Timeout: 120
Algorithm *
Create / Reset Auth. Prev. Keys      Expire Date
Create / Reset Sig. Prev. Keys      Expire Date
----- < > -----
Create / Reset Auth. Pubkey | 0200.0A7E.686B.49FC.B1E1.1BC3.E844.BA5B. |
Create / Reset Sig. Pubkey | 0301.46C6.20A2.0F09.70F0.9C7D.E401.DE1A. |
Algorithm Names | DESCBC56,TDESCBC112,IDEACBC128 |
Auth. Rmt. Key | 0303.1C2E.630C.E557.4F27.B1BB.358F.BB3A. |
Sig. Rmt. Key | 0206.4C8C.575F.5956.35D8.C91F.EB0E.CAE8. |
-----
Import Remote Keys      Get Record      OK      Cancel

```

Import STS Keys Manually

If you do not have a key file from which to import, but you have a hard copy printout of the keys, you must type the keys into the parameter file.

To type keys into the parameter file:

1. Locate an existing export file.
2. Create a data set with DCB= LRECL=255 BLKSIZE=23200 RECFM=VB and copy the export file from step 1 on page 129 into this data set.

Note: This step provides a guide for preallocation of the file.
 If you get the error *Import fails with "024 0 entries imported"* when the file is preallocated with this DCB information, do not specify DCB information. Use the DCB information defined in the input file.
 If you preallocate the file and the error message is displayed, delete the preallocated file and reallocate the file with larger lrecl and blksize values.

3. Using the DMADR002 display tool, display the existing export file in readable format to create a temporary data set.
4. Use the REPRO command on the temporary data set to copy the data into the data set created in step 2 on page 129.

Note: You can also cut the data from the temporary data set and paste it into the new data set.

5. Edit the new data set, and change the auth.pubkey and sig.pubkey data according to the key file. Also change the export and import node names to match exactly the names of your local and remote nodes.
6. Import the new data set as an export file into your parameter file using the procedure described in *Import STS Keys from a File* on page 126.

Enable and Validate Secure+ Operation

After you initially configure the local and remote nodes for Connect:Direct Secure+ Option, you must save and submit the parameter file and prepare Connect:Direct for operation, as described in the following procedures. As a final step, validate and test connections between you and your business partners to establish secure communications and then test to make sure you can change your security defaults for a session.

During normal maintenance after you make changes to the parameter file, you can use the Save Active option to dynamically save updates without having to restart Connect:Direct. See *Save Changes to Node Records Using the Save Active Option* on page 145.

If Connect:Direct is running and Secure+ is enabled, you must use the Save Active option. Otherwise, you must use the Save As option. (When Secure+ is enabled, the Secure+ parameter file is open and Connect:Direct cannot perform the Save As option since that option deletes, redefines, and reloads the parameter file.)

Save and Submit the Secure+ Parameter File

This procedure assumes that you have verified that the following required Connect:Direct ISP libraries (all of the same release) have been allocated in your TSO session:

- ◆ ISPCLIB (must be allocated as SYSPROC)
- ◆ ISPLLIB
- ◆ ISPPLIB
- ◆ ISPSLIB
- ◆ ISPMLIB

If these required libraries have not been allocated, or have been allocated incorrectly, when you perform this procedure, the JCL for the SAVE AS job is not generated, and you have to repeat the procedures to configure the local and remote nodes. For information on how to allocate these libraries, see *Connect:Direct for z/OS Installation Guide*.

To save the Secure+ parameter file:

1. From the **Secure+ Option Admin Tool: Main Screen**, select **File** and press **Enter**.

```

File Edit Key Management Help
+-----+
| 2 1. New          | |
| 2. Open          | | Secure+ Admin Tool: Main Screen
| *. Close         | | Scroll CSR
| 4. Info...       | |
| *. Rekey         | | Table Line Commands are:
| *. Save Active   | |
| 7. Save as...    | | H View History          D Delete node
| *. Unload        | | I Insert node
| 9. Exit          | |
+-----+
                Secure
LC Node Name      Type  123C  Override Encryption Signature ExtAuth Autoupd
+-----+
. CLIENT          R    NNNN   Y           N           N           N           N
.EASERVER         R    N*YN   N           N           N           N           N
SC.OKC.PARIS1    L    NNNN   Y           N           N           N           N
SC.REMOTE01      R    NNYN   Y           N           N           N           N
SC.REMOTE02      R    NYNN   Y           N           N           N           N
SC.REMOTE03      R    NYNY   Y           N           N           N           N
***** BOTTOM OF DATA *****

```

2. Type **7** to select **Save As**.
3. Type the file name you want to use for the Secure+ parameter file and press **Enter**. (You will use this same file name in step 1 on page 133 when you add this information as a parameter to the Connect:Direct initialization parameter file to tell Connect:Direct where security information is located.)

Caution: The default **Save As** file name is the name of the last file that you opened. When you create the Secure+ parameter file from the Connect:Direct network map, you risk overwriting the network map file with the Secure+ parameter file if you do not change the name in this field.

If the file specified here exists, you will be prompted that it will be deleted, redefined, and built anew if you select **OK**.

4. On the Save As information panel, type site-specific job card information, allocation information, STEPLIB DSNs, and Access file Dsname, using the library names created when you saved the parameter file.

Caution: For the Access file Dsname, you must not use the same name specified for the Secure+ parameter file. However, the Access file and Secure+ parameter file act as a pair and one is not any good without the other. To more easily track the files if necessary, it is recommended that at least the first two qualifiers match. For example, you could use \$CD.SECURE.PARM and \$CD.SECURE.ACCESS for the names of the related Secure+ parameter and access files.

5. Type **3** to select **Submit** and press **Enter** to save your parameter file. (If you need to change anything later, type **2** to edit the JCL before submitting the job. Edit the JCL and then submit the job.)

Caution: Closing the JCL without submitting the job loses all of the changes you made to the parameter file.

6. Research any return code other than zero before closing the parameter file or exiting the Admin Tool.

Prepare Connect:Direct for Secure+ Option Operations

After you set up the Secure+ Option environment, you must prepare Connect:Direct to use Secure+ Option.

To set up Connect:Direct to run with Secure+ Option:

1. Add the following parameter to the Connect:Direct z/OS initialization parameters: **SECURE.DSN=*filename***, where *filename* is the name of the Secure+ parameter file for that node.
2. If you are operating in a CD/Plex environment, add the **SECURE.SSL.Path.Prefix=*prefix*** parameter, where *prefix* is the prefix location of the key database or key ring that contains the certificates for the TLS or SSL protocol.
3. Verify that the correct AP key is in the initialization parameter file.
4. Restart Connect:Direct on that node.
5. To verify that Connect:Direct Secure+ Option initialization is complete, after you restart Connect:Direct with SECURE.DSN, review the started task output for the following messages: *SITA028I Secure+ initialization* and *SITA165I Secure+ initialization complete*, if you are using the TLS or SSL protocol.

Save the Parameter File After the Initial Setup

After you save the Secure+ parameter file the first time using the Save As option, you must stop Connect:Direct before using the Save As option again, and then restart Connect:Direct.

When you are maintaining Connect:Direct Secure+ Option and want to save your changes without recycling Connect:Direct, use the Save Active option. For more information, see *Save Changes to Node Records Using the Save Active Option* on page 145.

Validate and Test Connections by Session

To validate and test a connection between two business partners, follow this general procedure. After you confirm that the secure connection has been established and that you can change your default security settings for a session, you can finalize the settings in the Secure+ parameter file of each business partner, save the files, and begin transferring data.

1. For the selected protocol, make sure all prerequisites outside of Secure+ have been taken care of, such as the obtaining of server certificates and exchanging of keys.
2. Make sure each node is defined in the partner's network map.
3. For both the local and remote nodes, specify the protocol to be used when a secure connection is required (TLS, SSL, or STS).
4. For the selected protocol, make sure to define all settings required for a successful connection in the local and remote node records in the parameter files.
5. Perform the procedures in this chapter, namely, *Save and Submit the Secure+ Parameter File* on page 131, and *Prepare Connect:Direct for Secure+ Option Operations* on page 133.
6. To test the connection, perform a file transfer between the two partners.

Once you have successfully performed a file transfer using a secure connection, you are ready to finalize the parameter files.

7. Take one of the following actions, depending on whether you want to make your sessions default to secure or non-secure:
 - ◆ To have your sessions default to secure, specify `OVERRIDE=Y` in both the local and remote node records in the parameter files of both business partners.
 - ◆ To have your sessions default to non-secure, specify `OVERRIDE=Y` in both the local and remote node records in the parameter files of both business partners. Disable the selected protocol in the remote node record.
8. To test changing your security defaults for a session, take one of the following actions depending on whether you want to make your sessions default to secure or non-secure. For a complete description of the `SECURE` parameter and how to use it in the `PROCESS` statement, go to the [Connect:Direct Processes Web site](#). Also, see *Examples of Overriding Security Settings* on page 137.
 - ◆ To make a session non-secure, specify `SECURE=OFF` in the `PROCESS` statement preceding the `COPY` statement to transfer the file.
 - ◆ To make a session secure, specify `SECURE=TLS|SSL|STS` in the `PROCESS` statement.
9. After you valid and test your connections by session, save the parameter files and restart `Connect:Direct`.

Override Settings in Connect:Direct Processes

Once you have configured the Secure+ Option environment, security is either turned on or off each time that you use Connect:Direct with a node defined in the Secure+ parameter file. However, you can override some default security settings in a remote node record from a Connect:Direct Process using the SECURE parameter in the PROCESS or COPY statement.

To allow a business partner to override the default security setting of whether security is turned on or off for another business partner and to choose the protocol for the remote node, the following conditions must be in place:

- ◆ Each business partner agrees all sessions are secure or non-secure as the default
- ◆ Each business partner agrees to allow the override of the Secure+ parameters by specifying OVERRIDE=Y for both the local and remote nodes in their Secure+ parameter file.
- ◆ The remote node definition in each Secure+ parameter file specifies the parameters necessary for a secure session even if the protocol is disabled including all information necessary for exchanging and validating each partner's identity. All parameters related to a protocol are defined, such as STS keys and algorithms or SSL/TLS cipher suites and key databases.
- ◆ Secure+ Option is active on both nodes.

Once the Secure+ parameter files for both business partners have been set up properly, you can override the default security settings on a Process-by-Process basis to perform exception processing.

For a complete description of the SECURE parameter and how to use it in the PROCESS or COPY statement, go to the [Connect:Direct Processes Web site](#). Examples are also provided on the web site.

Overriding Secure+ Defaults in the PROCESS Statement

The first statement in all Connect:Direct Processes is the PROCESS statement which defines the attributes of a Process. The SECURE keyword in the PROCESS statement allows you to perform one or more of the following functions:

- ◆ Turn on security when non-secure sessions are the default
- ◆ Select the protocol (SSL, TLS, or STS) when non-secure sessions are the default
- ◆ Specify one or more cipher suites to override the default cipher suites defined in the Secure+ parameter file
- ◆ Turn off security when secure sessions are the default (if OVERRIDE=Y is specified in the Remote Node record settings in the Secure+ parameter file).
- ◆ Encrypt only the control block information contained in Function Management Headers (FMHs), such as a user ID, password, and filename. (The default is to encrypt both the control block information and the data being transferred.)

The following syntax example shows the options available for the SECURE keyword:

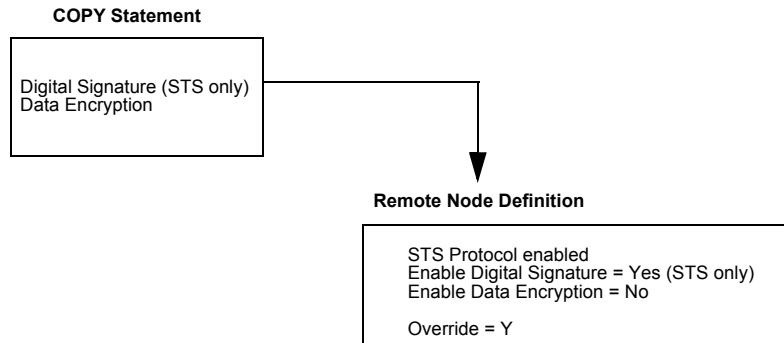
```
SECURE=OFF | STS | SSL | TLS
or
SECURE=ENCRYPT.DATA=Y|N
or
SECURE = (OFF | SSL | TLS | STS, ENCRYPT.DATA=Y|N)
or
SECURE = (OFF | SSL | TLS | STS, <cipher_suite> | (cipher_suite_list), ENCRYPT.DATA=Y|N)
```

If you use multiple SECURE subparameters, ENCRYPT.DATA must be the last (or only) value specified on the SECURE= parameter.

Overriding Secure+ Defaults in the COPY Statement

By using the COPY statement's SECURE parameter in a Connect:Direct Process to override the settings in the Secure+ parameter file and enabling the override feature in the remote node record, you can disable security for a particular file transfer. Secure+ Option uses the most secure connection available. Therefore, if the remote node record enables digital signatures (in the case of an STS environment) or encryption, the PNODE cannot turn those options off using the COPY statement override.

The following illustration shows how the COPY statement overrides the security functions set in a remote node record:



After the security settings of the PNODE and SNODE are merged, the strongest setting is always used. Therefore, the value specified from the COPY statement cannot disable data encryption or digital signatures if the SNODE has enabled them.

In an SSL or TLS environment, the following syntax example shows the options available for the SECURE keyword in a COPY statement (for the destination file that you are copying to):

```
SECURE = ENCRYPT.DATA=Y|N
OR
SECURE = ENC=Y|N
```

In an STS environment, you can also specify the algorithm to use for encryption and whether to enable digital signatures, as the following syntax example shows:

```
SECURE = (ENCRYPT.DATA=Y|N|algorithm name,SIGNATURE=Y|N)
OR
SECURE = ENC=Y|N|algorithm name,SIG=Y|N
```

Examples of Overriding Security Settings

These examples illustrate how business partners use the SECURE parameter to override the security defaults for a particular session.

Default is Secure Sessions

The business partners agree by default all sessions are secure and choose SSL as the default protocol. Both partners enable the SSL protocol in the Secure+ parameter files and specify OVERRIDE=Y in both the Local and Remote Node records. To override the default and make a particular session non-secure, they use the following PROCESS statement:

```
SSLOFF PROCESS SNODE=OTHERBP SECURE=OFF
```

Default is Non-Secure Sessions

The business partners agree by default all sessions are non-secure. When a secure communication line is required for a particular session, the non-secure default is overridden and the SSL protocol used. The Remote Node records specify `OVERRIDE=Y`, but the SSL protocol is not enabled in the Secure+ parameter files. However, all other parameters required to perform the handshake to establish an SSL session are defined in the Remote Node records. To specify that the session for this PROCESS is to be secure using SSL, the business partners use the following PROCESS statement:

```
SSLON PROCESS SNODE=OTHERBP SECURE=SSL
```

Default Cipher Suite is TLS_RSA_WITH_RC4_128_MD5

The business partners agreed by default all sessions are secure and chose TLS as the default protocol. Both partners specified the following configuration in their Secure+ parameter files:

- ◆ Enabled the TLS protocol
- ◆ Specified `OVERRIDE=Y` in both the Local and Remote Node records
- ◆ Selected `TLS_RSA_WITH_RC4_128_MD5` as the cipher suite to use when executing Processes

To override the default cipher suite and use `TLS_RSA_WITH_3DES_EDE_CBC_SHA` when executing a particular Process, they use the following PROCESS statement:

```
NEWCIPHER PROCESS SNODE=OTHERBP SECURE=(TLS,TLS_RSA_WITH_3DES_EDE_CBC_SHA)
```

Replace Single Default Cipher Suite with Cipher List

The business partners agreed by default all sessions are secure and chose TLS as the default protocol. Both partners specified the following configuration in their Secure+ parameter files:

- ◆ Enabled the TLS protocol
- ◆ Specified `OVERRIDE=Y` in both the Local and Remote Node records
- ◆ Selected `TLS_RSA_WITH_RC4_128_MD5` as the cipher suite to use when executing Processes

To override the default protocol and use a list of other TLS cipher suites when executing a particular Process, they use the following PROCESS statement:

```
NEWCIPHERS PROCESS SNODE=OTHERBP
SECURE=(TLS, (TLS_RSA_WITH_3DES_EDE_CBC_SHA, TLS_RSA_AES_128_SHA,
TLS_RSA_AES_256_SHA, TLS_RSA_WITH_DES_CBC_SHA) )
```

Default Encrypts Both Control Block Information and Data Being Sent

The business partners agreed by default to encrypt all information sent during the handshake to set up communication sessions and the actual files being transferred. Both partners specified the following configuration in their Secure+ parameter files:

- ◆ Specified ENCRYPT=Y in both the Local and Remote Node records
- ◆ Specified OVERRIDE=Y in both the Local and Remote Node records

To not go through the expense of encrypting and decrypting data being transferred, they use the following PROCESS statement when transferring a particular file:

```
ENCNO PROCESS SNODE=OTHERBP SECURE=ENCRYPT.DATA=N
```

In this scenario, both business partners are more concerned with increasing throughput and using less CPU while protecting the information being exchanged to establish the session.

Overriding Secure+ Option Settings in an STS Protocol Environment

The business partners agreed by default all sessions are none-secure and chose STS as the default protocol when secure transfers are required. Both partners specified the following configuration in their Secure+ parameter files:

- ◆ Disabled the STS protocol in the remote node record
- ◆ Specified OVERRIDE=Y in both the Local and Remote Node records

To enable data encryption and digital signatures in the following example, SAMPLE, and override the default non-secure connection, they use the following COPY statement, which copies the data set TEST.INPUT.DATASET from the PNODE to the SNODE (THE.OTHER.NODE) and renames it to TEST.OUTPUT.DATASET.

```
SAMPLE PROCESS SNODE=THE.OTHER.NODE SECURE=STS
*
COPYFILE COPY FROM ( PNODE
                        DSN= 'TEST.INPUT.DATASET'
                        DISP=SHR
                    )
                TO ( SNODE
                    DSN= 'TEST.OUTPUT.DATASET'
                    DISP= (NEW.CATLG)
                )
                SECURE= (ENC=Y, SIG=Y)
```

Maintain Secure+ Option

After you set up the Secure+ Option environment, you will need to maintain both the Secure+ parameter file and the records it contains whenever changes in your system or a trading partner's environment require updates. For SSL and TLS node records, you can also use Sterling Control Center to display and modify information, and to delete nodes.

After you make changes, refer to *Save Changes to Node Records Using the Save Active Option* on page 145 to put the updates in effect immediately.

Maintaining the Secure+ Parameter File

The File Menu on the Secure+ Admin Tool: Main Screen contains options for maintaining the parameter file. To open this menu, select **File** from the action bar and press **Enter**. The following panel sample shows the available options when a parameter file is already open.

Open a Secure+ Parameter File

Before you can modify node records, you must open the parameter file that contains these records.

To open a Secure+ parameter file:

1. With the **Secure+ Admin Tool Main Screen** open, select **File** and press **Enter**:
2. Type **2** to select **Open** and press **Enter** to display the file selection screen:

```

Secure+ Admin Tool: File Selection

Enter file name for: INPUT SECURE PARM FILE

File
Name: $CD.SECURE.PARMPFILE                               Browse

File System Type:
1 1. MVS  2. HFS                                         Cancel
    
```

3. Type the parameter file name prefix or partial prefix followed by an asterisk (*), select **Browse**, and press **Enter**. The following screen is displayed:

```

Secure+ Admin Tool: File Selection                               Row 1 of 3

Option: _____ Scroll CSR

Enter "S" on the line of the file for for MVS.

LC Filename or Directory
S $CD.PARMPFILE
- $CD.PARMPFILE.DATA
- $CD.PARMPFILE.INDEX
***** Bottom of data *****
    
```

Note: You can also type the complete parameter file name and press **Enter**.

4. Type **S** next to the file name to open and press **Enter**. The **Secure+ Admin Tool: Main Screen** displays nodes populated from the parameter file you opened. See *Maintaining Node Records* on page 146 for more information on this panel and how to select the available options.

View Information about the Secure+ Parameter File

To view information about the Secure+ parameter file:

1. Open the Admin Tool.
2. Select **File** and press **Enter**.
3. Type **4** to select **Info**. The **File Information Panel** is displayed:

```

+----- Secure+ Admin Tool File Information Panel -----+
**                               Secure+ Admin Tool File Information Panel
***
****
***
** *                               Secure+ Admin for Connect:Direct for z/OS
** *
* *                               Node: CSG.PROD390
* *                               Name Filter Applied: *
* *                               File: $CD.SECURE.PARMPFILE
* **                               $CD.SECURE.ACCESS
* **                               Update           Current
***                               Events:           0   Records:           9
****
***                               Toolkit msg/Rc:CSPA000I/   0
**                               Last 3
                               Events:

```

The fields in the **File Information Panel** are described in the following table:

Field Name	Description
Node	The name of the local node for the parameter file that is open.
Admin Version	The version of Secure+ Admin Tool being used.
Name Filter Applied	Name of the filter used to determine which remote node records to display.
File	The name of the current parameter file and the access file.
Update Events	Number of updates to the parameter file.
Current Records	Total number of remote node records.
Toolkit msg/Rc:CSPA0001	Message ID of the last Toolkit call.
Last 3 Events	List of the last 3 updates.

Save Changes to Node Records Using the Save Active Option

The **Save Active** option on the Secure+ Admin Tool **File** menu enables you to dynamically save changes to all records in an existing Secure+ parameter file.

You must use the Save As option the first time you create the parameter file. This option deletes, defines, and reloads the Secure+ parameter file. See *Save and Submit the Secure+ Parameter File* on page 131.

Note: Connect:Direct must be running in order to use the **Save Active** option. Any changes made to node records take effect immediately.

To save the Secure+ parameter file and put any changes made in effect immediately:

1. Open the Admin Tool.
2. Select **File** and press **Enter**.
3. Type **6** to select **Save Active**.
4. Read all warning and error messages. You can continue without resolving warning messages, but you must resolve all errors before you save the parameter file.

Caution: Because Connect:Direct cannot restrict multiple users from attempting to apply updates to the same parameter file, measures should be in place to ensure that only one user is accessing a particular parameter file at one time. If precautions are not taken, updates will be lost and conflicts among multiple user interfaces will occur.

Resecure the Secure+ Parameter File and Access File

Routinely, or if your passphrase is compromised, you should resecure the Secure+ parameter and access files. You must open a parameter file before you perform this procedure.

To resecure the Secure+ parameter file and access file:

1. From the **Secure+ Option Admin Tool Main Screen**, select **File** and press **Enter**.
2. Type **7** to select **Save As** and press **Enter**.
3. If any warning messages are displayed, read them and press **F3** to close the warning panel.
4. On the **File Selection** panel, the file name of the parameter file that you have open is displayed. Press **Enter**.
5. At the confirmation prompt, select **OK**. The old parameter file is deleted and a new parameter file with the same name is created.
6. On the **Save As** screen, type **2** to select **Edit**, then select **Make Pass Phrase** and press **Enter**.
7. Select **OK** to confirm that you want to create a new passphrase.
8. Type a 32-byte string, using uppercase, lowercase, numeric, and alphabetic characters.
9. On the **Save As** panel, type **3** to select **Submit** and press **Enter**.
10. Select **OK** to submit the job.
11. When the *Job Submitted* message is displayed, press **Enter**.

12. Verify that the job completed with a return code of zero before closing the parameter file or exiting the Secure+ Admin Tool. Research any return codes other than zero.

Maintaining Node Records

After you start the Secure+ Admin Tool, the main screen is displayed:

```

File  Edit  Key Management  Help
-----
                                Row 9 to 13 of 13
Q2A.ZOS.PARIS      Secure+ Admin Tool: Main Screen
Option ==>>>                                Scroll CSR

                                Table Line Commands are:

E Export pub. key      H View History          D Delete node
U Update node          I Insert node

                                Secure
LC Node Name          Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
Q2A.ZOS.PARIS        L    NNNN   Y      N      N      N      N
Q3A.ZOS.PARIS        R    ***N   N      *      *      *      *
Q3B.ZOS.PARIS        R    ***N   N      *      *      *      *
SOL36SP              R    ***N   N      *      *      *      *
W2S.4200.CDWOPS8    R    ***N   N      *      *      *      *
*****
***** BOTTOM OF DATA *****

```

The following table line commands are available to use on the nodes currently displayed:

- ◆ **E—Export** allows you to export the STS keys for the selected STS node only. For more information on exporting STS keys for all nodes or a single node, see *Export STS Keys* on page 124 (STS nodes only).
- ◆ **U—Update** allows you to perform the following functions:
 - ◆ *Disable Secure+ Option* (all nodes)
 - ◆ *Change the Cipher Suites* (TLS or SSL nodes only)
 - ◆ *Change the Encryption Algorithm Names* (STS nodes only)
- ◆ **H—History** lists the individual dates including the time the selected node was updated.
- ◆ **I—Insert** allows you to create a new remote node record. For more information, see *Insert a Node* on page 149.
- ◆ **D—Delete** allows you to delete the selected node. For more information, see *Delete a Secure+ Option Remote Node Record* on page 149.

Disable Secure+ Option

To disable Secure+ Option on a SSL or TLS node:

1. From the **Secure+ Admin Tool Main Screen**, type **U** next to the SSL or TLS node to update and press **Enter**. The **Secure+ Create/Update Panel** displays the information for the selected node.
2. Type **2** beside the **Enable SSL** and **Enable TLS** fields. Select **OK** and press **Enter**.
3. To reselect the same node, type **U** next to the node to update and press **Enter**. Select **STS Parameters** in the panel selection line and press **Enter**. Type **2** beside the **Enable STS** field.
4. Select **OK** and press **Enter**.
5. Save the Secure+ parameter file using the procedure in *Save Changes to Node Records Using the Save Active Option* on page 145.

Note: To continue Connect:Direct operations with Secure+ Option disabled, *both* trading partners must disable Secure+ Option.

To disable Secure+ Option on an STS node:

1. From the **Secure+ Admin Tool Main Screen**, type **U** next to the STS node to update and press **Enter**. The **Secure+ Create/Update Panel** displays the information for the selected node.
2. Type **2** beside the **Enable STS** field.
3. Select **OK** and press **Enter**.
4. To reselect the same node, type **U** next to the node to update and press **Enter**. Select **SSL/TLS Parameters** in the panel selection line and press **Enter**.
5. Type **2** beside the **Enable SSL** and **Enable TLS** fields. Select **OK** and press **Enter**.
6. Save the Secure+ parameter file using the procedure in *Save Changes to Node Records Using the Save Active Option* on page 145.

Change the Cipher Suites

When you activate the SSL or the TLS protocol for a node, cipher suites are used to encrypt transmitted data. The same cipher suite must be defined at both ends of the transmission. Secure+ Option searches the enabled cipher suite list and locates the first cipher suite that is common for communications at both the PNODE and the SNODE. It then uses this cipher suite to encrypt data. You defined cipher suites when you configured the local node record.

To change the cipher suites enabled for a node and the priorities assigned to them:

1. From the **Secure+ Admin Tool Main Screen**, type **U** next to the node to update.
2. On the Create/Update Panel, select the **Cipher Suites** field and press **Enter** to display the Update Cipher Suites panel.

```

More: +
Update the order field below to enable and order cipher suites.

O
r
d
e
r  All Available Cipher-Suites          Enabled Cipher-Suites

==  =====
1  SSL_RSA_AES_128_SHA                  SSL_RSA_AES_128_SHA
2  SSL_RSA_AES_256_SHA                  SSL_RSA_AES_256_SHA
3  SSL_RSA_WITH_3DES_EDE_CBC_SHA        SSL_RSA_WITH_3DES_EDE_CBC_SHA
4  SSL_RSA_WITH_DES_CBC_SHA             SSL_RSA_WITH_DES_CBC_SHA
5  SSL_RSA_EXPORT_WITH_RC2_CBC_40_MD5   SSL_RSA_EXPORT_WITH_RC2_CBC_40_MD5
6  SSL_RSA_WITH_RC4_128_SHA             SSL_RSA_WITH_RC4_128_SHA
7  SSL_RSA_WITH_RC4_128_MD5            SSL_RSA_WITH_RC4_128_MD5
8  SSL_RSA_EXPORT_WITH_RC4_40_MD5       SSL_RSA_EXPORT_WITH_RC4_40_MD5
9  SSL_RSA_WITH_NULL_SHA                SSL_RSA_WITH_NULL_SHA
10 SSL_RSA_WITH_NULL_MD5                SSL_RSA_WITH_NULL_MD5
11 DEFAULT_TO_LOCAL_NODE                DEFAULT_TO_LOCAL_NODE

```

The list on the left side contains all available cipher suites. The active cipher suites are listed on the right side of the screen and are assigned a numerical order in the **Order** column on the left side of the screen.

3. Take one or more of the following actions as needed:
 - ◆ Type **1** by the cipher you want to enable and give the highest priority. Type **2** by the cipher suite you want to enable and place second in priority. Continue typing numbers next to the ciphers you want to enable, in order of priority. The ciphers you enable appear in the order of priority in the **Enabled Cipher-Suites** list.
 - ◆ To deactivate a cipher suite, clear the number in the **Order** field and press **Enter**.
 - ◆ To change the order of a cipher suite, type new numbers in the **Order** fields of the cipher suites to reorder and press **Enter**.
4. Press **PF3** to save the new enabled cipher-suite list and return to the **Secure+ Create/Update Panel**.
5. Save the parameter file using the procedure described in *Save Changes to Node Records Using the Save Active Option* on page 145.

Change the Encryption Algorithm Names

When you activate the STS protocol for a node, it uses algorithms to encrypt the data being transmitted. A common algorithm must exist on both endpoints of the transmission.

Secure+ Option searches the enabled algorithm list and locates the first algorithm that is common for communications at both nodes.

To enable algorithms for a node record:

1. From the **Secure+ Admin Tool Main Screen**, type **U** next to the node to update.

The **Algorithm Names** field specifies the algorithms being used. They are listed from left to right in priority order.

2. Edit or rearrange the algorithm names by typing over the existing names.
3. Select **OK** and press **Enter**.
4. Save the parameter file using the procedure described in *Save Changes to Node Records Using the Save Active Option* on page 145.

Insert a Node

The Insert node option follows the same procedure as creating a remote node manually using the Edit menu.

To create a remote node:

1. To create a remote node, type **I** next to any node on the Secure+ Admin Tool Main Screen and press **Enter**.
2. Type the name of the existing remote node that corresponds to its name in the network map and enter the rest of the information on the **Node Identification panel**. The **Secure+ Create/Update Panel** is displayed.
3. If you are creating a node record for the SSL or TLS protocol, go to step 4 on page 62 and proceed from there. If you are creating a node record for the STS protocol, go to step 4 on page 76 and proceed from there.
4. After you finish the procedure and saved the record by selecting **OK** and pressing **Enter**, save the parameter file using the procedure described in *Save Changes to Node Records Using the Save Active Option* on page 145.

Delete a Secure+ Option Remote Node Record

If you remove a remote node record from the network map in Connect:Direct, you can also remove it from the Secure+ parameter file. This process deletes nodes from the Secure+ parameter file.

Note: If you have implemented the Strong Password Encryption feature, you cannot use the Delete node table line command to delete the .PASSWORD record. To determine if SPE is in effect, see *Disabling Strong Password Encryption* on page 91 for instructions on how to access the SPE Parameters screen. You will see either the message, *SPE not currently in use* or *SPE currently in use* on that panel. If SPE is not in use, you can delete the .PASSWORD record; otherwise, follow the procedure to disable SPE, and then reinitialize Connect:Direct. This initialization with SPE disabled will remove any SPE encryption that has been previously applied.

To delete a remote node record:

1. Type **D** next to the node to delete and press **Enter**.
The Secure+ Option **Confirmation Prompt** displays the message *Are you sure you want to delete 'selected node'?*
2. Select **OK** and press **Enter** to delete the record.
3. Save the Secure+ parameter file using the procedure in *Save Changes to Node Records Using the Save Active Option* on page 145.

Modify STS Keys

If you are using STS protocol with Secure+ Option, you periodically need to update and clear keys. This section provides the following procedures for modifying key files for Secure+ Option.

- ◆ Updating keys in node records configured for STS
- ◆ Resetting keys in node records configured for STS

Update Keys in Node Records Configured for the STS Protocol

In order to maintain communications with a trading partner when you update your keys, you must maintain a copy of the previous keys until your trading partner receives the updated keys. You must perform this procedure for both the local node record and remote nodes that use the STS protocol.

To update your signature and authentication keys:

1. From the **Secure+ Admin Tool Main Screen**, type **U** next to the node record name (local or remote) to update and press **Enter**.
2. From the **Secure+ Create/Update Panel - STS Parameters** panel select **Create Auth. Prev. Keys**.

Secure+ Option copies the current authentication keys to previous keys and assigns an expiration date of 30 days from the time that you generated the previous key value. Only the expiration date and time are displayed on the screen. The previous keys are stored internally.

3. To change the expiration date, position your cursor in the date field and change any of the information.
4. Select **Create Sig. Prev. Keys**.

Secure+ Option copies the current signature keys to previous keys and assigns an expiration date of 30 days from the time that you generated this previous key value. Only the expiration date and time are displayed on the screen. The previous keys are stored internally.

5. To change the expiration date, position your cursor in the date field and change any of the information.
6. Select **Create Auth. Pubkey** and press **Enter**.

7. On the **Generate Seed** screen, type **2** to select **Sample Value** and press **Enter**.

8. Change some of the text by typing over it. Press **PF3**.

When the **Secure+ Create/Update Panel** displays the message *Seed generation complete*, your public key for strong authentication is created.

9. Select **Create Sig. Pubkey** and press **Enter**.
10. On the **Generate Seed** screen, type **2** to select **Sample Value** and press **Enter**.
11. Change some of the text by typing over it. Changing the text creates a secure key value. Press **PF3**.

- When the **Secure+ Create/Update Panel** displays the message *Seed generation complete*, your public key for digital signatures is created.
12. Select **OK** and press **Enter**.
 13. Save the Secure+ parameter file using the procedure in Chapter 11, *Enable and Validate Secure+ Operation*.

Reset Keys in Remote Node Records Configured for STS

To reset the keys in remote node records to default to the settings in the local node record.

1. From the **Secure+ Admin Tool Main Screen**, type **U** next to the node to update and press **Enter**.
2. From the **Secure+ Create/Update Panel - STS Parameters**, select **Reset Auth. Pubkey** and press **Enter**.

This step resets the public key used for strong authentication to the default value in the local node record.

3. From the **Secure+ Create/Update Panel - STS Parameters**, select **Reset Sig. Pubkey** and press **Enter**.

This step resets the public key used for the digital signature to the default value in the local node record.

Secure+ Option Statistics

Connect:Direct logs statistics for Connect:Direct Process activity. The Connect:Direct statistics include Secure+ Option information for a Process.

The following samples of Connect:Direct Process statistics records contain information for Secure+ Option support. For information about viewing Connect:Direct for z/OS Process statistics, refer to the *Connect:Direct for z/OS User's Guide*.

SSL or TLS Statistics Record

When you use the **Select Statistics** command to view the information about a Connect:Direct Process that uses SSL or TLS security, you see a screen similar to the following. (Be sure to use the Display option by typing **D** on the CMD line to see the detailed version of the SELECT STATISTICS report.) The Secure+ Option fields are in bold. A description for the fields follows the samples.

```

=====
CD.OS390.V40000                SELECT STATISTICS                05/04/2010
=====
Function      => PROCESS SUBMIT          Start Time => 19:30:00
Process Name  => OS3903                  Stop Time  => 19:30:00
Process Num   => 1                      Comp Code  => 00000000
                                           Comp Msg   => SSPA001I

Userid       => JWHITE
Primary Node  => SC.DUB.JWHITE          Step Name  =>
Submitted DSN=> JWHITE.NDM.PROCESS.LIB(OS3903)

-----
Function      => COPY                    Start Time => 19:30:14
Process Name  => OS3903                  Stop Time  => 19:30:14
Process Num   => 1                      Comp Code  => 00000000
                                           Comp Msg   => SCPA000I

Userid       => JWHITE
Secondary Node=> SC.DUB.JWHITE          Step Name  => PUSH01
Other addr   => 10.20.201.2
Other port    => 04399

V2 Buffer Size      => 65,536
Negotiated V2 Buffer Size => 65,536
TCP Buffer Size Used => 262,144
Session Protocol =  TCP
CRC Requested
CRC Not Performed

TLS Enabled           => Yes
TLS Ciphersuite      => TLS_RSA_AES_256_SHA
Subject => (SN=47:b2:1a:10:00:0e:07:72/C=US/ST=a/L=t/O=g/CN=mikey3/)
Issuer  => (C=US/ST=a/L=t/O=g/CN=mikey3/)

**** CHECKPOINTED; Interval => 1,000
From ( Pnode
Dsn=>JWHITE.TCPIP.DATA.FILE)
  recs => 0                      blks => 1
  I/O BYTES => 266
  VTAM BYTES => 53
  Cmpr Perc => 80.1%
  VOL=SER=> USER17
To ( Snode
Dsn=>JWHITE.FTST.AA1030B)
  recs => 0                      blks => 1
  I/O BYTES => 266
  VTAM BYTES => 53
  Cmpr Perc => 80.1%
  VOL=SER=> WRKPK3

```

The following statistics are displayed for the copy function:

Field	Description	Valid Values
TLS (or SSL) Enabled	Specifies whether TLS (or SSL) x.509 certificate use is enabled.	Yes No
TLS (or SSL) Ciphersuite	Specifies the cipher suite used in the session and whether the Process defined an override. Note: If ENCRYPT.DATA=N was in effect, NONE appears in this field.	Any valid cipher suite
Subject	Specifies the subject name on the certificate.	Any valid subject name
Issuer	Specifies the issuer name on the certificate.	Any valid issuer name

SSL or TLS Extended Option Statistics Record

When you use the **Select Statistics** command with the extended option enabled to view the information about a Connect:Direct Process that uses SSL or TLS security, you see a screen similar to the following. The Secure+ Option fields are in bold. A description for the fields follows the sample.

```

=====
CD.OS390.V40000                SELECT STATISTICS                05/04/2010
=====

Function => PROCESS SUBMIT          Start Time => 10:26:32
Process Name => STATSAMP             Stop Time  => 10:26:32
Process Num => 338                   Comp Code  => 00000000
                                       Comp Msg   => SSPA001I

Userid   => $CD
Primary  => CD.OS390.V40000          Step Name  =>
Submitted DSN=> $CD.CD.PROCESS(SUB1)

-----
Function   => Session Begin         Start Time => 19:30:14
                                                Start Date => 2008.11.17
Process Name => OS3903
Process Num => 1                     Comp Code  => 00000000
                                       Comp Msg   => SVTM055I

Userid     => JWHITE
Primary Node => SC.DUB.JWHITE
Secondary Node => SC.DUB.JWHITE2
Submitter Node => SC.DUB.JWHITE3

TLS Enabled => Yes
TLS Ciphersuite => TLS_RSA_AES_256_SHA
Subject => (SN=47:b2:1a:10:00:0e:07:72/C=US/ST=a/L=t/O=g/CN=mikey3/)
Issuer => (C=US/ST=a/L=t/O=g/CN=mikey3/)

Session Protocol = TCP
Socket for Origin => 04199 ; 10.20.201.2
Socket for Destination => 04399 ; 10.20.201.2
Bind Attempts => 0
Remote Node Communications Address => 10.20.201.2

-----
Function   => COPY                  Start Time => 19:30:14
Process Name => OS3903             Stop Time  => 19:30:14
Process Num => 1                   Comp Code  => 00000000
                                       Comp Msg   => SCPA000I

Userid     => JWHITE
Secondary Node => SC.DUB.JWHITE    Step Name  => PUSH01
Other addr => 10.20.201.2
Other port => 04399

V2 Buffer Size => 65,536
Negotiated V2 Buffer Size => 65,536
TCP Buffer Size Used => 262,144
Session Protocol = TCP
CRC Requested
CRC Not Performed

TLS Enabled => Yes
TLS Ciphersuite => TLS_RSA_AES_256_SHA
Subject => (SN=47:b2:1a:10:00:0e:07:72/C=US/ST=a/L=t/O=g/CN=mikey3/)
Issuer => (C=US/ST=a/L=t/O=g/CN=mikey3/)

**** CHECKPOINTED; Interval => 1,000
From ( Pnode
Dsn=>JWHITE.TCPIP.DATA.FILE)
  recs => 0                               blks => 1
  I/O BYTES => 266
  VTAM BYTES => 53
  Cmpr Perc => 80.1%
  VOL=SER=> USER17
To ( Snode
Dsn=>JWHITE.FTST.AA1030B)
  recs => 0                               blks => 1
  I/O BYTES => 266
  VTAM BYTES => 53
  Cmpr Perc => 80.1%
  VOL=SER=> WRKPK3

```

The following fields are included for the Connect:Direct for z/OS extended option statistics:

Field	Description	Valid Values
TLS (or SSL) Enabled	Specifies whether TLS (or SSL) x.509 certificate use is enabled.	Yes No
TLS (or SSL) Ciphersuite	Specifies the cipher suite used in the session and whether the Process defined an override.	Any valid cipher suite
Subject	Specifies the subject name on the certificate.	Any valid subject name

Field	Description	Valid Values
Issuer	Specifies the issuer name on the certificate.	Any valid issuer name

STS Statistics Record

When you use the **Select Statistics** command to view the information about a Connect:Direct Process that uses STS security, you see a screen similar to the following. The Secure+ Option fields are in bold. A description for the fields follows the samples.

```

=====
CD.OS390.V40000                SELECT STATISTICS                05/04/2010
=====

Function => PROCESS SUBMIT          Start Time => 10:26:32
Process Name => STATSAMP             Stop Time  => 10:26:32
Process Num => 1                     Comp Code  => 00000000
                                       Comp Msg   => SSPA001I

Userid   => $CD
Primary  => CD.OS390.V40000         Step Name  =>
Submitted DSN=> $CD.CD.PROCESS(SUB1)

-----

Function   => COPY                  Start Time => 10:26:35
Process Name => STATSAMP             Stop Time  => 10:26:39
Process Num => 338                  Comp Code  => 00000000
                                       Comp Msg   => SCPA000I

Userid     => $CD
Secondary  => SC.OS390.V40000       Step Name  => COPYFIL1

From ( Pnode
Dsn=$CD.SECURE.TESTFILE)
  recs => 0                          blks => 157
  I/O BYTES => 5,000,000
  VTAM BYTES => 5,000,314
  Cmpr Perc => 0.0%
  Digital Signature enabled = Yes
  VOL=SER=> USER05
To ( Snode
Dsn=$CD.SECURE.RESTART.OUT1)
  recs => 0                          blks => 157
  I/O BYTES => 5,000,000
  VTAM BYTES => 5,000,314
  Cmpr Perc => 0.0%
  Digital Signature enabled = No
  Merged Signature enabled = Yes
  Merged Encryption enabled = No
  Verified Signature = Curr
  VOL=SER=> USER02

```

The following fields are included for the Connect:Direct statistics for the copy function:

Field	Description	Valid Values
Digital Signature enabled	Specifies whether digital signature are enabled in the Secure+ parameter file. This information is displayed in the statidtics sepately for the PNODE and SNODE.	Yes No
Merged Signature enabled	Specifies the resulting value of the merge between the PNODE and the SNODE Secure+ parameter files and the COPY statement parameters for digital signature.	Yes No

Field	Description	Valid Values
Merged Encryption enabled	Specifies the resulting value of the merge between the PNODE and the SNODE Secure+ parameter files and the COPY statement parameters for data encryption.	Algorithm ID No
Verified Signature	Specifies whether the current or previous key verified the digital signature of digital signature is enabled for the session. This information is displayed only if the merged signature value is Yes .	Curr Prev

STS Extended Option Statistics Records

When you use the **Select Statistics** command with extended option enabled to view the information about a Connect:Direct Process that uses STS security, you see a screen similar to the following. The Secure+ Option fields are in bold.

Note: You must type * in the RECORD field of the Select Statistics Extended Options screen to ensure that the complete information is displayed.

A description of the fields follows the samples.

```

=====
CD.OS390.V40000                SELECT STATISTICS                05/04/2010
=====

Function => PROCESS SUBMIT          Start Time => 10:26:32
Process Name => $STATSAMP           Stop Time  => 10:26:32
Process Num => 338                  Comp Code  => 00000000
                                       Comp Msg   => SSPA001I

Userid    => $CD
Primary   => CD.OS390.V40000        Step Name  =>
Submitted DSN=> $CD.CD.PROCESS (SUB1)

-----

Function    => Session Begin        Start Time => 10:26:35
Process Name => $STATSAMP           Start Date => 05/04/2003
Process Num => 338                  Comp Code  => 00000000
                                       Comp Msg   => SVTM055I

Userid      => $CD
Primary Node => CD.OS390.V40000
Secondary Node CD.OS390.V40000      ALT.NODE => CD.DALLAS.OFFICE
Submitter Node CD.OS390.V40000

Node Signature Enabled = No
Snode Signature Enabled = No
Merged Signature Enabled = No
Node Encrypt.Data Algorithms...
  IDEACBC128
  TDESCBC112
  DESCBC56
Snode Encrypt.Data Algorithms...
  IDEACBC128
  TDESCBC112
  DESCBC56
System Data Encryption = IDEACBC128

-----

Function    => COPY STEP START      Time       => 10:26:35
Process Name => $STATSAMP           Process Num => 338
Primary Node => CD.OS390.V40000     Secondary Node => SC.OS390.V40000
Node Signature Enabled = No
Snode Signature Enabled = No
Merged Signature Enabled = No
Node Encryption Enabled = No
Snode Encryption Enabled = No
Encryption Enabled = No

-----

Function    => COPY                 Start Time => 10:26:35
Process Name => $STATSAMP           Stop Time  => 10:26:39
Process Num => 338                  Comp Code  => 00000000
                                       Comp Msg   => SCPA000I

Userid      => $CD
Secondary    => SC.OS390.V40000     Step Name  => COPYFILL1

From ( Pnode
Dsn=$CD.SECURE.TESTFILE)
  recs => 0                          blks => 157
  I/O BYTES => 5,000,000
  VTAM BYTES => 5,000,314
  Cmpr Perc => 0.0%
  Digital Signature enabled = Yes
  VOL=SER=> USER05
To ( Snode
Dsn=$CD.SECURE.RESTART.OUT1)
  recs => 0                          blks => 157
  TYPE      => LRGSAM
  I/O BYTES => 5,000,000
  VTAM BYTES => 5,000,314
  Cmpr Perc => 0.0%
  Digital Signature enabled = No
  Merged Signature enabled = Yes
  Merged Encryption enabled = No
  Verified Signature = Curr
  VOL=SER=> USER02

```

Session Begin (SB) Record

The following fields are included for the Connect:Direct extended option statistics in the Session Begin record:

Field	Description	Valid Values
Pnode Signature Enabled	Specifies whether digital signatures are enabled in the Secure+ parameter file of the PNODE.	Yes No
Snode Signature Enabled	Specifies whether digital signatures are enabled in the Secure+ parameter file of the SNODE.	Yes No
Merged Signature Enabled	Specifies the resulting value of the merge between the PNODE and SNODE Secure+ parameter files for digital signature.	Yes No
Pnode Encrypt.Data Algorithms	Specifies the available encryption algorithms of the PNODE. The algorithms are displayed in the priority order set by the Secure+ Option administrator of the PNODE.	Algorithm ID List
Snode Encrypt.Data Algorithms	Specifies the available encryption algorithms of the SNODE. The algorithms are displayed in the priority order set by the Secure+ Option administrator of the SNODE.	Algorithm ID List
System Data Encryption	Specifies the algorithm used for encrypting Connect:Direct control blocks during security-enabled transfers.	Algorithm ID

Copy Step Start (CI) Record

The following fields are included for the Connect:Direct extended option statistics in the Copy Step Start record shown on page 158.

Field	Description	Valid Values
Pnode Signature Enabled	Specifies whether digital signature is enabled in the Secure+ parameter file of the PNODE.	Yes No
Snode Signature Enabled	Specifies whether digital signature is enabled in the Secure+ parameter file of the SNODE.	Yes No
Merged Signature Enabled	Specifies the resulting value of the merge between the PNODE and SNODE Secure+ parameter files for digital signature.	Yes No
Pnode Encryption Enabled	Specifies whether data encryption is enabled in the Secure+ parameter file of the PNODE.	Yes No
Snode Encryption Enabled	Specifies whether data encryption is enabled in the Secure+ parameter file of the SNODE.	Yes No

Field	Description	Valid Values
Encryption Enabled	Specifies the resulting value of the merge between the PNODE and SNODE Secure+ parameter files for data encryption.	Algorithm ID No

Copy Termination (CT) Record

The following fields are included for the Connect:Direct extended option statistics in the Copy Termination record as shown on page 158.

Field	Description	Valid Values
Digital Signature enabled	Specifies whether digital signatures are enabled in the Secure+ parameter file. This information is displayed in the statistics separately for the PNODE and the SNODE.	Yes No
Merged Signature enabled	Specifies the resulting value of the merge between the PNODE and SNODE Secure+ parameter files <i>and</i> the COPY statement parameters for digital signature.	Yes No
Merged Encryption enabled	Specifies the resulting value of the merge between the PNODE and SNODE Secure+ parameter files <i>and</i> the COPY statement parameters for data encryption.	Algorithm ID No
Verified Signature	Specifies whether the current or previous key verified the digital signature if digital signature is enabled for the session. This information is displayed only if the merged signature value is Yes .	Curr Prev

Troubleshooting

Use the following table to help troubleshoot problems with Secure+ Option:

Note: For all errors related to Strong Password Encryption, see *Troubleshooting Possible SPE Problems* on page 92.

Problem	Possible Cause	Solution
The following message is received at startup: SITA166I or SITA167I Secure+ SSL or TLS initialization failed. rc=00000134, rs=NO DFLT UNIX PATH.	The Connect:Direct system does not have a default directory created for it in UNIX system services. The DLL files and other facilities related to SSL or TLS require the presence of a default UNIX directory.	Contact your z/OS system programmer.
The following message is received at startup: SITA166I Secure+ SSL or TLS initialization failed. rc=000000002, rs=GSK_KEYFILE_OPEN_ FAILED.	The Secure+ parameter file, in combination with the SECURE.SSL.PATH.PREFIX initialization parameter, specifies a nonexistent key database, the key database has incorrect file permissions, or the password typed is incorrect.	Correct the name specified in the initialization parameter or the Secure+ parameter file, the UNIX permissions, or the password.

Problem	Possible Cause	Solution
<p>The following message is received at startup or when Connect:Direct performs a certificate validation check and discovers a certificate that will soon expire: CSPA600W WARNING Cert: &cert for Node: &node expires: &date. The named certificate will expire on the specified date.</p> <p>Note: A message will not contain the node name if the certificate did not have a Secure+ parameter file record associated with it.</p>	<p>The warning message will appear based on the validation check controlled by the following initialization parameters, CHECK.CERT.EXPIRE, CHECK.CERT.WARN.DAYS and CHECK.CERT.EXPIRE.TIME.</p>	<p>Take the appropriate action to generate or obtain a new certificate.</p>
<p>The following message is received at startup or when Connect:Direct performs a certificate validation check and discovers that a certificate has expired: CSPA601E ERROR Cert: &cert for Node: &node expired on: &date. The named certificate has expired on the specified date.</p> <p>Note: A message will not contain the node name if the certificate did not have one associated with it.</p>	<p>The warning message will appear based on the validation check controlled by the following initialization parameters, CHECK.CERT.EXPIRE.</p>	<p>Take the appropriate action to generate or obtain a new certificate.</p>

Problem	Possible Cause	Solution
<p>The following message is received at startup or when Connect:Direct performs a certificate validation check and discovers a certificate it cannot validate: CSPA607W WARNING Cert: &cert for Node: &node does not exist. The Certificate Expiration Validation function has obtained a Certificate label for the Secure Parmfile however that certificate can not be retrieved.</p> <p>Note:A message will not contain the node name if the certificate did not have one associated with it.</p>	<p>The most likely cause of this is the certificate does not exist in the Key database or Key ring.</p>	<p>Ensure that the Certificate exist and that the Secure Parmfile entry specifies the correct label name. The label is case sensitive and must match exactly.</p>
<p>The following message is received when an SSL or TLS Process is run: SSL or TLS handshake failure, reason= GSK_ERROR_SOCKET_CLOSED.</p>	<p>The trading partners have not enabled a matching cipher suite.</p>	<p>Update the remote node record for the trading partner to enable a cipher suite recognized by the trading partner and resubmit the Process.</p>
<p>The following message is received: CSPA202E SSL handshake failure, reason=GSK_ERROR_BAD_CERTIFICATE.</p>	<p>The certificate is not valid on the system issuing GSK_ERROR_BAD_CERT. This error occurs if the certificate is not validated on any local trusted CA certificate.</p> <p>This error is common if you use self-signed certificates because the remote Connect:Direct system does not have the CA certificate.</p>	<p>Verify that each trading partner can validate the certificates of other trading partners and resubmit the Process.</p> <p>Ensure that the remote node record for the trading partner has enabled the correct protocol.</p>
<p>The following error is received from the SNODE: CSPA202E SSL or TLS handshake failure, reason=GSK_ERROR_UNKNOWN_ERROR.</p>	<p>A conflict within the IBM System SSL toolkit occurred because a certificate being processed did not use version 3 of the toolkit.</p>	<p>Ensure that all certificates and CA certificates are using version 3.</p>

Problem	Possible Cause	Solution
Secure+ Option features are enabled in the Secure+ parameter file, but the statistics record indicates that these functions are disabled.	<p>The Connect:Direct network maps do not contain entries for the PNODE and SNODE.</p> <p>The node that you are connecting with is a V1 flow (such as LU0 or Netex). Secure+ Option is not supported for V1 flows because of reliance on XDR support.</p>	<p>Verify that the network map entries for both the PNODE and the SNODE exist, and use a V2 protocol such as LU6.2, TCP/IP, or UDT.</p> <p>Check for the existence of the extended statistics record for <i>Session Begin</i> (the SB record). This record is only created in V2 flows. The absence of this record indicates V1 flows were used.</p>
Secure+ Option parameters specified from the copy statement cause the copy step to fail with message CSPA077E.	<p>The node that you are connecting with is a V1 flow (such as LU0 or Netex). Secure+ Option is not supported for V1 flows because of reliance on XDR support.</p>	<p>Check for the existence of the extended statistics record for Session Begin (the SB record). This record is only created in V2 flows. The absence of this record indicates V1 flows were used.</p>
An error occurs in ESTAE with a bad return code (RC=3) when running a Process with a remote node and the Process fails.	<p>The value for Secure+ Option Export version is incorrect in the remote node definitions for one or both of the nodes. If one node is EXPORT and the other node is NOT EXPORT, the elliptic curves that enable you to create keys and generate Diffie-Hellman shared secrets are not correct.</p>	<p>Verify that the remote node definitions on both sites accurately state the Secure+ Option Export information.</p>
Running a Process with a remote node fails with an authentication error.	<p>Unique public/private key pairs are generated for the remote node record and the local node record is set to OVERRIDE=N.</p>	<p>Change the local node record to OVERRIDE=Y or do not use unique public/private key pairs in the remote node record.</p>
The Save Active option is not selectable.	<p>You can only use the Save Active function once each time you open the Secure+ parameter file.</p>	<p>Reopen the Secure+ parameter file to use the Save Active function or use the Save As function.</p>

Problem	Possible Cause	Solution
<p>The text entry fields on the Create/Update panel of the Secure+ Option Admin Tool are not visible.</p>	<p>The CUA attributes in your ISPF profile are not set correctly.</p>	<p>Change the value for Normal Text entry in the CUA attributes of the ISPF profile to uscore in the Highlight column.</p>
<p>The Secure+ Option parameter, ENCRYPT.DATA specified from the copy statement causes the copy step to fail with an error message CSPA080E.</p>	<p>The algorithm name used in the COPY statement is not in the supported algorithm list for both nodes.</p>	<p>Verify that the algorithm name in the copy statement is in the supported algorithm list for both nodes.</p>
<p>A Process including a COPY statement with a SECURE parameter was submitted and failed. The following CSPA011E error message is displayed: Illegal attempt to override Secure+ parameters</p>	<p>You attempted to use the SECURE parameter in a COPY statement for the STS protocol but did not specify OVERRIDE=Y in the remote node record to enable the security override feature.</p>	<p>Take one of the following actions:</p> <ul style="list-style-type: none"> ◆ Remove the SECURE= parameter from the COPY statement and resubmit the Process. ◆ Change the OVERRIDE setting in the remote node record in the parameter file and make sure all other necessary protocol settings are specified. Resubmit the Process including the SECURE= parameter. <p>See Chapter 12, <i>Override Settings in Connect:Direct Processes</i>.</p>
<p>An SSL or TLS session was attempted with a Connect:Direct system that does not implement SSL or TLS.</p>	<p>The trading partner does not have the protocol enabled.</p>	<p>Request that the trading partner configure its node for the correct protocol or disable Secure+ Option for the node.</p>

Problem	Possible Cause	Solution
Either the CSPA203E error message or the CSPA204E message is displayed: SSL or TLS send failure, rc=&RC, rsn=&RSN or SSL or TLS receive failure, rc=&RC, rsn=&RSN.	The client cannot validate the server's certificate.	Ensure that client authentication is turned on and certificate information is defined in the remote node record.
The following CSPA205E error message is displayed: SSL or TLS support requires the TCP/IP protocol.	One of the trading partners is not using TCP/IP for communications.	Determine which trading partner does not have TCP/IP enabled and change the configuration of that trading partner.
The following CSPA200E error message is displayed: Secure+ version mismatch.	You are attempting to use the SSL or TLS protocol to securely communicate with a trading partner that does not have the protocol enabled.	Change the configuration of the remote node record to enable the correct protocol.
The following CSPA206E error message is displayed: Remote certificate is invalid.	The root certificate was not found.	Check the parameter file configuration and ensure that the correct certificate is identified in the remote node record.
The following CSPA207E error message is displayed: Root certificate not found.	The remote certificate could not be validated.	Check the parameter file configuration and ensure the correct key database file is identified in the remote node record.
The following SITA1901 error message is displayed: Sec+ Init failed. Secure= No. Override=No.	The local node record has all Secure+ Option protocols disabled and has override set to no.	Either enable the appropriate protocol in the remote node record or enable override=yes in the local node record.

Problem	Possible Cause	Solution
<p>A Process was submitted and failed. The following CSPA078E error message is displayed: Invalid specification of SECURE= on PROCESS statement. SECURE= cannot be specified in a non-Secure+ environment or when the Remote Node record in the Secure+ Parmfile does not specify OVERRIDE=Y.</p>	<p>You attempted to use the SECURE parameter in a PROCESS statement but did not specify OVERRIDE=Y in the remote node record to enable the security override feature.</p>	<p>Take one of the following actions:</p> <ul style="list-style-type: none"> ◆ Remove the SECURE= parameter from the PROCESS statement and resubmit the Process. ◆ Change the OVERRIDE setting in the remote node record in the parameter file and make sure all other necessary protocol settings are specified. Resubmit the Process including the SECURE= parameter. <p>See Chapter 12, <i>Override Settings in Connect:Direct Processes</i>.</p>
<p>The submit within a Process failed with a reason code of 8. The following SCBI514E or SSUB267E error message is displayed: Equal sign required after SECURE keyword. The SECURE keyword in the PROCESS must be followed by an equal sign.</p>	<p>You attempted to use the SECURE parameter in a PROCESS statement but did not include an equal sign after the SECURE keyword.</p>	<p>Correct the PROCESS statement syntax by inserting an equal sign and resubmit the Process.</p>
<p>The submit within a Process failed with a reason code of 8. The following SCBI515E or SSUB268E error message is displayed: A parsing error occurred on the SECURE keyword when processing the SECURE keyword on the PROCESS statement.</p>	<p>You attempted to use the SECURE parameter in a PROCESS statement but the syntax was faulty.</p>	<p>Correct the PROCESS statement and resubmit the Process. For a complete description of the SECURE parameter and how to use it in the PROCESS statement, go to the Connect:Direct Processes Web site.</p>

Definitions of Certificate Parameters

To avoid some problems associated with CA-signed and self-signed certificates, refer to the following information about certificate parameter definitions required to use Connect:Direct Secure+ Option. Minimum parameter definitions for certificates generated with the RACF, gskkyman, CA-ACF2, and CA-Top Secret security applications are provided.

You may also want to record the parameter definitions you configure for certificates on the worksheets provided for the local and remote node records in Appendix B, *Configuration Worksheets*.

Parameter Definitions for Certificates Generated with the RACF Application

This table describes the minimum parameter definitions required for Connect:Direct Secure+ Option. When two parameters are listed in the same row, the first parameter name is used when you create a certificate and the second parameter name is its equivalent, which is used when you display information about the certificate. Consult the RACF documentation for detailed information about all the certificate parameters and commands.

RACF Parameter	Description	Value Used for Secure+ Option
User ID	Security ID used to start the Connect:Direct Job or Started Task.	RACF-defined ID
Label	Certificate label. LABEL keywords are case and blank sensitive; therefore, the values specified for these keywords must be exact.	Information that identifies the certificate, for example, CD Secure Plus Note: Specify the exact value in the Certificate Label field in the Local Node record of the Secure+ parameter file.

RACF Parameter	Description	Value Used for Secure+ Option
Status	Status of the certificate.	Status=TRUST All certificates used by Connect:Direct Secure+ Option must be Trusted.
NOTBEFORE Start Date	Specifies the local date and time from which the certificate is valid.	Must be a valid date and time
NOTAFTER End Date	Specifies the local date and time after which the certificate is no longer valid. All certificates used in the SSL/TLS handshake, including issuer certificates, must not be expired.	Must be a valid date and time
Key Usage	Facilitates identification and key exchange during SSL/TLS security handshakes.	HANDSHAKE (Required): Indicates that digital signature and key encipherment are enabled. DOCSIGN (Optional): Indicates that non-repudiation is enabled. DATAENCRYPT (Optional): Indicates that data encipherment is enabled. CERTSIGN: Indicates the certificate can sign other digital certificates and CRLs. Note: Do not specify CERTSIGN. Only Certificate Authority (Issuer) certificates should have keyCertSign and cRLSign indicators.
X.509 Subject's Distinguished Name Issuer's Name	Specifies the distinguished name of the issuer that issued or signed a certificate. The name identifies the trusted certificate of the issuer or CA that signed the server certificate. The name identifies the trusted certificate of the issuer or CA that signed the server certificate. The CA or entity certificate with that name must be available within the key database or Keyring. The Issuer Name keywords are case and blank sensitive. Note: Self-signed certificates display the same information in the Issuer Name and Subject Name parameters.	The following fields, which must be enclosed in single quotes, are attributes of the Issuer's Name parameter and the Subject's Name parameter: CN=Common Name of the certificate in single quotes, for example, 'RACF SELF SIGN COMMON' T='Title of person creating certificate' OU='Organizational Unit associated with the person creating the certificate' O='Organization for which the certificate is being created' L='Locality (city) of the entity for which the certificate is created' SP='State/Province of the locality' C='Country of the locality'

RACF Parameter	Description	Value Used for Secure+ Option
X.509 Subject's Distinguished Name Subject's Name	Specifies the certificate's subject distinguished name. It identifies the certificate. This name can identify certificates that may have issued or signed other certificates and can match to other certificates Issuer's Name.	
Private Key Size	Specifies the size of the private key expressed in decimal bits. Key size of 1024 provides a secure encryption. A larger size provides a more secure encryption but requires more CPU to encrypt.	1024
Private Key Type	Specifies how the private key should be stored for future use. Type can be none, non-ICSF, or ICSF. If Type= none, the certificate does not have a private key.	If ICSF is specified, see <i>Connect:Direct Access to System Resources for SSL or TLS</i> on page 22 for requirements.
Ring Name	Specifies the name of the keyring that a certificate is connected with.	If you use a key ring, the exact value in this field must be specified in the Certificate Pathname field for the Local Node record in the Secure+ parameter file.
Usage	Specifies how this certificate should be used in a keyring for the USERID of the person submitting a batch job or signed on to TSO.	PERSONAL
Default	Specifies that the certificate is the default certificate. Only one certificate can be the default certificate. Define the end-user server certificate of the local Connect:Direct node as the default.	YES

Parameter Definitions for Certificates Generated with the GSKKYMANT Utility

This table describes the minimum parameter definitions required for Connect:Direct Secure+ Option. Consult the GSKKYMANT documentation for detailed information about all the certificate parameters and commands.

GSKKMAN Parameter	Description	Value Required for Secure+ Option
Label	Certificate label. LABEL keywords are case and blank sensitive; therefore, the values specified for these keywords must be exact.	Information to identify the certificate, for example, CD Secure Plus Note: Specify the exact Label value in the Certificate Label field in the local node record of the Secure+ parameter file.
Version	X.509 certificates with version number 3 are supported.	3
Trusted	Specifies the certificate status.	Yes
Effective Date	Specifies the local date and time from which the certificate is valid.	Must be a valid date and time
Expiration Date	Specifies the local date and time after which the certificate is no longer valid. All certificates used in the SSL/TLS handshake, including issuer certificates, must not be expired.	Must be a valid date and time
keyUsage	Facilitates identification and key exchange during SSL/TLS security handshakes.	Digital Signature (Required) Non-repudiation Key encipherment Data encipherment
Issuer Name	Specifies the distinguished name of the Issuer that issued or signed a certificate. The name identifies the trusted certificate of the issuer or CA that signed the server certificate. The CA or entity certificate with that name must be available within the key database or keyring. The Issuer Name keywords are case and blank sensitive. Self-signed certificates have the same Issuer name and Subject name.	

GSKKYMAN Parameter	Description	Value Required for Secure+ Option
Certificate Subject Name	Specifies the certificate's subject distinguished name. It identifies the certificate. This name can identify certificates that may have issued or signed other certificates and can match to other certificates Issuer's Name.	The following fields are attributes of the Certificate Subject Name parameter: CN=Common Name of the certificate in single quotes, for example, 'RACF SELF SIGN COMMON' T='Title of person creating certificate' OU='Organizational Unit associated with the person creating the certificate' O='Organization for which the certificate is being created' L='Locality (city) of the entity for which the certificate is created' SP='State/Province of the locality' C='Country of the locality'
Public Key Algorithm	Specifies the algorithm used to encrypt data.	rsaEncryption
Public Key Size	Specifies the size of the public key expressed in decimal bits. Key size of 1024 provides a secure encryption. A larger size provides a more secure encryption but requires more CPU to encrypt.	1024
Key database password	Specifies the password used when you created a key database file.	When you specify a gskkyman key database file name in the Certificate Pathname field for the local node record, you must specify the key database password in the Certificate Pathname Pass Phrase field.

Parameter Definitions for Certificates Generated with the CA-ACF2 Application

This table describes the minimum parameter definitions required for Connect:Direct Secure+ Option. Consult the CA-ACF2 documentation for detailed information about all the certificate parameters and commands.

CA-ACF2 Parameter	Description	Value Used by Secure+ Option
ACID	Security ID used to start the Connect:Direct Job or Started Task.	CA-ACF2 defined ID

CA-ACF2 Parameter	Description	Value Used by Secure+ Option
Label	Certificate label. LABEL keywords are case and blank sensitive; therefore, the values specified for these keywords must be exact.	Information that identifies the certificate, for example, CD Secure Plus Note: Specify the exact value in the Certificate Label field in the Local Node record of the Secure+ parameter file.
Subjsdsn	Specifies the subject's distinguished name. It identifies the certificate. This name can identify certificates that may have issued or signed other certificates and can match to other certificates Issuer's Name.	The following fields, which must be enclosed in single quotes, are attributes of the Issuer's Name parameter and the Subject's Name parameter: CN=Common Name of the certificate in single quotes, for example, 'RACF SELF SIGN COMMON' T='Title of person creating certificate' OU='Organizational Unit associated with the person creating the certificate' O='Organization for which the certificate is being created' L='Locality (city) of the entity for which the certificate is created' SP='State/Province of the locality' C='Country of the locality'
Size	Specifies the size of the private encryption key in bits.	1024
Active	Specifies the local date and time from which the certificate is valid.	Must be a valid date and time
Expire	Specifies the local date and time after which the certificate is no longer valid. All certificates used in the SSL/TLS handshake, including issuer certificates, must not be expired.	Must be a valid date and time
Keyusage	KeyUsage certificate extension, of which one or more of the following values might be coded.	HANDSHAKE (Required): Indicates that digital signature and key encipherment are enabled. DOCSIGN (Optional): Indicates that non-repudiation is enabled. DATAENCRYPT (Optional): Enables the certificate to be used to. CERTSIGN: Indicates the certificate can sign other digital certificates and CRLs. Note: Do not specify CERTSIGN. Only Certificate Authority (Issuer) certificates should have keyCertSign and cRLSign indicators.

CA-ACF2 Parameter	Description	Value Used by Secure+ Option
KEYRING	Specifies the record key of a KEYRING record to which the certificate is associated.	If you use a keyring, the value in this field must be specified in the Certificate Label field for the Local Node record in the Secure+ parameter file.
RINGNAME	Specifies the ring name of a KEYRING record to which the certificate information is associated.	If you use a keyring, the value in this field must be specified in the Certificate Pathname field for the Local Node record in the Secure+ parameter file.
USAGE	Specifies how this certificate should be used in a keyring for the USERID of the person submitting a batch job or signed on to TSO.	PERSONAL
DEFAULT	Specifies that the certificate is the default certificate. Only one certificate can be the default certificate. Define the end-user server certificate of the local Connect:Direct node as the default.	YES

Parameter Definitions for Certificates Generated with the CA-Top Secret Application

This table describes the minimum parameter definitions required for Connect:Direct Secure+ Option. Consult the CA-ACF2 documentation for detailed information about all the certificate parameters and commands.

CA-Top Secret Parameter	Description	Value Used for Secure+ Option
SUBJECTDSN	Specifies the subject's distinguished name. It identifies the certificate. This name can identify certificates that may have issued or signed other certificates and can match to other certificates Issuer's Name.	The following fields, which must be enclosed in single quotes, are attributes of the Issuer's Name parameter and the Subject's Name parameter: CN='Common Name of the certificate in single quotes,' for example, 'RACF SELF SIGN COMMON' T='Title of person creating certificate' OU='Organizational Unit associated with the person creating the certificate' O='Organization for which the certificate is being created' L='Locality (city) of the entity for which the certificate is created' SP='State/Province of the locality' C='Country of the locality' UID='userid'
UID	Security ID used to start the Connect:Direct Job or Started Task.	CA-Top Secret defined ID
NBDATE/NBTIME	Specifies the local date and time from which the certificate is valid.	Must be a valid date and time
NADATE/NATIME	Specifies the local date and time after which the certificate is no longer valid. All certificates used in the SSL/TLS handshake, including issuer certificates, must not be expired.	Must be a valid date and time
KEYSIZE	Specifies the size of the private encryption key in bits.	1024
LABLCERT	Certificate label. LABEL keywords are case and blank sensitive; therefore, the values specified for these keywords must be exact. This parameter is specified when you associate a certificate with an ACID.	Information to identify the certificate, for example, CD Secure Plus Note: Specify the exact value in the Certificate Label field in the Local Node record of the Secure+ parameter file.
ICSF	If Private Key type is ICSF, the private key is stored in the ICSF PKDS (public key data set). Access to the private key then requires that the ICSF application be executing and Connect:Direct have access authority to the ICSF application	If ICSF is specified, see <i>Connect:Direct Access to System Resources for SSL or TLS</i> on page 22 for requirements.
TRUST NOTRUST	Specifies the status of the certificate when you associate a certificate with an ACID.	TRUST

Parameter Definitions for Certificates Generated with the CA-Top Secret Application

CA-Top Secret Parameter	Description	Value Used for Secure+ Option
KEYRING	Specifies the key ring being added to the user's ACID.	If you use a keyring, the value in this field must be specified in the Certificate Label field for the Local Node record in the Secure+ parameter file.
LABLRING	Specifies the label to be associated with the keyring being added to the user, which is used as the identifier of the digital certificate.	If you use a keyring, the value in this field must be specified in the Certificate Pathname field for the Local Node record in the Secure+ parameter file.
DEFAULT	Specifies how this certificate should be used in a keyring for the USERID of the person submitting a batch job or signed on to TSO.	PERSONAL
USAGE	Specifies that the certificate is the default certificate. Only one certificate can be the default certificate. Define the end-user server certificate of the local Connect:Direct node as the default.	YES

Configuration Worksheets

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

- ◆ The *Local Node Security Feature Definition Worksheet* records the security functions defined for the local Connect:Direct node.
- ◆ The *Remote Node Security Feature Definition Worksheet* records the security functions defined for connections with remote nodes with which the local Connect:Direct node communicates. Make a copy of the blank *Remote Node Security Feature Definition Worksheet* for each remote node that you are configuring for Secure+ Option operations.
- ◆ The *.EASERVER Node Security Feature Definition Worksheet* on page 183 records information for the Sterling External Authentication Server remote node record.
- ◆ The *.CLIENT Node Security Feature Definition Worksheet* on page 184 records information for the remote node record used to allow secure TCP API connections.

Local Node Security Feature Definition Worksheet

Record the security feature definitions for the Secure+ Option local node record on this worksheet. Refer to this worksheet as you configure the local node record.

Local Node Name: _____	
TLS protocol enabled:	Yes ____ No ____
SSL protocol enabled:	Yes ____ No ____
STS protocol enabled:	Yes ____ No ____
Configured Security Functions	
Override enabled:	Yes ____
Encryption enabled:	Yes ____ No ____
Autoupd enabled:	Yes ____
Applies only to STS	
Authorization Timeout:	_____ (Numeric value equal to or greater than 0 seconds)
Set the value equal to or greater than the value set for the Connect:Direct TCP.TIMER initialization parameter.	
Certificate Label (label specified when the certificate was generated using one of the security applications; may be called LABLCERT):	_____ Valid only for SSL or TLS
Certificate Pathname:	_____ Valid only for SSL or TLS
key database or key ring	
Password:	_____
Valid only for certificates created in a gskkyman database; leave blank for key rings.	
Cipher Suite(s) to Enable:	_____
Applies only to SSL and TLS	
Enable Digital Signatures:	Yes ____ No ____
Applies only to STS	
Create Public Auth and Sig Key:	Yes ____ No ____
Applies to all protocols	
Algorithm Names:	_____
Applies only to STS	
Algorithms Enabled	__ DES __ TDES __ IDEA
Applies only to STS	
Enable External Authentication:	Yes ____ No ____
Applies only to SSL and TLS	

Remote Node 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+ parameter file that you are configuring for Secure+ Option operations. Refer to this worksheet when you configure a remote node record.

Remote Node Name:	_____
Security Options	
TLS protocol enabled:	Yes ____ No ____
SSL protocol enabled:	Yes ____ No ____
STS protocol enabled	Yes ____ No ____
Enable Override: When override is enabled in a remote node record, <ul style="list-style-type: none"> ◆ Values in the COPY statement override values in the remote node record that uses the STS protocol ◆ Values in the PROCESS statement override values in the remote node record that uses any protocol. 	Yes ____ No ____ Default to local node ____
Encryption enabled:	Yes ____ No ____
Enable External Authentication: Valid only for SSL or TLS	Yes ____ No ____ Default to local node ____
Authorization Timeout: Set the value equal to or greater than the value set for the Connect:Direct TCP.TIMER initialization parameter.	_____(Numeric value equal to or greater than 0 seconds)
TLS or SSL Protocol Functions	
If you enabled the TLS or SSL protocol and you did not define this information in the local node record, set one or more of the following functions:	
Certificate Label: Label specified when the certificate was generated using one of the security applications; may be called LABLCERT.	_____ You can type an asterisk (*) to default to the local node record.
Cipher Suite(s) Enabled:	_____

Ask the trading partner which cipher suites are enabled. Circle all applicable cipher suites.	
<ul style="list-style-type: none"> ◆ SSL_RSA_WITH_AES_128_SHA ◆ SSL_RSA_WITH_AES_256_SHA ◆ SSL_RSA_WITH_3DES_EDE_CBC_SHA ◆ SSL_RSA_WITH_DES_CBC_SHA ◆ SSL_RSA_EXPORT_WITH_RC2_CBC_40_MD5 ◆ SSL_RSA_WITH_RC4_40_SHA ◆ SSL_RSA_WITH_RC4_128_MD5 ◆ SSL_RSA_EXPORT_WITH_RC4_40_MD5 ◆ SSL_RSA_WITH_NULL_SHA ◆ SSL_RSA_WITH_NULL_MD5 	<ul style="list-style-type: none"> ◆ TLS_RSA_WITH_AES_128_SHA ◆ TLS_RSA_WITH_AES_256_SHA ◆ TLS_(or TLS_) RSA_WITH_3DES_EDE_CBC_SHA ◆ TLS_RSA_WITH_DES_CBC_SHA ◆ TLS_RSA_EXPORT_WITH_RC2_CBC_40_MD5 ◆ TLS_RSA_WITH_RC4_40_SHA ◆ TLS_RSA_WITH_RC4_128_MD5 ◆ TLS_RSA_EXPORT_WITH_RC4_40_MD5 ◆ TLS_RSA_WITH_NULL_SHA ◆ TLS_RSA_WITH_NULL_MD5
Certificate Pathname key database or key ring	<p>_____</p> <p>You can type an asterisk (*) to default to the local node record.</p>
To add a second level of security by enabling Client Authentication, set the following two options:	
Enable Client Authentication:	Yes ____ No ____
If client authentication is enabled, specify the certificate common name of the local node certificate in the Client Auth. Compare field.	_____
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 ____
Note: If the trading partner uses an earlier version of Secure+ Option, you need identify the version of Secure+ Option the partner is using.	
Algorithm Names:	_____
Algorithms Enabled:	__ DES __ TDES __ IDEA

.EASERVER Node Security Feature Definition Worksheet

Use the following worksheet to record information to configure the remote node record for .EASERVER node. Refer to this worksheet when you configure the .EASERVER remote node record.

Remote Node Name:	.EASERVER (Required)
TLS protocol enabled:	Yes _____ No _____
SSL protocol enabled	Yes _____ No _____
Note: You must enable either SSL or TLS to communicate with the EA server.	
External Auth Server Def	<p>_____</p> <p>Name of the certificate validation definition configured on the EA server that defines how to validate certificates. This parameter is case sensitive.</p>
External Auth Server Address	<p>_____</p> <p>IP address of server for the Sterling External Authentication Server application</p>
External Auth Server Port	<p>_____</p> <p>Number of the port to use to connect to the EA server</p>
Client Authentication enabled:	Yes _____
Client Authentication Common Name: If client authentication is enabled, specify the certificate common name of the local node certificate in the Client Auth. Compare field.	<p>_____</p>
Certificate Label:	<p>_____</p> <p>You can type an asterisk (*) in the Certificate Label field to default to the local node record.</p>
Certificate Pathname key database or key ring	<p>_____</p> <p>You can type an asterisk (*) in the Certificate Pathname field to default to the local node record.</p>

.CLIENT Node Security Feature Definition Worksheet

Record the security feature definitions for a remote node record named .CLIENT that you create to allow secure connections. Refer to this worksheet when you configure the .CLIENT node record.

Remote Node Name:	.CLIENT (Required)
Note: The node name must be defined as .CLIENT to allow secure connections.	
Security Options	
Autoupd enabled:	Yes _____ No _____
TLS protocol enabled:	Yes _____ No _____
SSL protocol enabled:	Yes _____ No _____
Enable Override:	Yes _____ No _____
ISPF IUI and DMBATCH Options	
ISPF IUI protocol defined as SNA:	Yes _____ No _____
DMBATCH protocol defined as SNA	Yes _____ No _____

Test Secure+ Option with the STS Protocol

This appendix describes the steps for testing Connect:Direct Secure+ Option with the STS protocol. To test an STS configuration and verify that Secure+ Option is working as intended, you must install and set up Connect:Direct and Secure+ Option on two different nodes. You can define two new test nodes as described in the following procedures (Node A and Node B) or you can use two existing nodes. Each node must be defined in the partner's Connect:Direct network map. These existing nodes can be two nodes within your enterprise or you can coordinate testing with a trading partner.

For the initial setup and testing, define *only* the minimum required fields for both nodes as described in the procedures in this section. Specific settings are provided to create nodes to test.

Task Summary

The following list summarizes the order of the tasks necessary to test Secure+ Option with the STS protocol.

1. Define Secure+ Option for Node A
 - a. Create the Secure+ Option local node record and keys
 - b. Create the Secure+ Option remote node record (Node B) and keys
 - c. Export the public keys of Node A for Node B
 - d. Save the parameter file
2. Define Secure+ Option for Node B
 - a. Create the Secure+ Option local node record and keys
 - b. Create the Secure+ Option remote node record (Node A) and keys
 - c. Export the public keys of Node B for Node A
 - d. Import the public keys from Node A to Node B
 - e. Save the parameter file
3. Import public keys from Node B to Node A

4. Save the parameter file for Node A
5. Update Connect:Direct network maps for Node A and Node B
6. Add the **SECURE.DSN=*filename*** parameter to the Connect:Direct for z/OS initialization parameter file of Node A and Node B (where *filename* is the name of the Secure+ parameter file for that node)
7. Restart Connect:Direct (Node A and Node B)
8. Verify that Secure+ Option is enabled (Node A and Node B)
9. Exchange data and compare results
 - a. Send data from Node A to Node B
 - b. Review statistics records for transaction

Access the Secure+ Option Admin Tool

Use the Secure+ Option Administration Tool to set up Secure+ Option to test an STS installation.

To access the Secure+ Option Admin tool, from the **Connect:Direct Administrative Options Menu**, type **SA** and press **Enter**.

Define Secure+ Option for Node A

Defining Secure+ Option for Node A involves:

- ◆ Creating the Secure+ Option local node record and generating the public keys
- ◆ Creating the Secure+ Option remote node record and generating the public keys
- ◆ Exporting the public keys of Node A
- ◆ Saving the parameter file

Create the Secure+ Option Local Node Record and Keys for Node A

To create the local node record:

1. From the **Secure+ Option Admin Tool: Main Screen**, select **Edit** and press **Enter**.
2. Type **1** to select **Create/Update Record** and press **Enter** to display the **Node Identification** panel.

Secure+ Create/Update Panel - Node Identification		
Option:		
EA Parameters	SSL/TLS Parameters	STS Parameters
Node		
NODEA	1 1. Local 2. Remote	
Alias		
Names:	TCP Information: IPaddr: Port:	
<input type="button" value="Import Remote Keys"/> <input type="button" value="Get Record"/> <input type="button" value="OK"/> <input type="button" value="Cancel"/>		

3. To define values for Node A:
 - a. Select each of the panels listed in the following table.
 - b. Type the sample values in the fields listed for each panel.

Fields that are either not valid for the STS protocol or for the type of record being configured are identified and should be left blank.

Panel	Field	Value
Node Identification	Node	Name of Node A (NODEA for this example)
	Local/Remote	Local (1)
	TCP Information	Not valid for local node
	Alias Names	Not valid for local node
STS Parameters	Auth Timeout	90
	Autoupdt	No (2)
	Override	Yes (1)
	Enable STS	Yes (1)
	Signature	Yes (1)
	Encrypt	Yes (1)
	Algorithm	* to provide access to all available algorithms defined in the local node record

Panel	Field	Value
EA Parameters	Enable Auth	No (2)
SSL/TLS Parameters	Enable SSL	No (2)
	Enable TLS	No (2)
	Client Auth	No (2)

4. When you finish setting the sample values, select **STS Parameters** and press **Enter**.
5. Generate the authentication key for the STS protocol:
 - a. Select **Create / Reset Auth. Pubkey** and press **Enter** to display the **Generate Seed** screen.

```
Secure+ Admin Tool: Generate Seed

2 1. Specify Value          Specify the seed value by typing it
                             into the text field.
   2. Sample Value         Generate a seed by processing text
                             entered from the keyboard.

Random Number
Seed:
```

- b. Press **Enter** to accept the default value (**2 - Sample Value**).
- c. On the **Command Prompt** screen, select **OK** and press **Enter**.
- d. When the following screen is displayed, if the message *This is Loop 2 of 10* is displayed, type over data on any line and press **PF3** up to 10 times. This step is only necessary the first time you generate keys within the Secure+ Option Admin Tool for each parameter file. Repeating the process increases the randomness of keys.
- e. Press **PF3** to save the information.

```

File  Edit  Edit_Settings  Menu  Utilities  Compilers  Test  Help
-----
EDIT          SYS06254.T160411.RA000.SSCHR1.R0207298          Columns 00001 00072
Command ==>                                         Scroll ==> PAGE
024: This process cannot proceed if the data in the edit file is unchanged.
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 262144
000002
000003 134217728
000004
000005 32
000006
000007 4
000008
000009 8589934592
000010
000011 131072
000012
000013 8388608
000014
000015 2097152
000016
000017 1073741824

```

- f. When the **Pass Phrase Generation** panel is displayed, type a string at least 32 characters long containing at least one uppercase character, one lowercase character, and one numeric value and press **Enter**.

Note: You do not need to remember this number.

When the **Secure+ Option Create/Update Panel** displays the message *Seed generation complete*, your public key for strong authentication is created, as illustrated in the following sample.

```

Secure+ Create/Update Panel - Node Identification
Option:

EA Parameters          SSL/TLS Parameters      STS Parameters

Node
NODEB                  1 1. Local
                       2. Remote

Alias
Names:                TCP Information:
                       IPaddr:
                       Port:

Import Remote Keys    Get Record            OK            Cancel
    
```

6. Select **Create Sig. Pubkey** and press **Enter**.
7. Press **Enter** to accept the default value (**2 - Sample Value**).

When the message *Seed generation complete* is displayed on the **Secure+ Option Create/Update Panel**, your public key for digital signature is created.

8. Select **OK** and press **Enter**. The node field clears.

Create the Secure+ Option Remote Node Record and Keys for Node B

To create the remote node record:

1. From the **Secure+ Option Admin Tool: Main Screen**, select **Edit** and press **Enter**.
2. Type **1** to select **Create/Update Record** and press **Enter** to display the **Node Identification** panel.

```

Secure+ Create/Update Panel - Node Identification
Option:

EA Parameters          SSL/TLS Parameters    STS Parameters

Node
NODEB                  1 1. Local
                       2. Remote

Alias
Names:                 TCP Information:
                       IPaddr:
                       Port:

Import Remote Keys    Get Record           OK           Cancel
    
```

3. Select each of the panels listed in the following table and type the sample values in the fields listed for each panel. Fields that are either not valid for the STS protocol or for the type of record being configured should be left blank.

Panel	Field	Value
Node Identification	Node	Name of Node B (NODEB for this example)
	Local/Remote	Remote (2)
	TCP Information	Leave blank because Connect:Direct gets IP address information from the network map
	Alias Names	Leave blank
STS Parameters	Auth Timeout	90
	Autoupdt	Default to local (3)
	Override	Default to local (3)
	Enable STS	Default to local (3)
	Signature	Default to local (3)
	Encrypt	Default to local (3)
	Algorithm	* to provide access to all available algorithms defined in the local node record

Panel	Field	Value
EA Parameters	Enable Auth	Default to local (3)
SSL/TLS Parameters	Enable SSL	No (2)
	Enable TLS	No (2)
	Client Auth	No (2)

4. Select **STS Parameters** and press **Enter** when you finish setting these values.
5. Generate the authentication key for the STS protocol:
 - a. Select **Create / Reset Auth. Pubkey** and press **Enter** to display the **Generate Seed** screen.

```
Secure+ Admin Tool: Generate Seed

2 1. Specify Value          Specify the seed value by typing it
                             into the text field.
   2. Sample Value         Generate a seed by processing text
                             entered from the keyboard.

Random Number
Seed:
```

- b. Press **Enter** to accept the default value (**2 - Sample Value**).
- c. On the **Command Prompt** screen, select **OK** and press **Enter**.
- d. When the following panel is displayed, if the message *This is Loop 2 of 10* is displayed, type over data on any line and press **PF3** up to 10 times. This step is only necessary the first time you generate keys within the Secure+ Option Admin Tool for each parameter file. Repeating the process increases the randomness of keys.
- e. Press **PF3** to save the information.


```

File Edit Edit_Settings Menu Utilities Compilers Test Help
-----
EDIT          SYS06254.T160411.RA000.SSCHR1.R0207298          Columns 00001 00072
Command ==>                                         Scroll ==> PAGE
024: This process cannot proceed if the data in the edit file is unchanged.
==MSG> -Warning- The UNDO command is not available until you change
==MSG>          your edit profile using the command RECOVERY ON.
000001 262144
000002
000003 134217728
000004
000005 32
000006
000007 4
000008
000009 8589934592
000010
000011 131072
000012
000013 8388608
000014
000015 2097152
000016
000017 1073741824

```

- f. When the **Pass Phrase Generation** panel is displayed, type a string at least 32 characters long containing at least one uppercase character, one lowercase character, and one numeric value and press **Enter**.

Note: You do not need to remember this number.

When the **Secure+ Option Create/Update Panel** displays the message *Seed generation complete*, your public key for strong authentication is created.

6. Select **Create Sig. Pubkey** and press **Enter**.
7. On the **Generate Seed** screen, press **Enter** to accept the default value (**2 - Sample Value**).
8. Change some of the text by typing over it and press **PF3**.

On the **Secure+ Option Create/Update Panel**, the message *Seed generation complete* is displayed when your public key for digital signature is created.

9. Select **OK** and press **Enter**.
10. Select **Cancel** and press **Enter** to return to the **Secure+ Option Admin Tool: Main Screen** and display the two node records you defined.

```

File  Edit  Key Management  Help
-----
                                         Row 1 of 2
Secure+ Admin Tool: Main Screen
Option ===>                                         Scroll CSR

Table Line Commands are:
E Export pub. key      H View History      D Delete node
U Update node         I Insert node

Secure
LC Node Name      Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
NODEA             L    YNNN   Y          Y          Y          N          N
NODEB             R    *NNN   *          *          *          *          *
***** BOTTOM OF DATA *****

```

Export Node A's Public Keys

To export the local node authentication and digital signature public key values to the remote node you are testing with:

1. From the **Secure+ Option Admin Tool: Main Screen**, select **Key Management** and press **Enter**.
2. Type **2** to select **Export Public Keys** and press **Enter**.

Note: If warning messages are displayed, read them. Press **PF3** to continue.

```

Secure+ Admin Tool: File Selection
Enter file name for: INPUT SECURE PARM FILE

File
Name: $CD.SECURE.EXPORT                                         Browse

File System Type:
1 1. MVS  2. HFS                                               Cancel

```

3. Press **Enter** to accept the default file name and MVS file system type.
4. Select **OK** and press **Enter**.

The **Secure+ Option Admin Tool: Main Screen** displays the message *Export Successful*.

Save the Secure+ Parameter File for Node A

When you save the Secure+ parameter file you created for Node A, the access file is also created.

To save the parameter file:


1. From the **Secure+ Option Admin Tool: Main Screen**, select **File** and press **Enter**.
2. Type **7** to select **Save As** and press **Enter**.

Note: If warning messages are displayed, read them. Press **PF3** to continue.


The **File Selection** screen is displayed.

3. Press **Enter** to accept the default file name or type a file name of your choice.
4. Type your site-specific job card information, allocation information, STEPLIB DSNs, and Access file Dsname as in the following example, using the library names created in your installation.


Site-dependent
job card information




Site-dependent
Allocation information



Site-dependent
STEPLIB DSNs



Access file Dsname



```

Secure+ Admin Tool: "Save As" information
(Or press PF3/PF12 to cancel)

What do you want to do with the generated JCL?
_ 1. Browse 2. Edit 3. Submit      Make Pass Phrase
Job statement information. Verify before proceeding.

=====> //SCDA JOB (ACCOUNT),NAME,MSGCLASS=,NOTIFY=$CD
=====> //* SECOND JCL
=====> //* THIRD JCL
=====> //* FOURTH JCL

Mgmt. Class _____ Volume Serial _____
Sfg. Class _____
Data Class _____

Product Load library information. Verify before proceeding.

=====> //* FIRST STEPLIB
=====> //* SECOND STEPLIB
=====> //* THIRD STEPLIB
Access file Dsname (long names may need quotation)
=====> _____

```

5. Type **3** to select **Submit** and press **Enter** to save your parameter file.
6. After the job is submitted, a screen similar to the following is displayed:

```
JOB $CDH(JOB01111) SUBMITTED
***
```

7. Press **Enter** to return to the **Secure+ Option Admin Tool: Main Screen**.
8. Verify that you get a return code of 0 (zero). Research any return code other than zero to determine the cause of the error condition.
9. Press **PF3** to return to the **Connect:Direct Administrative Options Menu**.

Define Secure+ Option for Node B

Defining Secure+ Option for Node B involves:

- ◆ Creating the Secure+ Option local node record and generating the public keys
- ◆ Creating the Secure+ Option remote node record and generating the public keys
- ◆ Exporting the public keys of Node B
- ◆ Importing the public keys from Node A
- ◆ Saving the parameter file

Create the Secure+ Option Local Node Record and Keys for Node B

At the second location you are using to test your setup of Secure+ Option, you must configure a local node record for Node B.

To configure Node B as the local node.

1. Start the Secure+ Option Admin Tool.
2. From the **Secure+ Option Admin Tool: Main Screen**, select **Edit** and press **Enter**.
3. Type **1** to select **Create/Update Record** and press **Enter** to display the **Secure+ Option Create/Update Panel - Node Identification** panel.
4. Select each of the panels listed in the following table and type the sample values in the fields listed for each panel. Fields that are either not valid for the STS protocol or for the type of record being configured are identified and should be left blank.

Panel	Field	Value
Node Identification	Node	Name of Node B (NODEB for this example)
	Local/Remote	Local (1)
	TCP Information	Leave blank because Connect:Direct gets IP address information from the network map
	Alias Names	Leave blank
STS Parameters	Auth Timeout	90
	Autoupdt	No (2)
	Override	Yes (1)
	Enable STS	Yes (1)
	Signature	Yes (1)
	Encrypt	Yes (1)
	Algorithm	* to provide access to all available algorithms defined in the local node record

Panel	Field	Value
EA Parameters	Enable Auth	No (2)
SSL/TLS Parameters	Enable SSL	No (2)
	Enable TLS	No (2)
	Client Auth	No (2)

5. Select **Create Auth. Pubkey** and press **Enter**.
6. On the **Generate Seed** screen, press **Enter** to accept the default value (**2 - Sample Value**).
7. On the **Command Prompt** screen, select **OK** and press **Enter**.
8. When the message *This is Loop 2 of 10* is displayed, type over the data on any line and press **F3** up to 10 times. This is only necessary the first time you generate keys within the Secure+ Option Admin Tool for each parameter file, and it increases the randomness of your keys.
9. Press **F3** to save the changes.
10. When the **PassPhrase Generation** panel is displayed, type a 32-byte character string with uppercase, lowercase, numeric, and alphabetic characters. Press **Enter**.

Note: You do not need to remember this number.

When the message *Seed generation complete* is displayed on the **Secure+ Option Create/Update Panel**, your public key for authentication is created.

11. Select **Create Sig. Pubkey** and press **Enter**.
12. On the **Generate Seed** screen, press **Enter** to accept the default value (**2 - Sample Value**).
When the message *Seed generation complete* is displayed On the **Secure+ Option Create/Update Panel**, your public key for digital signature is created.
13. Select **OK** and press **Enter**. The node field clears.

Create the Secure+ Option Remote Node Record and Keys for Node A

To configure a remote node record for Node A on Node B:

1. From the **Secure+ Option Admin Tool: Main Screen**, select **Edit** and press **Enter**.
2. Type **1** to select **Create/Update Record** and press **Enter** to display the **Node Identification** panel.
3. Select each of the panels listed in the following table and type the sample values in the fields listed for each panel. Fields that are either not valid for the STS protocol or for the type of record being configured should be left blank.

Panel	Field	Value
Node Identification	Node	Name of Node A (NODEA for this example)
	Local/Remote	Remote (2)
	TCP Information	Leave blank because Connect:Direct gets IP address information from the network map
	Alias Names	Leave blank
STS Parameters	Auth Timeout	90
	Autoupdt	Default to local (3)
	Override	Default to local (3)
	Enable STS	Default to local (3)
	Signature	Default to local (3)
	Encrypt	Default to local (3)
	Algorithm	* to provide access to all available algorithms defined in the local node record
EA Parameters	Enable Auth	Default to local (3)
SSL/TLS Parameters	Enable SSL	No (2)
	Enable TLS	No (2)
	Client Auth	No (2)

4. Select **STS Parameters** and press **Enter** when you finish setting these values.
5. Select **Create Auth. Pubkey** and press **Enter**.
6. On the **Generate Seed** screen, press **Enter** to accept the default value (**2 - Sample Value**).
7. On the **Command Prompt** screen, select **OK** and press **Enter**.
8. When the message *This is Loop 2 of 10* is displayed, type over the data on any line and press **F3** up to 10 times. This is only necessary the first time you generate keys within the Secure+ Option Admin Tool for each parameter file, and it increases the randomness of your keys.
9. Press **F3** to save the changes.
10. When the **PassPhrase Generation** panel is displayed, type a 32-byte character string with uppercase, lowercase, numeric, and alphabetic characters. Press **Enter**.

Note: You do not need to remember this number.

When the message *Seed generation complete* is displayed on the **Secure+ Option Create/Update Panel**, your public key for authentication is created.

11. Select **Create Sig. Pubkey** and press **Enter**.
12. On the **Generate Seed** screen, press **Enter** to accept the default value (**2 - Sample Value**).

- Change some of the text by typing over it. Press **PF3**.

On the **Secure+ Option Create/Update Panel**, the message *Seed generation complete* is displayed. Your public key for digital signatures is created.

- Select **OK** and press **Enter**.

- Select **Cancel** and press **Enter** to return to the **Secure+ Option Admin Tool: Main Screen**. Your screen should have two nodes populated, as shown in the following example.

```

File  Edit  Key Management  Help
-----
                                         Row 1 of 2
                Secure+ Admin Tool: Main Screen
Option ==>                                         Scroll CSR

                Table Line Commands are:

E Export pub. key          H View History          D Delete node
U Update node              I Insert node

                Secure
LC Node Name              Type  123C  Override Encryption Signature ExtAuth Autoupd
-----
   NODEA                   R    *NNN   *          *          *          *
   NODEB                   L    YNNN   Y          Y          Y          N          N
***** BOTTOM OF DATA *****

```

Export the Public Keys of Node B

To export the Node B public keys for Node A:

- From the **Secure+ Option Admin Tool: Main Screen**, select **Key Management** and press **Enter**.
- Type **2** to select **Export Public Keys** and press **Enter**.

Note: If warning messages are displayed, read them. Press **PF3** to continue.

- Press **Enter** to accept the default file name and MVS file system type.

On the **Secure+ Option Create/Update Panel**, the message *Export Successful* is displayed.

Import the Public Keys from Node A

To import your authentication and digital signature public key values from the remote node you are testing with (Node A):

- From the **Secure+ Option Admin Tool: Main Screen**, select **Key Management** and press **Enter**.
- Type **1** to select **Import Public Keys** and press **Enter**.
- Type the file name prefix or partial prefix followed by an asterisk (*), select **Browse** and press **Enter**.

4. Type **S** next to the export file name with an extension for this node name (for example, `$CD.SECURE.EXPORT.#NODEB`) and press **Enter**.

The message *2 entries imported from NODEA* is displayed on the **Secure+ Option Admin Tool: Main Screen**, indicating that both the authentication and the digital signature public keys have been imported.

5. Type **U** next to NODEA and press **Enter** to ensure that you now have keys for the remote (RMT).

Save the Secure+ Parameter File for Node B

When you save the Secure+ parameter file you created for Node B, the access file is also created.

To save the parameter file for Node B:

1. From the **Secure+ Option Admin Tool: Main Screen**, select **File** and press **Enter**.
2. Type **7** to select **Save As** and press **Enter**.
3. Press **Enter** to accept the default file name.
4. Type your site-specific job card information, allocation information, STEPLIB DSNs, and Access file Dsname as in the following example, using the library names created in your installation.

```

Secure+ Admin Tool: "Save As" information
(Or press PF3/PF12 to cancel)

What do you want to do with the generated JCL?
_ 1. Browse 2. Edit 3. Submit      Make Pass Phrase
Job statement information. Verify before proceeding.

=====> //SCDA JOB (ACCOUNT),NAME,MSGCLASS=,NOTIFY=$CD
=====> /* SECOND JCL
=====> /* THIRD JCL
=====> /* FOURTH JCL

Mgmt. Class _____ Volume Serial _____
Stg. Class _____
Data Class _____

Product Load library information. Verify before proceeding.

=====> /* FIRST STEPLIB
=====> /* SECOND STEPLIB
=====> /* THIRD STEPLIB
Access file Dsname (long names may need quotation)
=====> _____
    
```

Annotations on the left side of the screenshot:

- Site-dependent job card information (points to the job card section)
- Site-dependent Allocation information (points to the Mgmt. Class, Stg. Class, Data Class section)
- Site-dependent STEPLIB DSNs (points to the STEPLIB section)
- Access file Dsname (points to the Access file Dsname section)

5. Type **3** to select **Submit** and press **Enter** to save your parameter file.
6. After the job is submitted, a screen similar to the following is displayed:

```

JOB $CDH(JOB01111) SUBMITTED
***
    
```

7. Press **Enter** to return to the **Secure+ Option Admin Tool: Main Screen**.

8. Verify that you get a return code of 0 (zero). Research any return code other than zero to determine the cause of the error condition.
9. Press **PF3** to return to the **Connect:Direct Administrative Options Menu**.

Import the Public Keys of Node B to Node A

To import the authentication and digital signature public keys from Node B:

1. Start the Secure+ Option Admin Tool.
2. From the **Secure+ Option Admin Tool: Main Screen**, select **File** and press **Enter**.
3. Type **2** to select **Open** and press **Enter**.
4. Type the complete file name of the parameter file that you created for Node A and press **Enter**.
5. Select **Key Management** and press **Enter**.
6. Type **1** to select **Import Public Keys** and press **Enter**.
7. Type the file name prefix or partial prefix followed by an asterisk (*), select **Browse**, and press **Enter**.
8. Type **S** next to the export file name with an extension for this node name (for example, \$CD.SECURE.EXPORT.#NODEA) and press **Enter**.

The message *2 entries imported from NODEB* is displayed on the **Secure+ Option Admin Tool: Main Screen**, indicating that both the authentication and the digital signature public keys have been imported.

9. Type **U** next to NODEB and press **Enter** to ensure that you now have keys for the remote node record (RMT).

Save the Secure+ Parameter File for Node A

To save the Node A parameter file:

1. From the **Secure+ Option Admin Tool: Main Screen**, select **File** and press **Enter**.
2. Type **7** to select **Save As** and press **Enter**.
3. Press **Enter** to accept the default file name.
4. Type your site-specific job card information, allocation information, STEPLIB DSNs, and Access file Dsname as in the following example, using the library names created in your installation.

Site-dependent
job card information

Site-dependent
Allocation information

Site-dependent
STEPLIB DSNs

Access file Dsname

```

Secure+ Admin Tool: "Save As" information
(Or press PF3/PF12 to cancel)

What do you want to do with the generated JCL?
_ 1. Browse 2. Edit 3. Submit Make Pass Phrase
Job statement information. Verify before proceeding.

====> //SCDA JOB (ACCOUNT),NAME,MSGCLASS=,NOTIFY=$CD
====> /* SECOND JCL
====> /* THIRD JCL
====> /* FOURTH JCL

Mgmt. Class _____ Volume Serial _____
Stg. Class _____
Data Class _____

Product Load library information. Verify before proceeding.

====> /* FIRST STEPLIB
====> /* SECOND STEPLIB
====> /* THIRD STEPLIB
Access file Dsname (long names may need quotation)
====> _____
                
```

5. Type **3** to select **Submit** and press **Enter** to save your parameter file.
6. After the job submits, a screen similar to the following is displayed:

```

JOB $CDH(JOB01111) SUBMITTED
***
                
```

7. Press **Enter** to return to the **Secure+ Option Admin Tool: Main Screen**.
8. Verify that you get a return code of 0 (zero). Research any return code other than zero to determine the cause of the error condition.
9. Press **PF3** to return to the **Connect:Direct Administrative Options Menu**.

Update Connect:Direct Network Maps for Node A and Node B

If you used existing nodes for testing, this step is not necessary. If you created new test nodes (Node A and Node B), update Connect:Direct network maps (netmaps) that you created during the initial installation of Connect:Direct.

To add these test nodes to the Connect:Direct network map:

1. Update the network map of Node A to add Node B.
2. Update the network map of Node B to add Node A.

Note: Refer to the *Connect:Direct for z/OS Administration Guide* for specific instructions for updating the Connect:Direct network map.

Modify Connect:Direct Initialization Parameters

For both nodes, add the parameter **SECURE.DSN=*filename*** to the Connect:Direct for z/OS initialization parameters, where *filename* is the name of the Secure+ parameter file for that node.

Restart Connect:Direct

Restart Connect:Direct for both nodes.

Verify Secure+ Option is Enabled

When you have successfully finished the preceding procedures, verify that Secure+ Option is enabled. The following message is displayed:

```
SITTA028I Secure+ Initialization Complete
```

Exchange Data and Compare Results

To verify installation of Secure+ Option and test the configuration created in the preceding procedures:

1. Create and run the following sample Connect:Direct Process to send data from Node A to Node B.

```
SAMPLE  PROCESS  SNODE=NODEB
*
COPYFILE COPY FROM ( PNODE          -
                    DSN='TEST.INPUT.DATASET' -
                    DISP=SHR          -
                    )                  -
          TO      ( SNODE          -
                    DSN='TEST.OUTPUT.DATASET' -
                    DISP=(NEW,CATLG) -
                    )                  -
                    SECURE=(ENC=Y,SIG=Y)
```

2. Review the statistics record for the transaction to verify the success of the sample Process by selecting the extended record type for session begin (SB), as shown in the following sample record.

```

Function      =>  Session Begin           Start Time => 18:49:58
                                         Start Date => 04/28/2003

Process Name =>  SAMPLE
Process Num  =>  7                        Comp Code  => 00000000
                                         Comp Msg   => SVTM055I

Userid       =>  $CD
Primary Node =>  NODEA
Secondary Node NODEB
Submitter Node SC.MVS.$CD3
                Pnode Signature Enabled = Yes
                Snode Signature Enabled = Yes
                Merged Signature Enabled = Yes
                Pnode Encrypt.Data Algorithms...
                DESCBC56
                TDESCBC112
                IDEACBC128
                Snode Encrypt.Data Algorithms...
                TDESCBC112
                DESCBC56
                IDEACBC128
                System Data Encryption   = DESCBC56
    
```

Configure for a Secure Connection between z/OS and OpenVMS Nodes

This appendix provides a detailed example for defining a remote node record in both a Connect:Direct for z/OS Secure+ parameter file and a Connect:Direct for OpenMVS Secure+ parameter file to set up a secure connection between the two nodes.

In this example, two nodes have set up records in their respective Secure+ parameter files:

- ◆ The Connect:Direct for z/OS node is named Q1A.ZOA.V4700 and is defined in a remote node record in the Connect:Direct for OpenVMS Secure+ parameter file and in the Connect:Direct for OpenVMS network map.
- ◆ The Connect:Direct for OpenVMS node is named Q1A.ITAN.V3400 and is defined in a remote node record in the Connect:Direct for z/OS Secure+ parameter file and in the Connect:Direct for z/OS network map.

The Connect:Direct Secure+ Option records are defined to allow each node to act as either the client (PNODE) or the server (SNODE), depending on which one initiates the session.

Defining Records in the z/OS Secure+ Parameter File

In the Connect:Direct for z/OS Secure+ parameter file, the local node record has the following settings:

- ◆ Y in the Override field
- ◆ N in the Enable TLS (or SSL) field
- ◆ N in the Client Auth field

The settings for the local node record have the following effects: Disabling Connect:Direct Secure+ Option in the local node record means that the protocol and other settings for secure connections must be defined in each remote node record; enabling the Override parameter allows settings in remote node records to override those in the local node record; client authentication is not enabled for all remote nodes.

The remote node record defined for the OpenVMS node named Q1A.ITAN.V3400 in the z/OS Secure+ parameter file has the following settings:

- ◆ Node Identification is Q1A.ITAN.V3400. This value must correspond to the node name specified in the Connect:Direct for z/OS network map.
- ◆ Override is not applicable in the remote record and defaults to N.
- ◆ The TLS protocol is enabled for sessions to connect to this node.
- ◆ This OpenVMS node will not request client authentication of z/OS nodes with which it communicates.
- ◆ Auth Timeout is set to the two-minute default to identify the maximum time that the system waits to receive Connect:Direct control blocks exchanged during the authentication protocol.

The following Secure+ Create/Update Panel - SSL/TLS Parameters panel for Connect:Direct Secure+ Option for z/OS illustrates the settings for the OpenVMS node named Q1A.ITAN.V3400 and commentary on the values set for the parameters.

```

Secure+ Create/Update Panel - SSL/TLS Parameters
Option:
Node Identification      EA Parameters      STS Parameters
Node                    2 1. Y 2. N 3. D Enable SSL
Q1A.ITAN.V3400=remote node 2 1. Y 2. N 3. D Enable TLS
name
Common Parameters between SSL and TLS
2 1. Y 2. N 3. D Override ==> Override parm (N/A)
2 1. Y 2. N 3. D Client Auth
1 1. Y 2. N 3. D Encrypt
Auth Timeout: 120
Certificate Label      | mfcert_a ==> certificate for z/OS node |
Cipher Suites         | 352F04050A09030601 ==> z/OS cipher suites |
Certificate Pathname  | * ==> default to path in local node rec |
Client Auth. Compare  |
-----
OK                      Cancel

```

The information in the bottom half of the screen pertains to the key certificate for the z/OS node. The OpenVMS remote node record for the z/OS node has enabled client authentication, as shown in *Defining the z/OS Remote Node Record in the OpenVMS Secure+ Parameter File* on page 207. Therefore, when the z/OS node initiates the session, the OpenVMS node (the server) requests that the client send its ID certificate so that the OpenVMS node can authenticate the client by validating the key certificate defined on this panel (mfcert_a) against the key certificate specified in the Root Certificate file field (mfcert_a.txt) of the z/OS remote node record in the OpenVMS Secure+ parameter file, as illustrated on page 207. When the z/OS node is the server, it must send its public key, which is stored in the mfcert_a file, to the OpenVMS node during server authentication.

In this example, the z/OS key certificate resides in the default key database defined for the local node (indicated by *). If the certificate location does not default to the local node, the remote node definition must point to the absolute path. Definitions for the default key database are stored in the

local node record. Certificate information identifying the z/OS node to remote nodes and remote nodes to the z/OS node is stored in the GSKKYPAN database. When certificates are exchanged, trading partners send the ID certificate portion of their keys to each other. In the z/OS system, this information must be imported into the GSKKYPAN database.

Note: In the OpenVMS system, fully qualified paths are always required for file locations.

The TLS ciphers previously selected are shown using the standard two-byte IBM convention for displaying ciphers (352F04050A09030601). The systems negotiate a cipher suite common to both the z/OS and OpenVMS nodes to encrypt information during the handshake and when actual data is being transmitted.

Defining the z/OS Remote Node Record in the OpenVMS Secure+ Parameter File

The following example shows the remote node record that defines the Connect:Direct for z/OS node named Q1A.ZOA.V4700. The OpenVMS network map contains an adjacent (remote) node record with the exact same name.

```

Node Name:      Q1A.ZOS.V4700
Node type:      R
1. Protocol:    T
2. Client Authentication: y
3. Authentication timeout: 100
4. Certificate common name: mfsscert_a
5. Root Certificate file: disk$data:[qaitan.q1a]mfcert_a.txt
6. Key Certificate file: disk$data:[qaitan.q1a]2048sskeycert.txt
7. Passphrase:  ****
8. Cipher suites: EXP_RC4_MD5, RC4_MD5, RC4_SHA, EXP_RC2_CBC_MD5, IDEA_CBC_SHA,
                  EXP_DES_CBC_SHA, DES_CBC_SHA, DES_CBC3A
    
```

When the OpenVMS node is the server, it requests that the client authenticate itself (Client Authentication = Y) and send its certificate common name (mfsscert_a) for an extra layer of authentication. The public key information for the z/OS node is stored in the Root Certificate file named mfcert_a.txt; its location is specified (disk\$data:[qaitan.q1a]).

The key certificate file contains the information that identifies the OpenVMS node to other nodes (disk\$data:[qaitan.q1a]2048sskeycert.txt). In order for the OpenVMS system to access its private key to send information to the other node, the passphrase must be entered as well. The z/OS node validates this key certificate information against the information stored in its GSKKYPAN database.

The cipher suites are listed in the order of preference, and the first one that matches a cipher suite defined for the other node is used to establish a session.

A

Access File

A file that is generated automatically when you create the Secure+ parameter file for the first time and contains the Secure+ Option passphrase to encrypt and decrypt the private keys in the Secure+ parameter file. Your Secure+ Option administrator must secure the access file. This file can be secured with any available file access restriction tools. Availability of the access file to unauthorized personnel can compromise the security of data exchange.

Administration Tool (Admin Tool)

The Secure+ Option tool that enables configuring and maintaining the Secure+ Option environment. This is the only tool you can use to configure and maintain Secure+ Option.

Asymmetric Keys

A separate but integrated user key pair comprised of one public key and one private key. Each key either encrypts information or decrypts information but does not perform both functions.

Authentication

The process of verifying that a particular name really belongs to a particular entity and assurance that a message is not modified in transit or storage.

C

Certificate

A self-issued document or one obtained from a certificate authority 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 certificate authority; (3) period of validity (certificates expire and must be renewed); and (4) administrative and extended information.

The certificate authority analyzes the CSR fields, validates the accuracy of the fields, generates a certificate, and sends it to the requester.

Certificate Authority

An organization that issues digitally-signed certificates. The certificate authority authenticates the certificate owner's identity and the services that the owner is authorized to use, issues new certificates, renews existing certificates, and revokes certificates belonging to users who are no longer authorized to use them. The CA digital signature is assurance that anybody that trusts the CA can also trust that the certificate that it signs is an accurate representation of the certificate owner.

Certificate Common Name

A part of the certificate, which can be used in client authentication to enable the remote node (secondary node or SNODE) to verify the identify of the primary node (PNODE) when a secure connection is being established.

Certificate Revocation List

A list of certificates that have been revoked.

Certificate Signing Request

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 authentication 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.

D**Data Confidentiality**

Ensuring that data remains private during transmission.

Data Integrity

Ensuring that information is not altered during transmission.

Decryption

The process of converting encrypted data back into meaningful information.

Digital Certificate

A type of asymmetric cryptography used to simulate the security properties of a signature in digital, rather than written, form for verifying participant identity in the exchange of electronic information. The process involves using a private key for creating signatures and a public key for verifying signatures.

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 encrypts or decrypts data.

F**FTP**

Internet application and network protocol for transferring files between host computers. File transfer protocol.

I

Integrity

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

J

Java

A programming language that allows development of applications that can run from any kind of device or machine—a PC, a Macintosh computer, a network computer, the Internet, or a mobile phone. The Java language makes it possible to develop software that is portable, modular, and secure.

JDK

The Java Development Kit (JDK) contains the software and tools that developers need to compile, debug, and run applets and applications written using the Java programming language.

JRE

The Java Runtime Environment (also known as the Java Runtime or JRE) consists of the Java virtual machine, the Java platform core classes, and supporting files. It is the runtime part of the Java Development Kit and provides no compiler, debugger, or tools. The JRE is the smallest set of executable files that constitute the standard Java platform.

K

Key

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

Key Certificate File

A file that contains the encrypted private key and the ID (public key) certificate, which is used to authenticate an entity.

L

License Management Key (AP Key)

A file containing definitions that Connect:Direct and Secure+ Option use to enable the software. You must have a license management key to use either of these applications.

Local Node Record

The base record in a Secure+ parameter file that defines the Connect:Direct server. It includes the most commonly used settings at a site and is the central node through which all communication is filtered. Depending upon how each remote node record is configured, trading partner node records may use settings that are defined in the local node record.

N

Network Map (Netmap)

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

Nonrepudiation

Providing undeniable proof of origin of transmitted data.

P

Passphrase

Similar to a password but can be any characters, including spaces. A passphrase is stronger than a password, although not many programs support the use of a passphrase.

Password

A character-limited word or phrase that establishes identity to allow access to a system. Generally, a password is composed of letters, numbers, or both.

Primary Node (PNODE)

The Connect:Direct node that initiates a session and is the controlling node. Every Process has one primary node and one secondary node.

Private Key

String of characters used as the private, “secret” part of a complementary public-private key pair. The asymmetric cipher of the private key is used to sign outgoing messages and decrypt data that is encrypted with its complementary public key. Data that is encrypted with a Public Key can only be decrypted using its complementary Private Key. The private key is never transmitted.

Proof of Data Origin

A method of verifying the identity of the sender and that information is not altered during an electronic exchange.

Public Key

String of characters used as the publicly distributed part of a complementary public-private key pair. The asymmetric cipher of the public key is used to confirm *signatures* on incoming messages and encrypt data for the session key that is exchanged between server and client during negotiation for an SSL/TLS session. The public key is part of the ID (public key) certificate. This information is stored in the key certificate file and read when authentication is performed.

R

Remote Node Record

An entry in the Secure+ parameter file that defines the security settings used to communicate with a trading partner. A remote node record must be defined for every trading partner you communicate with.

Root Certificate File

File which contains one or more trusted root certificates used to authenticate ID (public) certificates sent by trading partners during the Secure+ protocol handshake.

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.

SMP/E

IBM System Modification Program Extended, used to simplify the installation of software.

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.

Sterling External Authentication Server

Application used to validate certificates that are passed to the external authentication server during an SSL or TLS session. You can configure the Sterling External Authentication Server application for certificate chain validation, including the option to validate certificates against one or more Certificate Revocation Lists (CRLs) that are stored on an LDAP server. The Sterling External Authentication Server application can also return attributes associated with the incoming certificate, such as group information, that are stored on an LDAP server.

T

Third-Party Certificate

A certificate, other than those that are preconfigured for the application, that identifies an organization. If third-party certificates are used by the server, the corresponding trusted certificate must be installed in the client manually.

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.

Trusted Root Certificate File

See *Root Certificate File*.

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.

A

Access File, defined 14
Accessing help 44
Accessing the Secure+ Admin Tool 186
Admin Tool
 function keys defined 43
 Main screen, description 38
 starting and using 37
algorithms, changing 148
Authentication, defined 9
Autoupdate, public keys 35

C

certificates
 application-specific requirements 28
 CA-signed
 advantages and disadvantages 24
 CA-ACF2 parameter definitions 173
 CA-Top Secret parameter definitions 175
 parameter definitions for GSSKYMANT 171
 RACF parameter definitions 169
 general requirements 28
 methods to obtain 29
 obtain for SSL and TLS 28
 security applications for generating 27
 self-signed
 advantages and disadvantages 24
 terminology 25
 types 24
cipher suites, changing 147
client authentication
 defined 20
 processing 22
Configured Security Functions 180
configuring a remote node record for SSL
 importing the network map 110
configuring a remote node record for STS

 importing the network map 114
 configuring the local node imported from the network
 map, configuration guidelines 95
 configuring the local node record for SSL, importing the
 network map 96
 configuring the local node record for STS, importing the
 network map 103
Connect:Direct
 prepare for Secure+ Option 133
 set up to use certificates 29
Connect:Direct for OpenVMS
 configuring node record 205
Connect:Direct for z/OS
 configuring node record 205
Connect:Direct Process, use to send export key file 125
Copy Step Start statistics record, fields defined 159
Copy Termination statistics record, fields defined 160

D

Data confidentiality, defined 9, 31
Data encryption
 definition 31
 merged settings 35
 supported encryption algorithms 32
Data integrity, defined 9
data security, using STS 31
Digital signature, merged settings 34
disable Secure+ Option 147

E

exchanging STS keys, initial exchange 123
exporting, STS keys 124
External authentication server, configuring remote
 record 83

External authentication, defined 9, 10

G

guidelines

- configuring parameter file manually 51
- configuring parameter file manually for STS 67
- configuring remote node records imported from the network 110
- configuring the local node record imported from the network map 95

I

implementation of SSL and TLS, planning 19

importing STS keys 126

manually 129

importing the network map

- configuration guidelines for remote node records 110
- configuring a remote node record for SSL 110
- configuring a remote node record for STS 114
- configuring remote node records 109
- configuring the local node record for SSL 96
- configuring the local node record for STS 103
- disabling Secure+ 119

K

Key exchange, method 36

Key update frequency 36

Keyfile management, defined 36

Keys

- planning implementation 35, 36
- updating 150

L

local and remote nodes, configuration scenarios 45

local node record

- adding manually for SSL protocol 52
- adding manually for STS 68
- updating keys 150

Local Node Security Feature Definition Worksheet 180

M

Managing keyfiles 36

Merged Secure+ Option settings

- defined 34
- using the STS Protocol 34

N

Network map, populating Secure+ parameter file 47

Non-repudiation, defined 9

O

OpenVMS

- configuring node record for Connect:Direct 205

overriding remote node values in a PROCESS statement 38, 42, 46, 63, 113

overriding security function values from the PROCESS statement, remote node records 135

P

parameter file

- methods to populate 44

parameters file

- opening 143
- populating from network map 47
- resecuring 145
- save and submit 131
- scenarios for creating 45
- viewing information about 143

preparing for Secure+ Option configuration 37

preventing nonsecure API connections, configuring remote node record for SSL 86

PROCESS statement

- overriding security function values for all protocols 135
- overriding values in a remote node record 38
- Secure+ Option examples 137

Proof of data origin, defined 31

protocol-specific panels, parameters displayed 39

Public keys

- resetting in remote node records 151

Q

Quickstart, populating Secure+ parameter file 47

R

- remote node record
 - adding for external authentication server 83
 - adding manually for SSL protocol 60
 - adding manually for STS 74
 - deleting 149
 - updating keys 150
- Remote Node Security Feature Definition Worksheet 181
- Resetting, keys in remote node records 151

S

- saving remote node records
 - save action option 145
- Secure+ Admin Tool
 - accessing Help 44
 - types of Help 44
 - using 37
- Secure+ Option access file, description 14
- Secure+ Option, maintaining 141
- Secure+ parameter file
 - populating manually 49
 - populating using Quickstart 47
- Secure+ parameters, type of record valid for 40
- Session Begin statistics record, fields defined 158
- SSL and TLS protocol, system resource requirements for Connect:Direct 22
- SSL protocol
 - create local node record manually 52
 - data security 20
 - defined 19
 - overview 19
- Station-to-station protocol (STS)
 - defined 11, 31
 - functions, overriding from the COPY statement 35
 - implementation, planning 31
 - keys
 - exporting 124
 - importing 126
 - initial exchange 123
 - managing overview 35
 - merged settings 34
 - Secure+ data exchange 16

- summary of processing 32
- statistics record
 - SSL and TLS extended option 154
 - viewing for SSL 153
 - viewing for STS 156
 - viewing for STS extended option 157
 - viewing for TLS 153

Sterling External Authentication Server application, function 10

STS

- adding local node record manually 68
- adding remote node manually 74
- configuring parameters manually for STS 67
- resetting keys 151
- securing data 31
- updating keys 150

Summary, processing using Secure+ Option 21

T

Testing Connect:Direct Secure+ Option with STS protocol 185

TLS

- additional security features 20
- data security 20
- levels of security 19
- overview 19
- protocol, defined 19

troubleshooting, Secure+ Option 161

turning security on and off 135

U

Updating, keys 36, 150

Using Secure+ Admin Tool 37

V

Viewing, node record change history 150

W**Worksheets**

- .CLIENT node 184
- .EASERVER node 183
- local node definition 180

Index

remote node definition 181

Z

z/OS

configuring node record for Connect:Direct 205