Reference Manual

truExchange EDI/IP-INT

Version: 5/8/2006

Table of Contents

TABLE OF CONTENTS 2

LIBRARY LIST 4

EDI-INT TRADING PARTNER MENU 6

1. WORK WITH TRADING PARTNER ADDRESSES 6
CREATING A NEW TRADING PARTNER PROFILE: 8
SETTING A TRADING PARTNER'S SECURITY VALUES. 12
TRADING PARTNER AS1 SETTINGS. 16
TRADING PARTNER AS2 SETTINGS. 17
ASSIGNING A CERTIFICATE TO A TRADING PARTNER 19
DEFINING TRADING PARTNER ALIASES 21
2. WORK WITH ADDRESS CROSS-REFERENCE TABLES 22
3. WORK WITH CERTIFICATES 25
SENDING A COPY OF A CERTIFICATE. 25
4. LOADING A TRADING PARTNER CERTIFICATE 28
5. SET TRADING PARTNER DEFAULT 29

THE AS1 MENU. 32

1. WORK WITH SMTP SERVERS 33
CREATING A NEW SMTP SERVER DEFINITION 35
2. WORK WITH POP MAILBOXES 37
CREATING A NEW POP MAILBOX DEFINITION 39

THE AS2 MENU 40

1. START AN AS2 SERVER INSTANCE 42
2. END AN AS2 SERVER INSTANCE 45
3. WORK WITH ACTIVE AS2 SERVER JOBS 46
4. SET AS2 SERVER DEFAULTS 47
5. SET AS2 CLIENT DEFAULTS 51
6. WORK WITH SESSION LOG 54

THE CONFIGURATION MENU. 55

1. WORK WITH PERSONAL KEYS 57
CREATING A NEW PERSONAL/LOCAL KEY 59
2. IMPORT THIRD PARTY KEYS 60
3. SET SYSTEM VALUES 62
4. SET LOCAL MAIL SERVER FOR MESSAGES 64

COMMAND REFERENCE 67

EXPIP2DBF – EXPORT RECEIVED EDI/IP DATA INTO DATABASE FILE. 67
INFORMATIONAL MESSAGES 70

EXCEPTION MESSAGES ____ 70
EXPIP2IFS – EXPORT RECEIVED EDI/IP DATA INTO IFS FILE. ____ 71
INFORMATIONAL MESSAGES ____ 73
EXCEPTION MESSAGES ____ 73
ZCHKEXPCER – CHECK FOR CERTIFICATE EXPIRY. ____ 74
INFORMATIONAL MESSAGES ____ 75
EXCEPTION MESSAGES ____ 75
ZSNDIFSATT – SEND AN IFS FILE TO A TRADING PARTNER. ____ 76
INFORMATIONAL MESSAGES ____ 77
EXCEPTION MESSAGES ____ 77
ZSNDDBFATT – SEND A DATABASE FILE TO A TRADING PARTNER. ____ 78
INFORMATIONAL MESSAGES ____ 80
EXCEPTION MESSAGES ____ 80
PRGSMTPDTA – PURGE DATA FROM EDI/IP SYSTEM. ____ 81
INFORMATIONAL MESSAGES ____ 82
EXCEPTION MESSAGES ____ 82
SNDMSGSMTP – SEND AS1 MESSAGES. ____ 83
INFORMATIONAL MESSAGES ____ 84
EXCEPTION MESSAGES ____ 84
SNDMSGHTTP – SEND AS2 MESSAGES. ____ 85
INFORMATIONAL MESSAGES ____ 86
EXCEPTION MESSAGES ____ 86
ZWRKINMAIL – WORK WITH RECEIVED MAIL. ____ 87
ZWRKOTMAIL – WORK WITH OUTBOUND MAIL. ____ 89

READERS' COMMENTS — WE'D LIKE TO HEAR FROM YOU ____ 91

Library List

The truExchange EDI-INT software is contained in the ZMODEDIIP Library. To have access to the truExchange EDI-INT Software, you will need to have the Library ZMODEDIIP in your Library list.

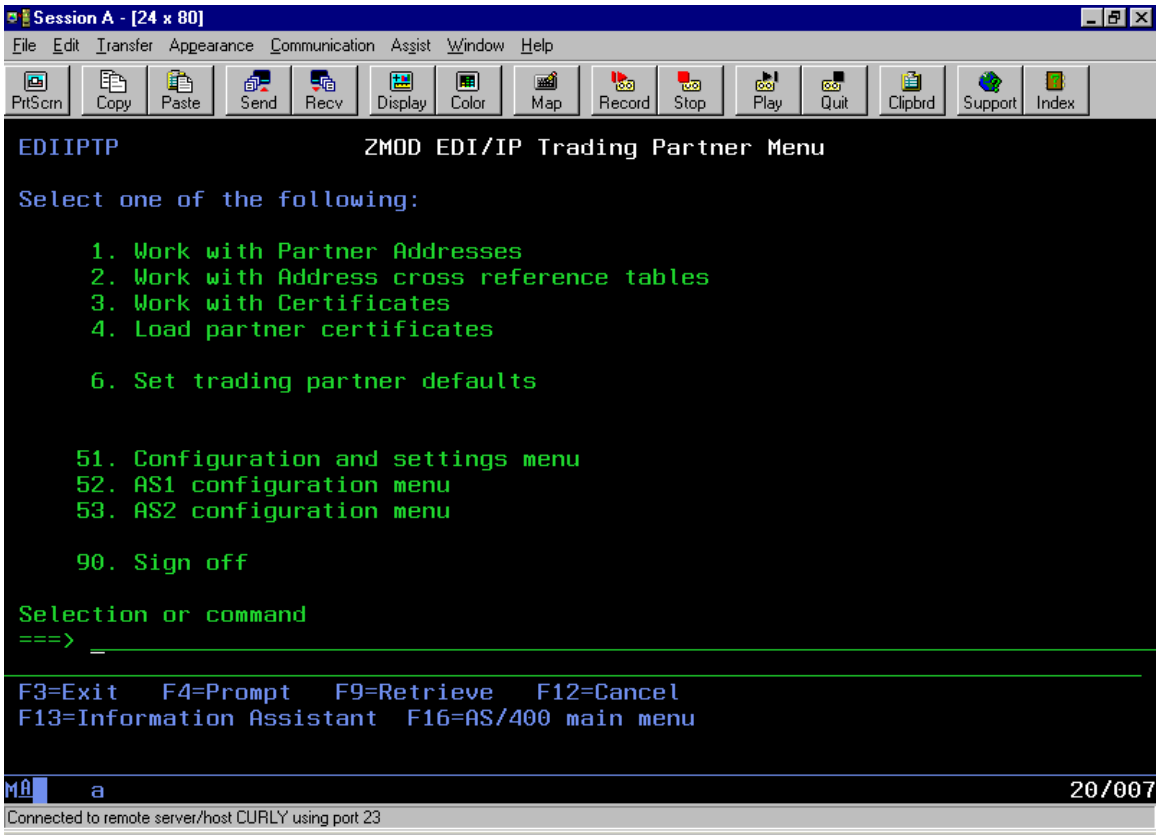
*Please make sure that you add ZMODEDIIP **before** Library ZMODLIB10.*

Library QTEMP needs to be in the Library List at all times.

To avoid any probable issues during test and parallel run with Wal-Mart, please ensure that all relevant USER Library List and JOBD are changed.
truExchange EDI-INT will not work correctly if the Library List is not setup correctly!

The Trading Partner Menu.

The Trading Partner menu will allow you to do most of the EDI/IP configurations that relate to setting up your trading partners. It can be reached by using the command **GO EDIIPTP**.



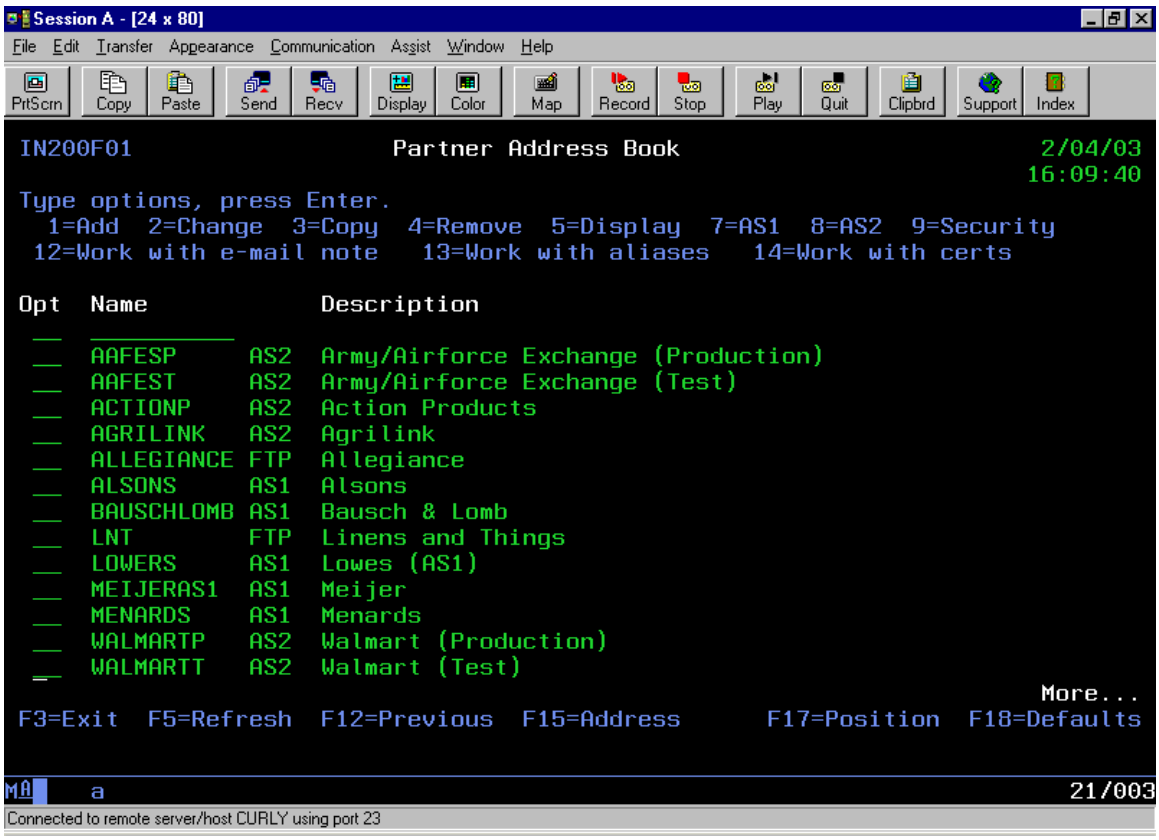
Options available from this menu are:

- *Work with Partner Addresses.* This is the trading partner address book where you create your trading partner profiles.
- *Work with Address Cross-Reference Tables.* This option lets you set up cross-reference tables that associate EDI addresses with trading partner profiles. If you need to split up a file of EDI interchanges to send to one or more trading partners then you'll need to set the partner EDI receiver values here.
- *Work With Certificates.* This option lets you work with the certificate database.
- *Load Partner Certificates.* Use this option to load new certificates into the certificate database.
- *Set Trading Partner Defaults.* There are certain trading partner attributes that can be shared and changed globally. This is where you'd set the defaults that apply to your trading partner profiles.

EDI-INT Trading Partner Menu

1. Work with Trading Partner Addresses

This option is used to view your Trading Partner address book where you create, change and display your trading partner profiles.



The options available from this screen are:

1=Add. Create a new trading partner profile

2=Change. Make changes to the basic elements of the profile definition not including security or transport parameters.

3=Copy. Copy an existing profile

4=Delete. Delete a profile

5=Display. Display the trading partner configuration.

12=Work with email note. If you want to have some set text attached to each message sent to this trading partner then this option lets you edit that text. This option is generally not used in S/MIME environments.

13=Work with Aliases. EDI/IP recognizes received messages as being from a certain trading partner by looking at the sender's transport address (e.g, their email or AS2 address). If a trading partner has multiple email or other transport specific addresses then you can add these aliases here.

14=Work with certificates. To view or assign certificates assigned for this trading partner use this option.

To create a new trading partner profile put a 1 in the top option column and press <enter>.

Creating a New Trading Partner Profile:

Before creating a trading partner definition you should first have collected the following pieces of information:

- A contact email address.
- Their EDI interchange address and qualifier.

For the cryptographic security services:

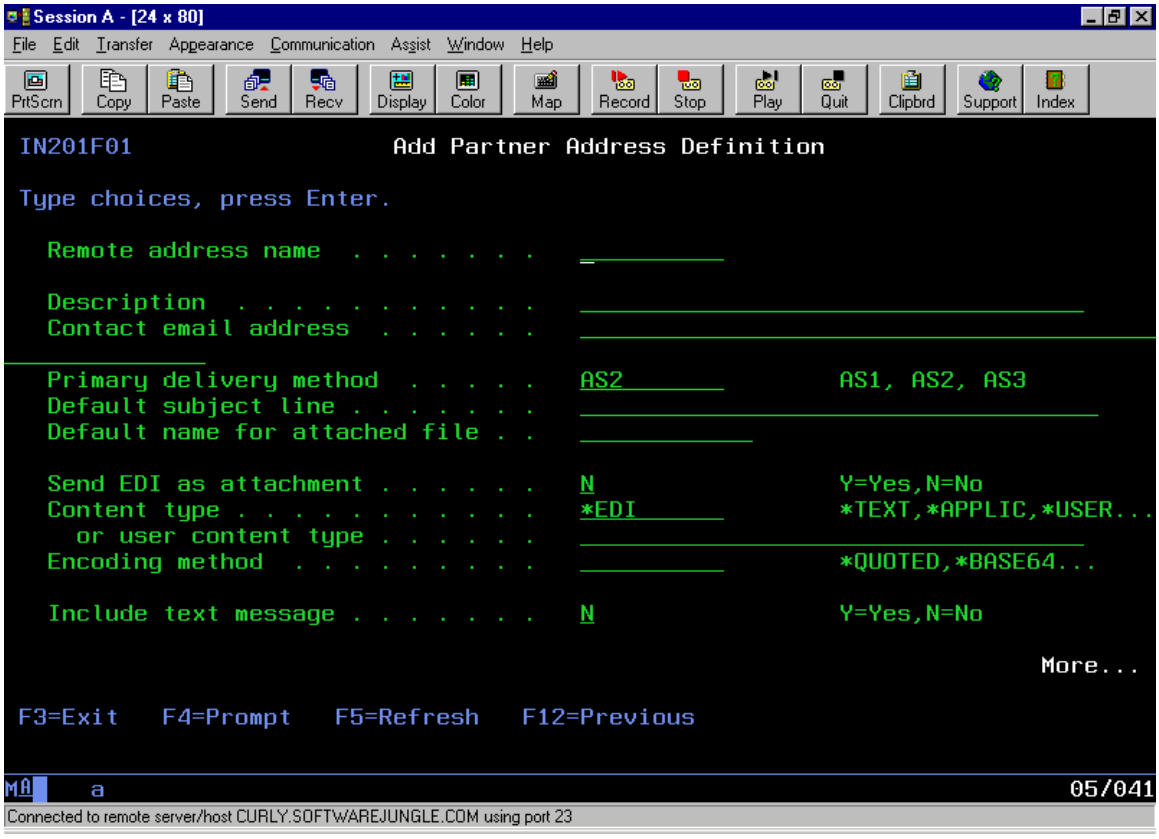
- A copy of your trading partner's certificate.
- The encryption and signing algorithms they would like you to use.

For AS1 trading partners:

- The email address that the secure messages will be sent to.

For AS2 trading partners:

- The URL that you will need to POST secure messages to.
- The AS2 identifier that the trading partner will use in their messages.



To create a trading partner profile, follow these basic steps:

- Load the trading partner certificate if provided.
- Create a trading partner name and description.

- Add the trading partner's AS1 and/or AS2 configuration information.
- Add the trading partner's security requirements.

To load your trading partner's certificate, see the section entitled *"Load Partner Certificates"* below.

Here's a definition of the various parameter setting available on this screen:

Remote address name: This is the short name you give to the trading partner that will be used in the logs and on any commands that have trading partner specific options. Any valid name value will do but it should start with an alpha character and should contain no embedded blanks.

Description: You may use any text that further identifies this trading partner profile.

Contact email address: This should be the email address of your primary point of contact for this application at the remote company. Although this value is not required to create a profile it's recommended that you provide it. It will be needed if you wish to alert your trading partner of certain error events (e.g. a late MDN or failed message delivery) or to mass mail any new local certificates.

Primary delivery method: This value controls the basic transport mechanism used to send messages to this trading partner. Valid values are either **AS1**, **AS2** or **AS3**. You can change the transport method simply by modifying this field. Changes to this value will take effect on messages created after the change. *Note: you should also ensure that you define any parameters necessary for the chosen method such as: providing an email address for AS1 and an id for AS2.*

If left blank some of the values that follow will default to the system default settings that can changed by selecting F18 from the "Partner Address Book" panel.

Default subject line: This is the subject line that will be included on all outbound messages. If this is left blank it will default to the system default value.

Default name for attached file: When dealing with batches of EDI interchanges or database files the concept of a file name for the interchange or data set is somewhat contrived. Some products however require an "attachment" name in order to correctly route messages on their end. This parameter is the default file or attachment name that we put on messages to this trading partner (usually something like data.edi will do). If you're using the ZSNDIFSATT (send an IFS file) or ZSNDDBFATT (send a database file) commands then you can override the attachment name on a file-by-file basis. If this is left blank it will default to the system default value.

Send EDI as attachment: For S/MIME applications leave this as 'N'. For non-S/MIME applications this flag controls whether the data is sent as an attachment to the mail item or in-line. If this is left blank it will default to the system default value.

Content-type: If this is left blank it will default to the system default value. This is the default content type on the message used primarily for the DSTEDIDTA command. For most S/MIME applications it should be set to *EDI. Valid values are:

- *TEXT – text/plain
- *APPLIC – application/octet stream
- *EDI – one of either application/edi-x12, application/edi-edifact or application/edi-consent depending on the EDI standard used.
- *MSWORD – Microsoft word document
- *USER – enter a value under the user content type.

Encoding method: If you intend to encrypt and/or compress the data you're sending or if you're using a transport mechanism other than SMTP (AS1) then you can set this value to *BINARY. This parameter controls the data encoding method used on the actual data to be sent. Typically encoding is used where the data is to be sent over a transport that does not guarantee a clear 8 bit channel. SMTP falls into this category. Valid options are *QUOTED, *BASE64, *BINARY, *7BIT, *8BIT. If this is left blank it will default to the system default value.

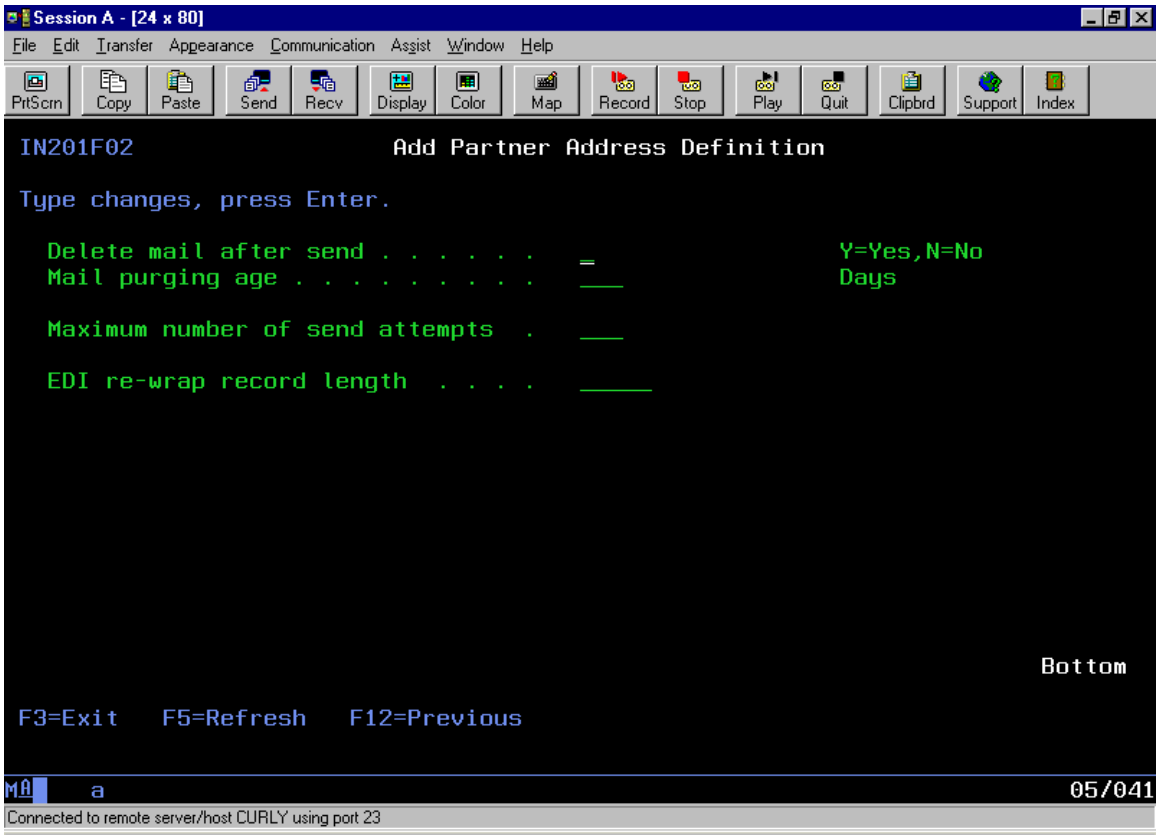
Include text message: If you've set up either a system wide default text message or a partner specific text message that you want prefixed to the data or attachment you're sending then answer 'Y' here. For S/MIME applications this would typically be 'N'.

Press rollup for more options

Delete mail after send: You have two options when it comes to purging mail that has been sent or inbound mail that has been delivered. If this is left blank it will default to the system default value. This one controls whether the actual mail data is automatically deleted after the message has been sent or processed. The mail header record will remain in the log. Setting this value to 'Y' means that the mail data will be permanently removed as soon as the message has been sent. You will not be able to re-send/re-process this message unless you re-create it from the original EDI/file data or re-receive it. The alternative is to use the **PRGSMTPDTA** command to schedule purge requests at appropriate intervals.

Mail purging age: An alternative to immediately deleting processed data (sent or received) is to use the PRGSMTPDTA command. This parameter allows you to set an aging period, expressed in days, after which processed mail is eligible for purging by this command. This value can be overridden on the PRGSMTPDTA command. If this is left blank it will default to the system default value.

Maximum number of send attempts: This is the number of attempts that are made to successfully deliver a mail item to the immediate destination site. For AS2 and AS3 this means delivery to the URL specified on the trading partner definition. For AS1 it means delivery to the relevant SMTP server. If an unsuccessful attempt is made to deliver a mail item then it will be re-tried the next time a send command is run until this number of attempts has been exceeded, whereupon the mail item will then be marked in error. If this is left blank it will default to the system default value.

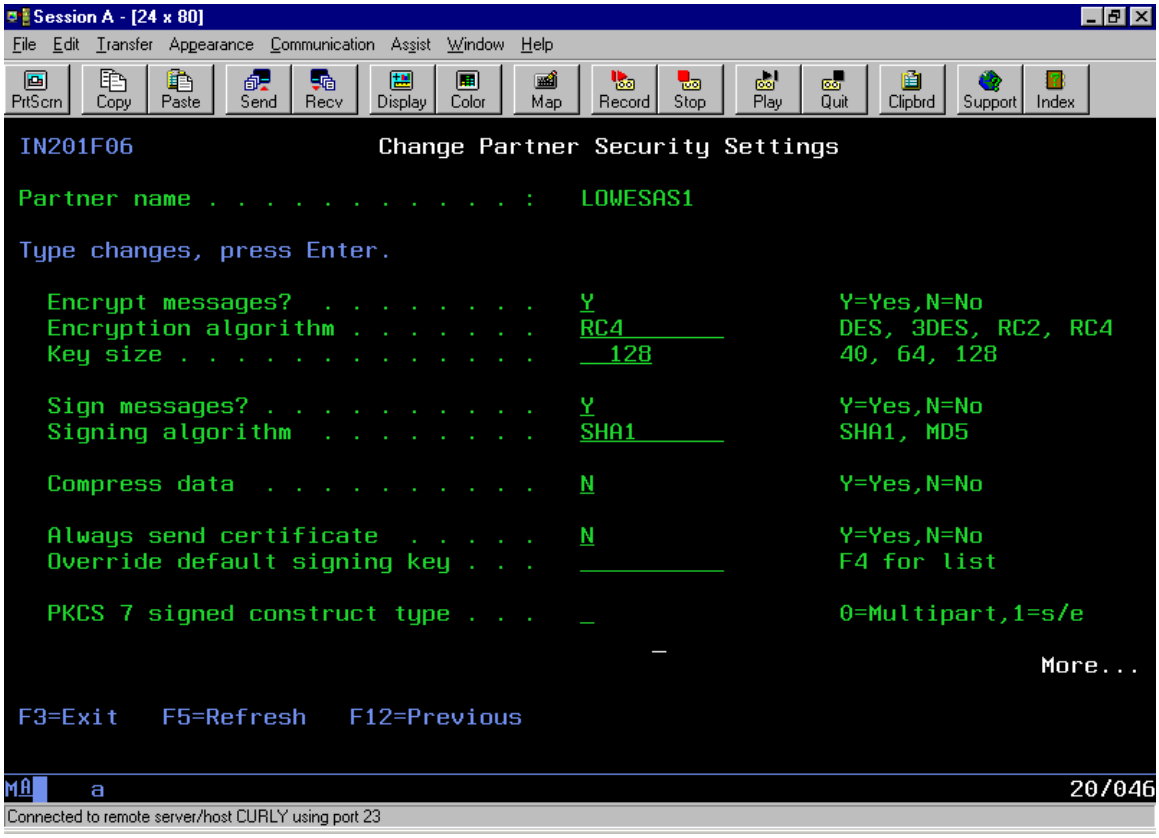


EDI re-wrap record length: This applies to EDI data that's processed through the DSTEDIDTA (distribute EDI data) command. Generally when EDI data is parsed and encapsulated into a message it's treated as a data stream. Rarely, a trading partner might require that the data appear to them as a file of "blocked" EDI data. By this we mean that the EDI data consists of a series of definite length records, usually 80 or 132 byte. If this trading partner requires this then you can set the record length they want here.

This completes the setup of a TP profile header. You'll now need to setup the TP's transport related information (AS1, AS2, AS3 values) and change their security requirements.

Setting a Trading Partner's Security Values.

Choose ***option 9*** beside a TP profile from the Partner Address Book panel to set a TP's requirements for encryption, signatures, MDNs and related information. You should see the screen shown below.



Encrypt messages?: Enter 'Y' if you want to encrypt messages sent to this trading partner otherwise set it 'N'.

Encryption algorithm: If the above option is set to Y then this parameter controls the encryption algorithm to use. If your trading partner has specified what algorithm they want you to use for data sent to them then select that option. Valid options are DES (56bit DES), 3DES (168bit triple DES), RC2 (specify key size), RC4 (specify key size).

Key size: If the chosen algorithm is RC2 or RC4 then specify the key size to use. Valid values are in the range 40-1024 for RC2 and 40-2048 bytes for RC4.

Sign messages: If you want to digitally sign messages sent to this trading partner then enter 'Y' here.

Signing algorithm: If you've selected to sign messages for this trading partner then this is the algorithm that will be used to compute the signature. Valid options are SHA1 and MD5.

Compress data: To have the message data zipped before sending then select 'Y' here.

Always send certificate: If the TP's product requires that you attach a copy of your certificate with every signed message then enter a 'Y' here otherwise it's best to choose 'N'.

Override default signing key: In most circumstances you'll probably only have one key that you use for signing data for all your trading partners. If you need to use a different key for signing messages sent to this trading partner then you can select that key name. Use *F4* to get a list of available keys.

PKCS7 signed construct type: This is quite a technical parameter and is used to change the format of the secure messages you send to your trading partner. Unless instructed otherwise by support you should leave this value as a blank or zero. Some older S/MIME products that are not certified AS1 or AS2 cannot read the multi-part signed messages that these specifications require. Setting this parameter to a 1 alters the format of secure messages generated from "encrypted multi-part signed" to "signed encrypted".

Press rollup for options related to delivery notifications.

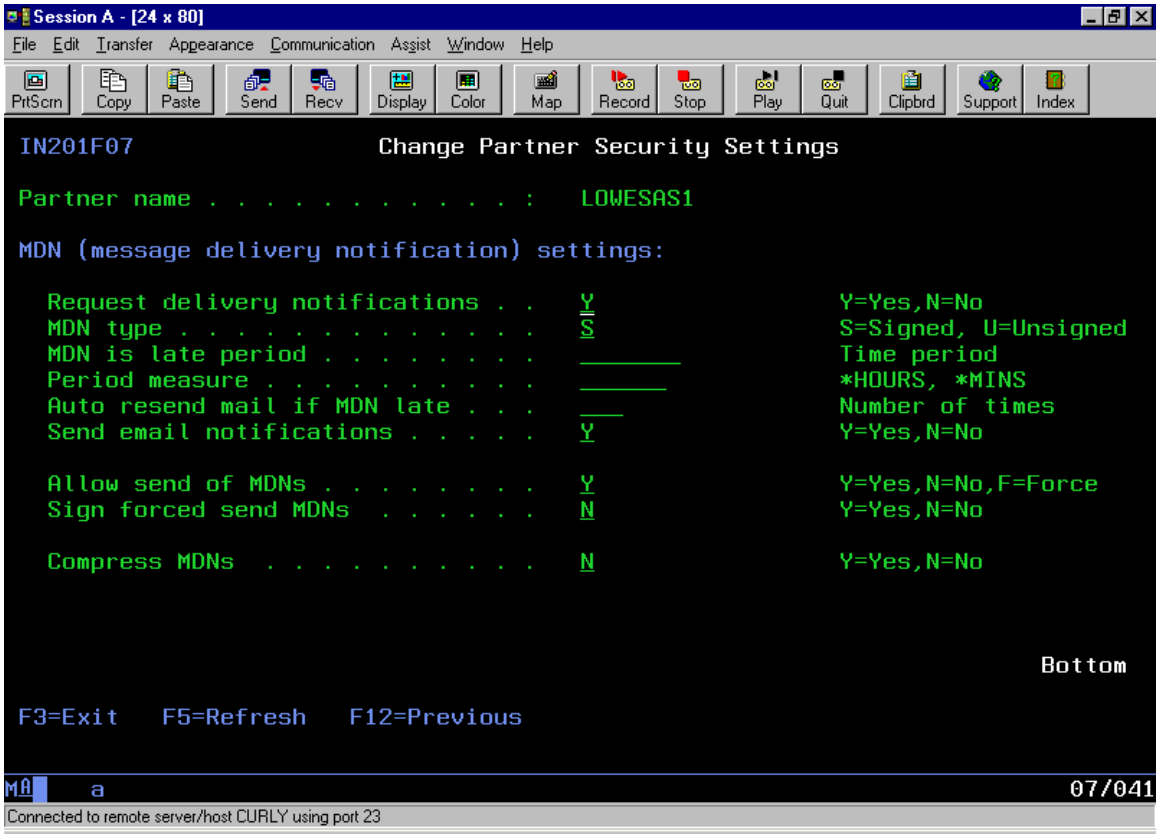
A message delivery notification (MDN) is a document that's returned from your trading partner to confirm the receipt of a message you sent. It will show whether they were able to decrypt the message and whether they could verify that it was signed by you.

Request delivery notification: MDNs are only returned when the sender requests them. If you want to request MDNs from this TP for messages that you send then enter 'Y'.

MDN type: MDNs can be either signed (the trading partner will sign the contents so you can prove they got the message) or unsigned (the MDN will just show receipt and status). To request a signed MDN enter an 'S', for unsigned MDNs enter a 'U'.

If left blank some of the values that follow will default to the system default settings that can changed by selecting F18 from the "Partner Address Book" panel.

MDN is late period: Use this value to provide a time period after which EDI/IP will consider any MDN not yet received to be late. Some transport mechanisms (AS2) have the option of requesting that the MDN be returned immediately in which case this value would be moot. The measure for the value you enter here is taken from the next parameter.



Period measure: Enter either *HOURS or *MINS to provide a measure for the "MDN is late period" parameter above.

Auto re-send mail if MDN late: This action is only triggered if the ZCHKACKSTS (check acknowledgement status) command encounters a sent message for which an MDN has not been received within the specified period. If a message is found with a late MDN then that message will be automatically released for re-send up to the number of times given here after which a permanent "MDN late" error will be signaled.

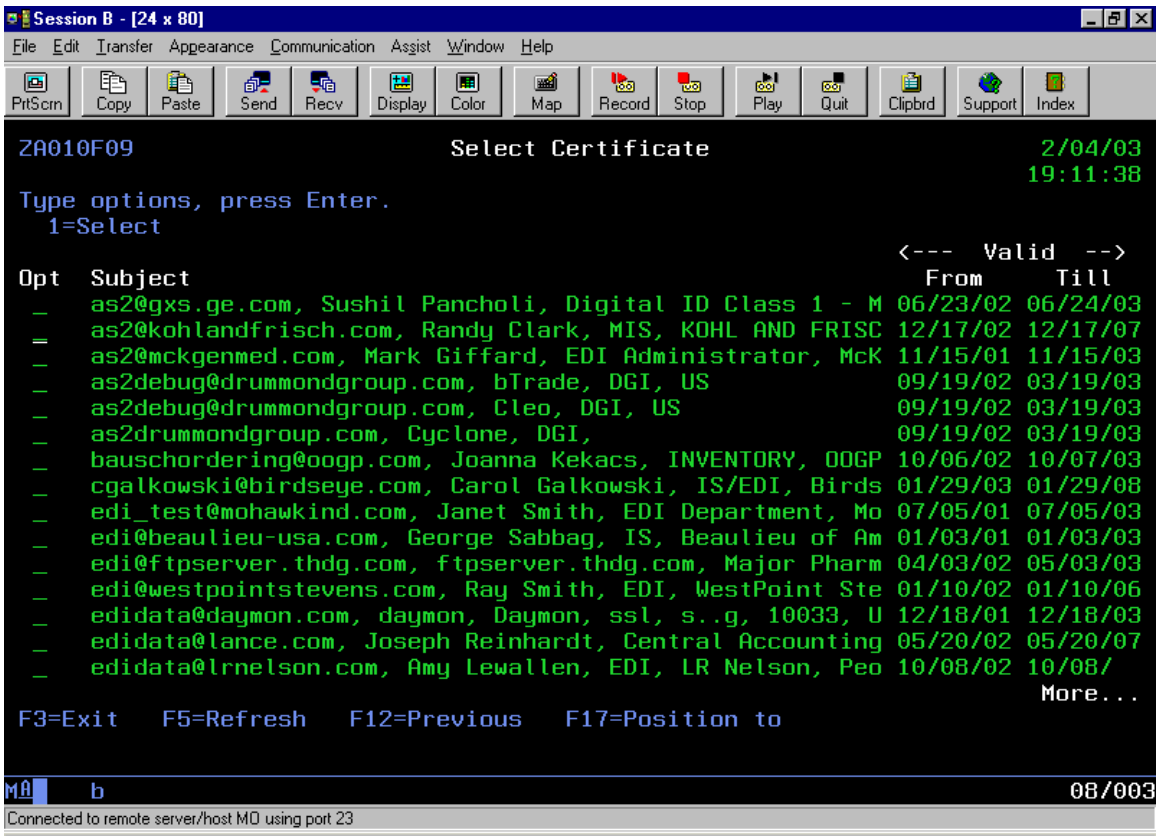
Send email notifications: If you've got a contact email address for this trading partner then setting this field to a 'Y' will result in MDN late notifications or delivery error messages to be automatically sent via email to the remote TP contact as well as locally.

Allow send of MDNs: Valid values are 'Y', 'N' or 'F'. For incoming messages this parameter can override the TP's request for an MDN. This setting is mostly used in AS1 scenarios where a local (or intermediate) mail server or firewall is set to strip mail headers it doesn't recognize. In these cases MDN request headers are often removed and never make it into EDI/IP so an MDN isn't generated. Setting this flag to an 'F' will force EDI/IP to generate an MDN for the TP's message regardless of whether the request header is found. A setting of 'N' will cause no MDNs to get generated. The 'Y' value will let EDI/IP generate an MDN if the sender requests one.

Sign forced-send MDNs: If the above field is set to F then this controls whether the MDN is signed or not. Using 'Y' will result in a signed MDN and 'N' in an unsigned one.

Compress MDNs: This option can be used in an AS1 setting if signed MDNs are being corrupted by the mail server. The TP's software must support compressing of MDNs too. Experience has shown that a compressed MDN normally escapes corruption through mail servers and firewalls.

If you entered Y for "Encrypt messages" and the trading partner doesn't already have a certificate associated with them you should now be presented with the "Select Certificate" screen shown below.

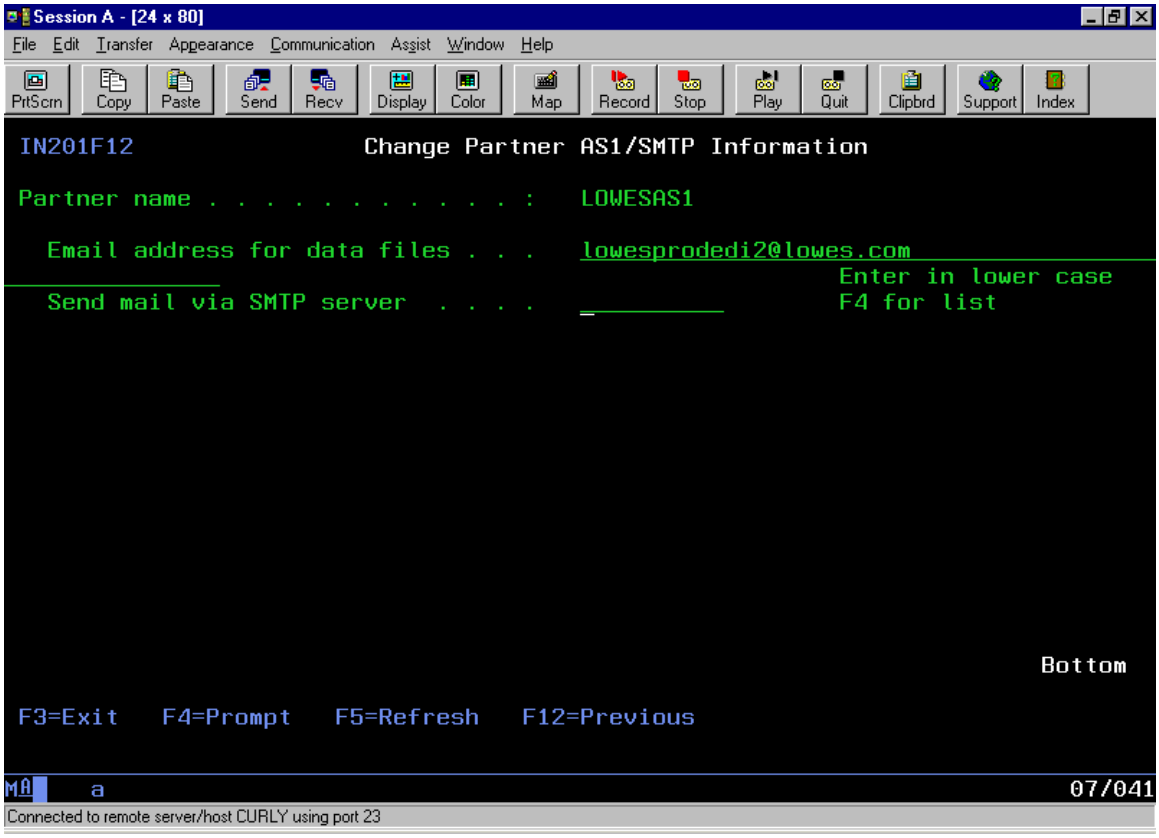


From the "Select Certificate" screen find the trading partner's certificate. If you have not already loaded the TP's certificate then just press <enter> and you can assign it later. Enter a 1 (select) beside the certificate and press <enter>. The certificate is now linked with this trading partner. You can work with certificates linked to this TP by using option 14 from the "Partner Address Book" panel.

This completes the setup of the TP profile security information.

Trading Partner AS1 settings.

Choose ***option 7*** beside a TP profile from the Partner Address Book panel to set a TP's email address and server information for AS1. The panel for trading partner AS1 information is shown below.



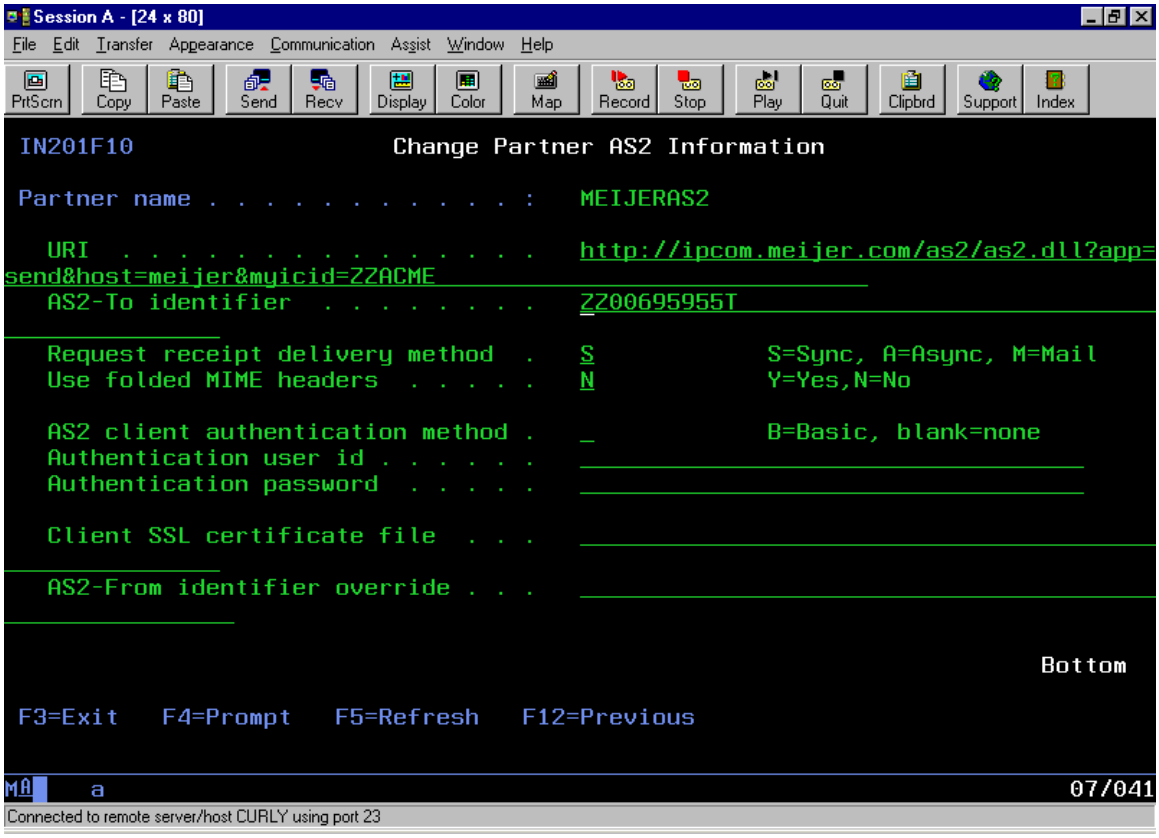
Email address for data files: This is the email address that the trading partner has given you to use when sending messages to them using AS1. You should usually enter it in lower case.

Send mail via SMTP server: If you leave this name blank then the SMTP server to deliver this mail to will default to the system default server. If you need to deliver mail to this partner over a specific SMTP mail server then enter that name here. You can use *F4* to select one of the existing server configurations.

This completes the setup of the TP profile AS1 information.

Trading Partner AS2 settings.

Choose ***option 8*** beside a TP profile from the Partner Address Book panel to set a TP's AS2 parameters. The panel for trading partner AS2 information is shown below.



URI: Your TP should have given you the URL of their AS2 server for you to connect to. Enter that value here.

AS2-To identifier: Your TP will also have given you their AS2 identifier. That's the value you enter here. If the AS2 identifier includes embedded spaces then enter the id with double quotes around it.

Request receipt delivery method: With AS2 there are three options as to how you would like the MDNs for messages you send to be returned to you. *Synchronous* delivery means that the AS2 client delivering the mail will wait for the MDN. The message will have to be processed (e.g. decrypted, decompressed etc) before the MDN can be created. *Asynchronous* delivery is where the AS2 client delivers the message and gets an OK but doesn't wait for the MDN. The MDN will be delivered at a later time to your AS2 server. The *mail* option is similar to the asynchronous option except that the MDN will be delivered to your mail server in an AS1 fashion.

Use folded MIME headers: Experience has shown that most AS2 servers support unfolded headers. Unless you know that the TP's AS2 server requires folder headers you should enter a 'N' here.

AS2-From identifier override: When you send AS2 messages to your TP the AS2 id that you use to identify your company usually comes from the "AS2 client defaults" setting for "AS2 From Address". If this TP requires that you use a specific value that they assign then you can enter that value here and it will be used instead.

The remaining fields on this screen are only relevant if the AS2 connection that you will make to your TP uses SSL (i.e. if the URI begins with HTTPS://)

AS2 Client Authentication Method: EDI/IP supports **Basic** authentication for logging into HTTP servers. Most AS2 SSL connections do not require client authentication, however if the AS2 server you're connecting to does then set this parameter to a 'B'. You'll be required to set the user id and password values below.

Authentication User Id: If you need to use Client Authentication to connect with this TP's SSL site then enter the user id they've given you here.

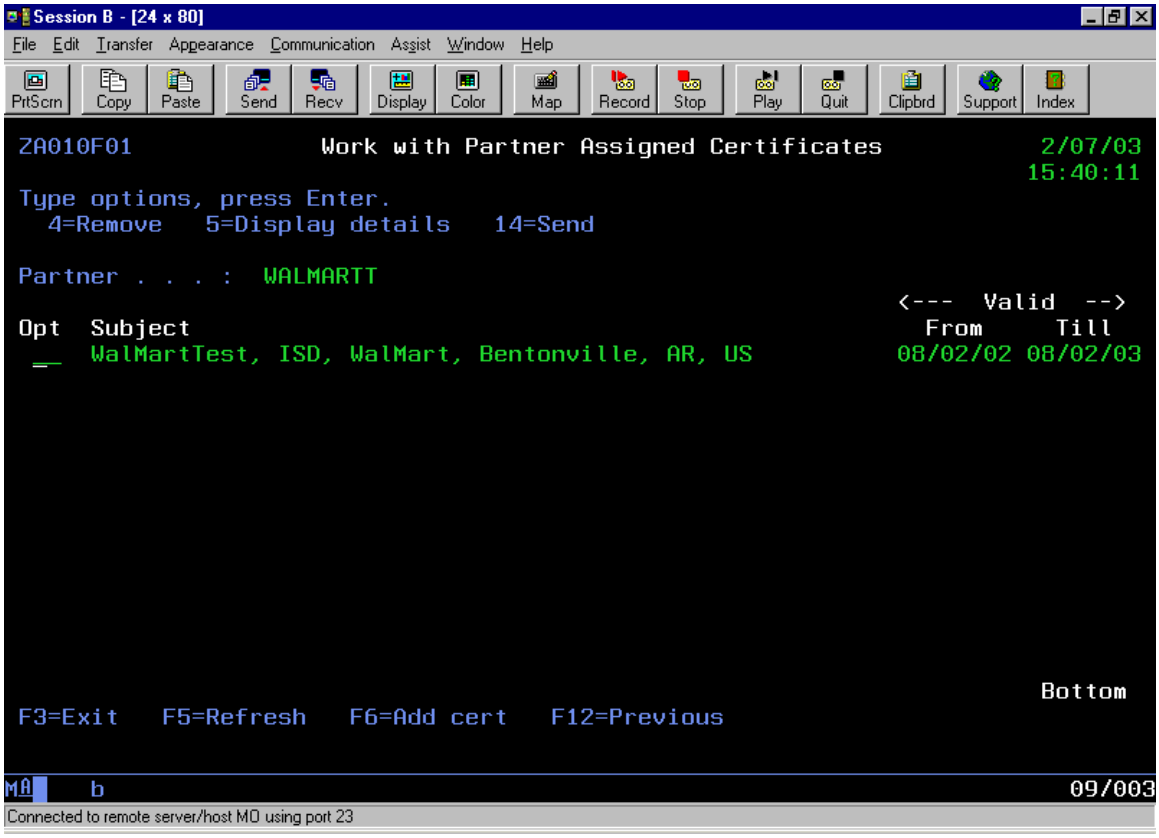
Authentication Password: Enter the password for the above user id.

Client SSL certificate file: If you need to use SSL Client Authentication to connect with this TP's SSL site then enter the path to the certificate you'll be using for this purpose.

This completes the setup of the TP profile AS2 information.

Assigning a Certificate to a Trading Partner

Before you can assign a certificate to a trading partner you must first have loaded it into EDI/IP. See the section on *"Loading a trading partner's certificate"* for information on how to do this.



You can either assign a certificate to a trading partner from the *"Work with Certificates"* option (described later) or by using ***option 14*** from the "Partner Address Book" screen. You can have multiple certificates assigned to a single trading partner. Normally EDI/IP will choose the first valid certificate it can find for that trading partner to use when encrypting data to send to them.

The options available from this screen are:

4=Remove. Remove the trading partner/certificate assignment. The certificate will not be deleted from the certificate database. To delete certificates from the EDI/IP database use the delete function under the *"Work with Certificates"* option.

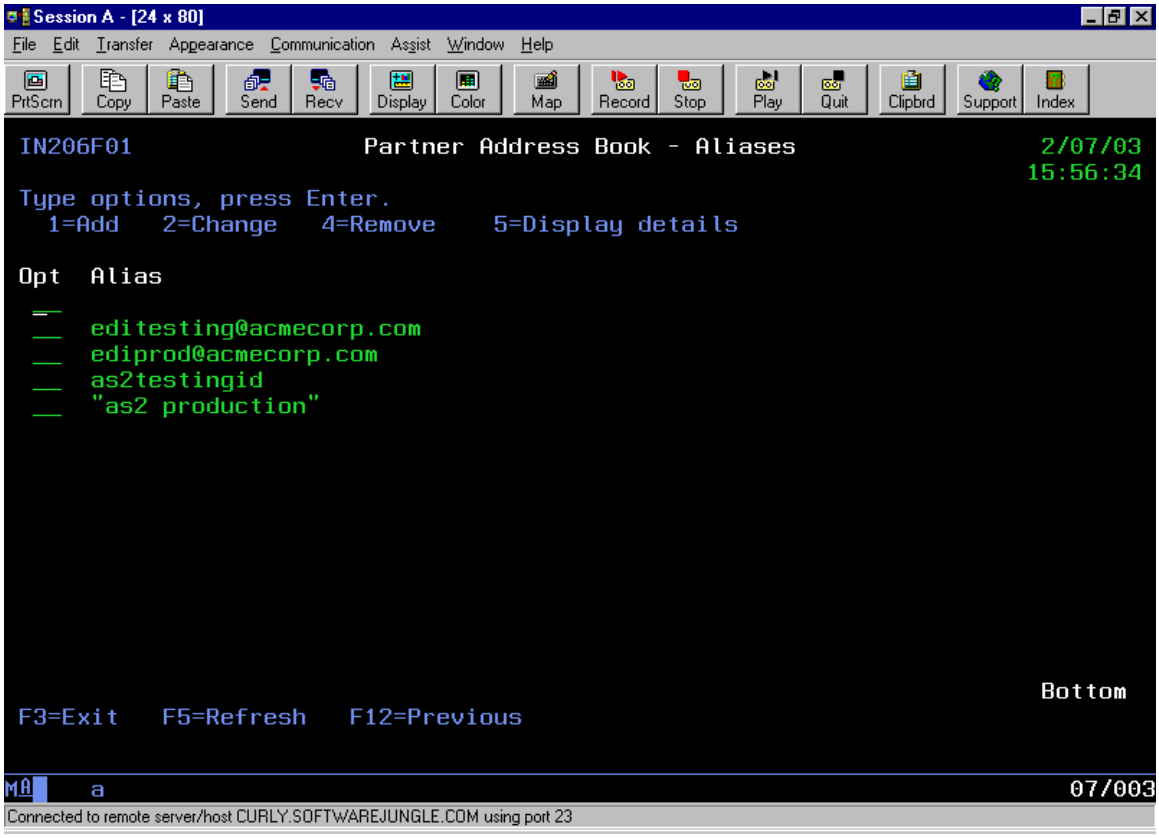
5=Display. Display the certificate details such as serial number, issuer etc.

14=Send. To email a copy of this certificate, choose this option and fill in the intended receivers' email address. This option is discussed fully in the *"Work with Certificates"* section.

Use ***F6*** to assign a certificate to this trading partner. Select the appropriate certificate from the certificate database and press <*enter*>.

This completes the assignment of a trading partner certificate.

Defining Trading Partner Aliases

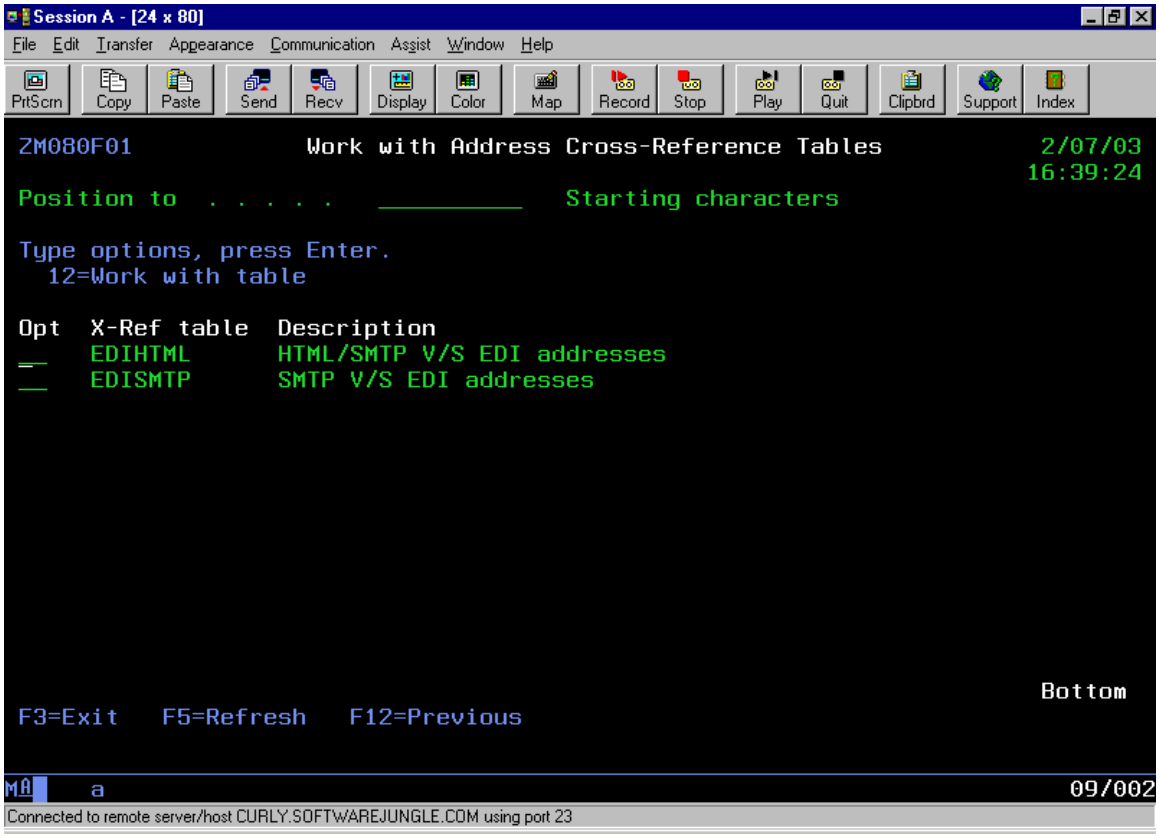


In most cases your trading partner will only have a single AS1, AS2 and/or AS3 identifier so you won't need to worry about setting up alternative id's in the alias table. Sometimes (e.g. testing) you may want to associate multiple identifiers with a single trading partner profile. The EDI/IP trading-partner-alias option lets you set up these multiple identifiers. To set up alternative identifiers for a TP profile select ***option 13*** from the "Partner Address Book" screen.

Both AS1 email addresses and AS2 identifiers can be mixed as aliases. If the AS2 identifier includes embedded spaces then enter the id with double quotes around it.

This completes setting up alternative trading partner identifiers.

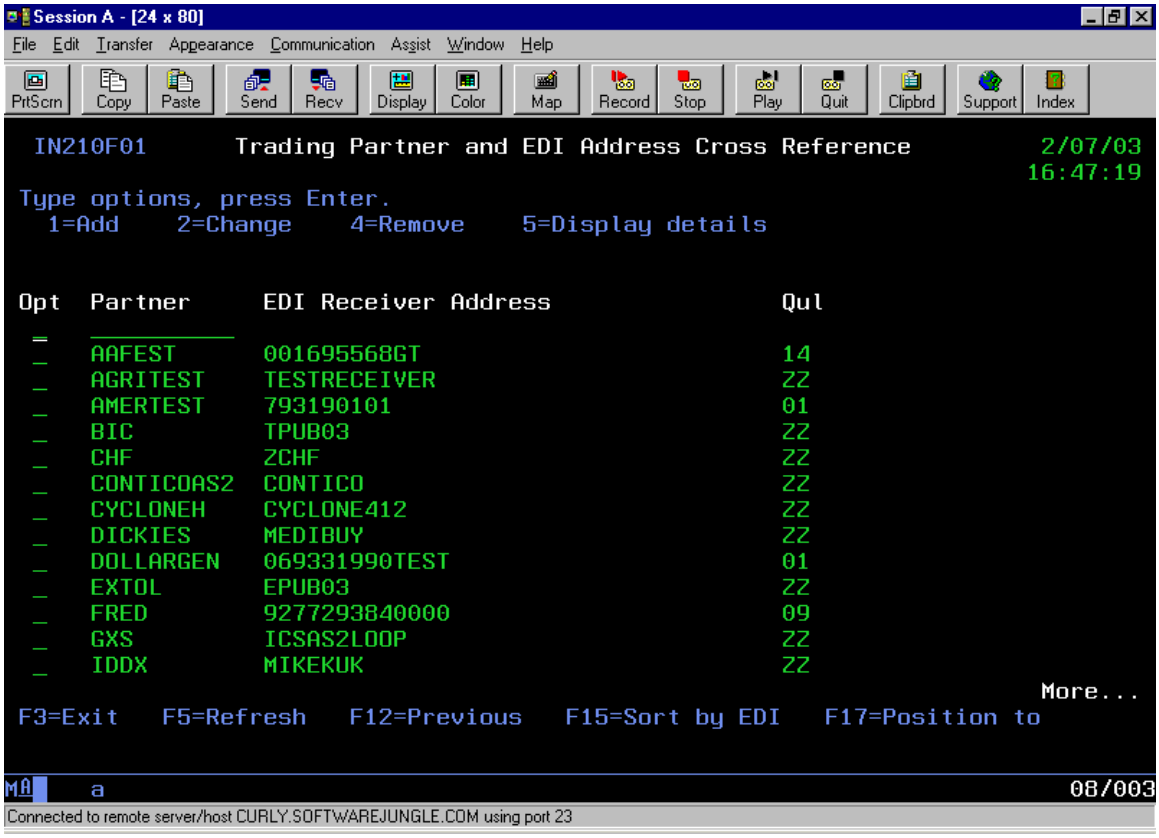
2. Work with Address Cross-Reference Tables



This option is used to associate an EDI receiver address with an EDI/IP trading partner profile. You will need to use this option if you want to use the DSTEDIDTA (Distribute EDI data) command to parse and split a file of EDI interchanges and assign each interchange to a trading partner. If you're not sending EDI data or if you want to send a file of EDI interchanges to a single trading partner profile then you don't need to set up an EDI address cross-reference.

Depending on which products you have loaded you may see more than one type of cross-reference table shown on the screen above. For a standard EDI/IP configuration choose the EDISMTP cross-reference table by entering ***option 12*** and press <enter>. *(The SMTP part of the table name is historical and has no bearing on the transport method used).*

The EDI parsing function of the command DSTEDIDTA scans either a file of EDI data or your EDI application outbox for interchanges. The EDI receiver's address and qualifier are extracted and looked up in this table. This should then provide a match to an EDI/IP trading partner profile. The interchange will then be encapsulated according to the TP's profile and placed on the EDI/IP outbound queue for transmission. If an interchange is found that does not have a matching cross-reference, a detailed error message will be posted to the broadcast system and the interchange will be dropped.



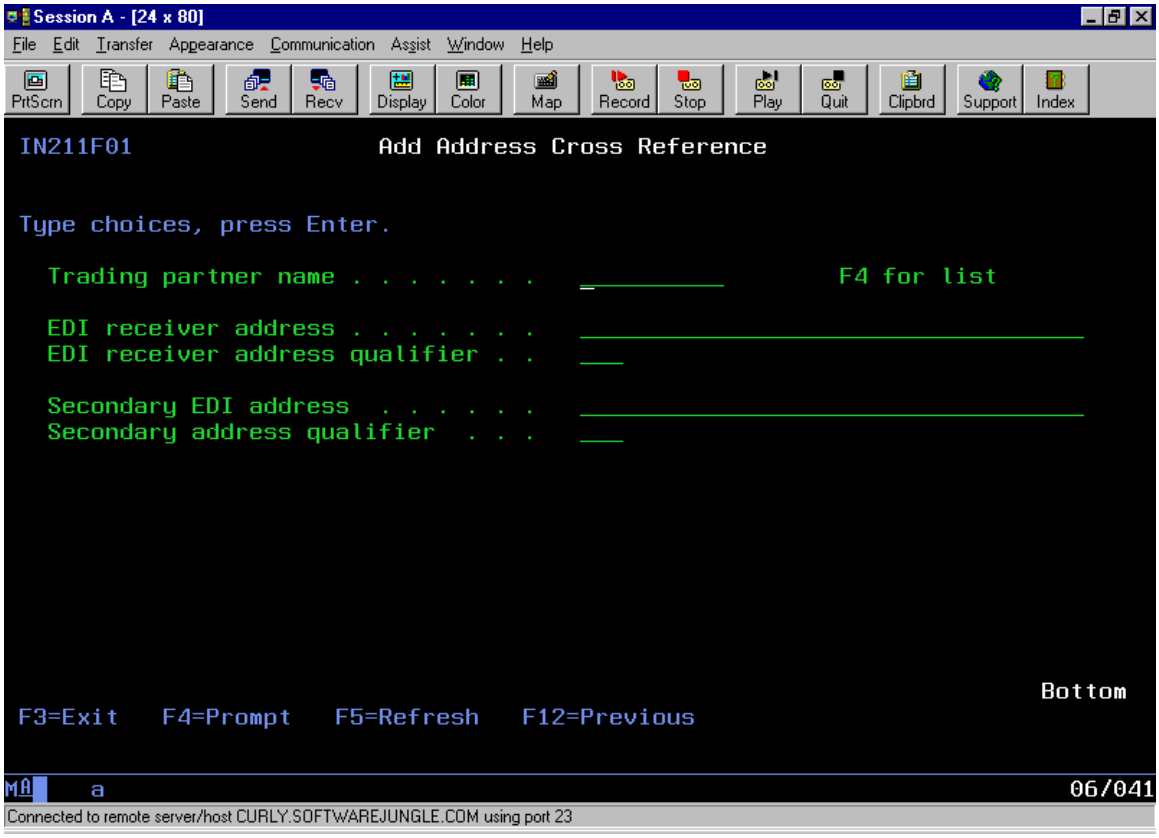
To add a new TP/EDI address cross-reference enter a “1” in the top column and press *<enter>*.

Trading Partner Name: Enter the name of the trading partner or use *<F4>* to select from a list of defined partners. A TP may have multiple EDI addresses associated with it.

EDI receiver address: This is your trading partner’s EDI id that appears at the interchange level. (e.g. for X12 it would be the value in ISA08.). The special value of *ANY (must be upper case) can be used to link any EDI id that doesn’t have a more specific match to this TP profile.

EDI receiver address qualifier: This is the qualifier for the EDI id given above. (e.g. for X12 it would be the value in ISA07.)

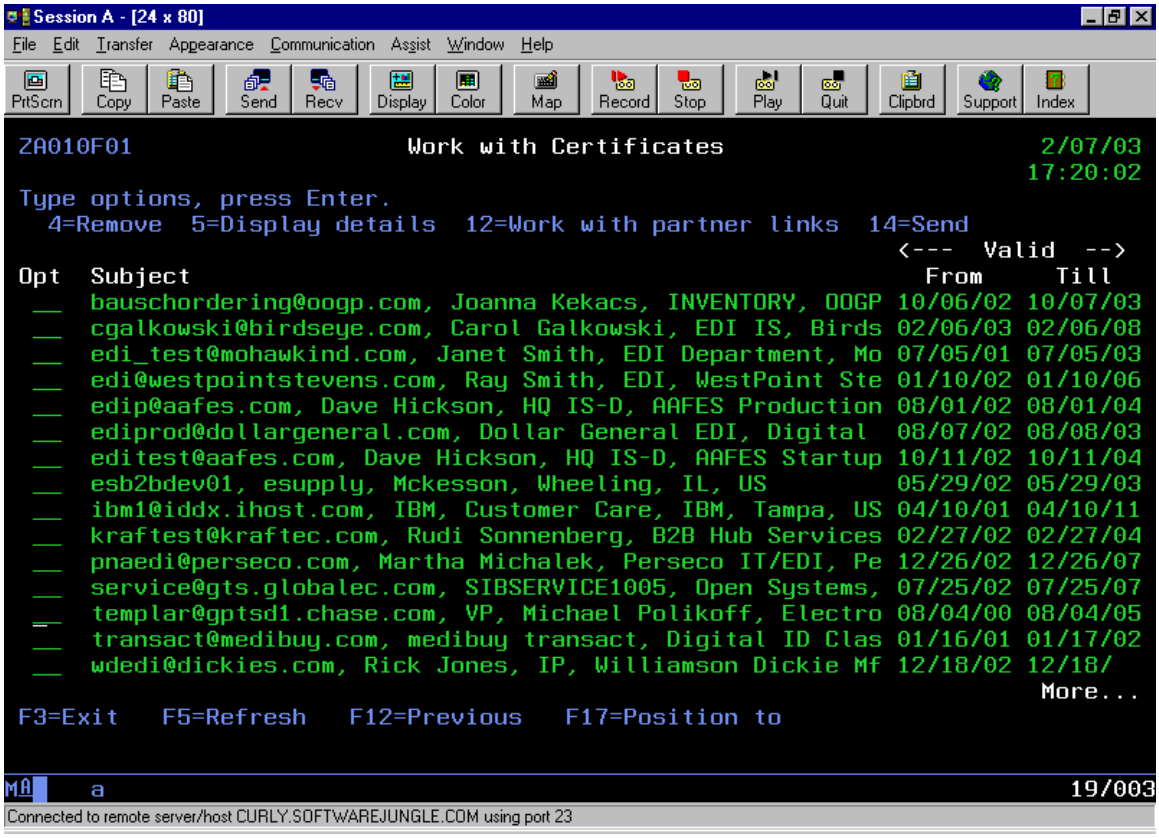
Secondary EDI address: This option is no longer used.



This completes setting up EDI Identifier cross-references.

3. Work with Certificates

This option shows you the certificate database. To load a certificate you must use the command **ZLODDCERT** or use ***option 4*** from the Trading Partner menu (EDIIPTP).



The options available from this screen are:

4=Remove. This option will delete the certificate from the database and remove any trading partner links to this certificate.

5=Display. Display more of the certificate details such as the issuer and serial number.

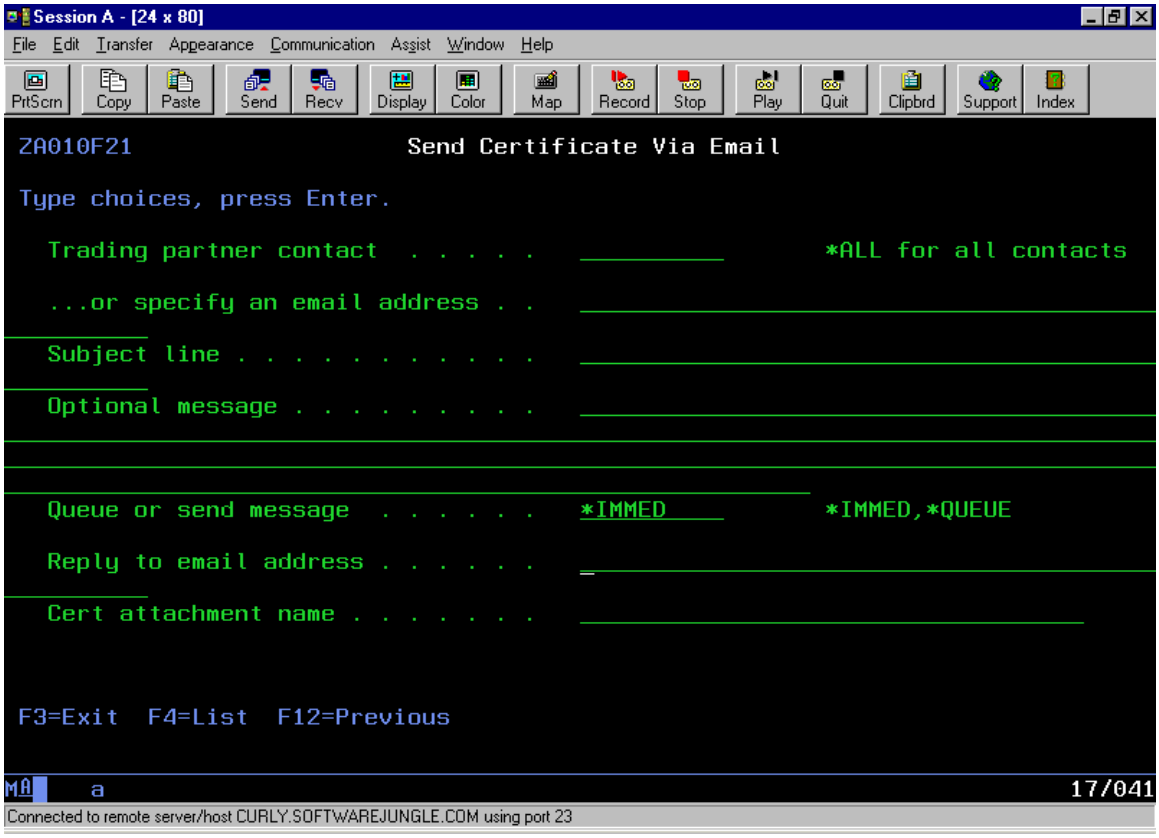
12=Work with partner links. This option will show you all trading partner profiles that are associated with a certificate. From this screen you can add and remove trading partner/certificate links.

14=Send. This option allows you to send a copy of a certificate via email.

Sending a copy of a certificate.

You can use option 14 from the "Work with Certificates" panel to send a copy of a certificate either to a single email address, a trading partner contact address or all trading partner contacts. The certificate will be received as an attachment.

You must have set up the local mail server from the configuration menu (EDIIPCFG) before you can email certificates.



Trading Partner Contact: Enter the name of the trading partner profile that has the contact email address of the person you want to email a copy of this certificate to. Use *<F4>* to get a list of trading partner profiles. You may use the special value of *ALL to email a copy of this certificate to all your trading partner contacts.

Or specify email address: If you don't enter a trading partner profile above you may enter the email address of the person you want to send a copy of this certificate to.

Subject line: You must enter something that will appear as the email subject line.

Optional message: You can enter some text that will be sent along with the certificate.

Queue or send message: You have the option of having this email sent immediately or queued till pending email is flushed.

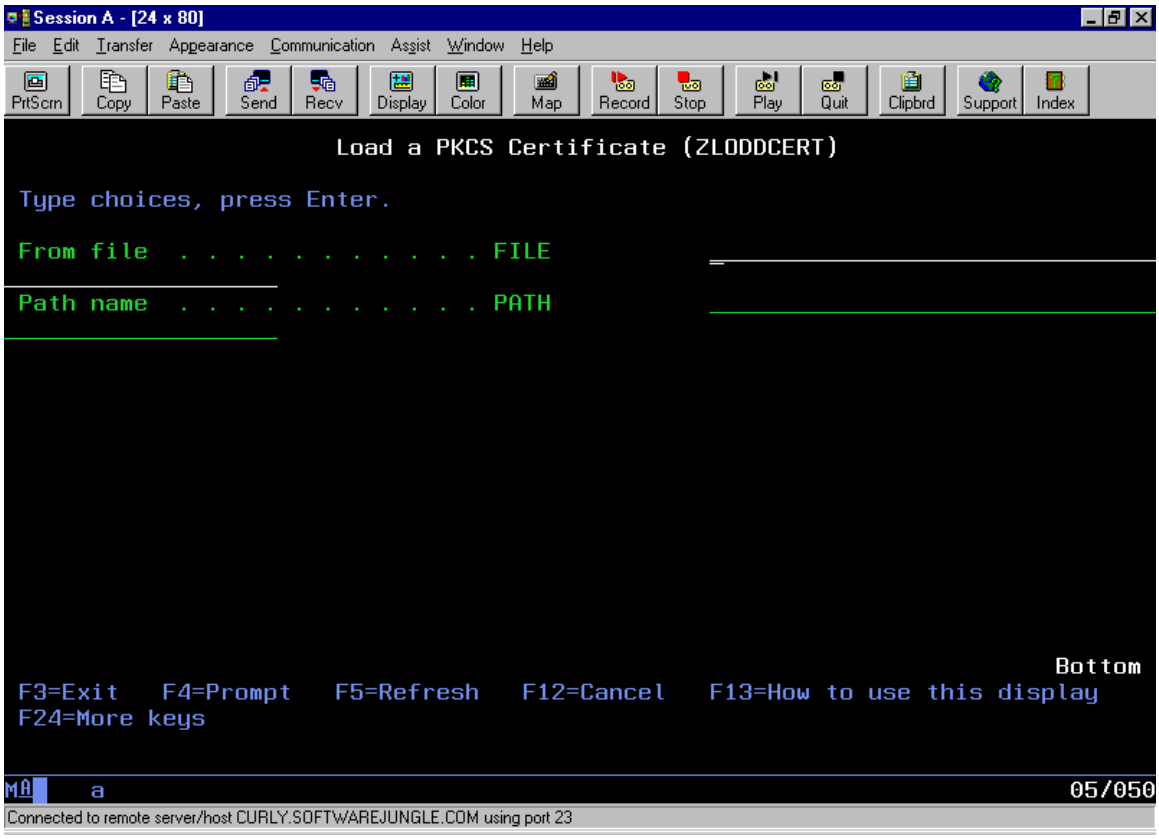
Reply-to email address: This value defaults to the setting on the local email options and is the reply address on the email you send. You may change it to any valid email address.

Certificate attachment name: Usually certificates are stored in the EDI/IP database with a name something like zcert12.cer. This is the default name that the attachment will be given. If you want to provided a more descriptive certificate file attachment name then enter one here (e.g. acmecorpAS2.cer).

4. Loading a Trading Partner Certificate

Your trading partner will usually send you a copy of their certificate via email. This will most likely be an attachment with an extension of either: **CER**, **PEM**, **P7B** or **DER**. (If the certificate has some other extension you can try to load it but it might not be a supported format). Before you can load this certificate in the EDI/IP product you must first save this file onto the iSeries machine somewhere in the IFS file system. You may save this file into any folder that you have authority to and with any name that you can remember. Once the certificate is loaded into EDI/IP you do not need to keep the copy that you save – EDI/IP will maintain its' own copy in its' certificate database. If you don't have an IFS folder in which to save this certificate then you can always use the folder "/ZMODEDIIP" that's created as part of the EDI/IP installation.

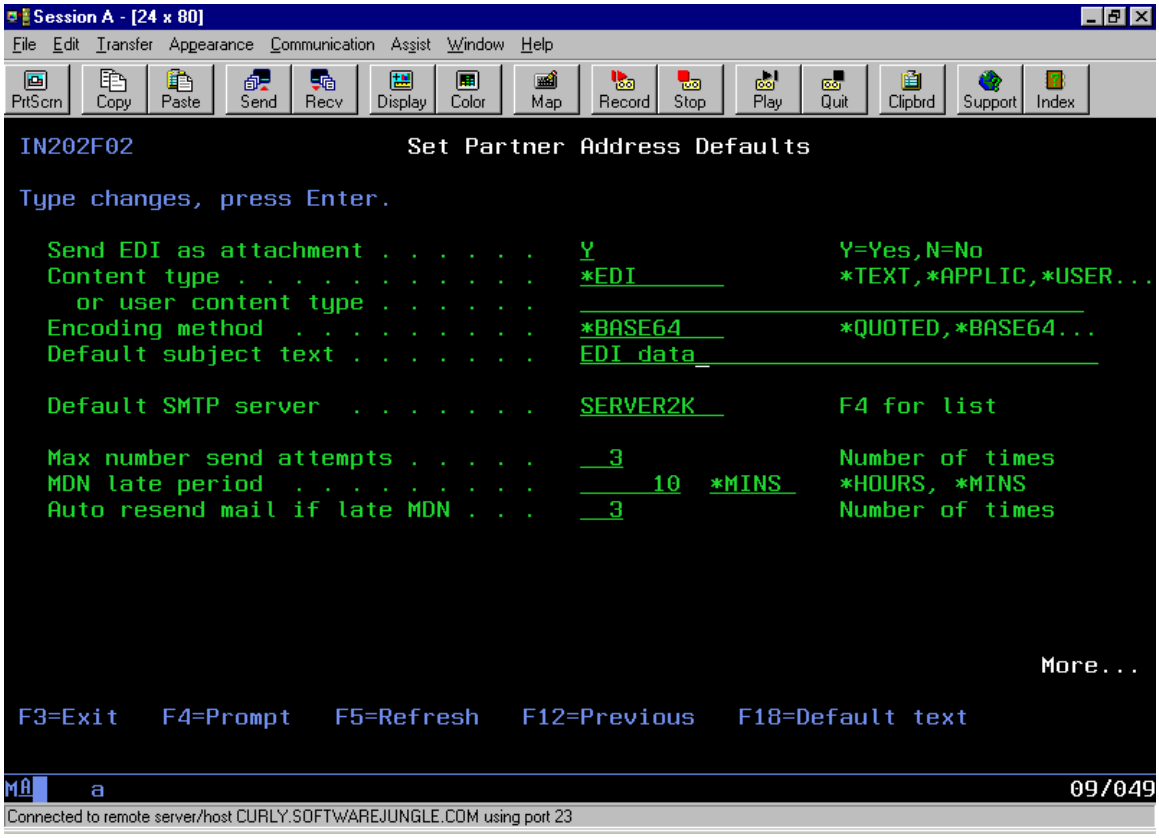
As well as using the menu option you can use the command ZLODDCERT to load a certificate. You should see the panel shown below.



The command will require the file name that you saved the certificate under and the path name. You may enter either the file name that includes the full path or the file name and path separately. Press <*enter*> to load the certificate. Once you've loaded the certificate you should be able to view it from the "*Work With Certificates*" option. You can associate this certificate with a trading partner profile either from this option or from the Trading Partner configuration option.

5. Set Trading Partner Default

There are certain trading partner profile settings for which system default options can be set. If no specific value is given in the profile then these values will be used.



Send EDI as attachment: For S/MIME applications leave this as 'N'. For non-S/MIME applications this flag controls whether the data is sent as an attachment to the mail item or in-line.

Content-type: This is the default content type on the message used primarily for the DSTEDIDTA command. For most S/MIME applications it should be set to *EDI. Valid values are:

- *TEXT – text/plain
- *APPLIC – application/octet stream
- *EDI – one of either application/edi-x12, application/edi-edifact or application/edi-consent depending on the EDI standard used.
- *MSWORD – Microsoft word document
- *USER – enter a value under the user content type.

Encoding method: If you intend to encrypt and/or compress the data you're sending or if you're using a transport mechanism other than SMTP (AS1) then you can set this value to *BINARY. This parameter controls the data encoding method used on the actual data to be sent. Typically encoding is used where the data is to be sent over a transport that does not guarantee a clear 8 bit channel. SMTP falls into this category. Valid options are *QUOTED, *BASE64, *BINARY, *7BIT, *8BIT.

Default subject text: This is the subject line that will be included on outbound messages.

Default SMTP server: For any SMTP (AS1) requests where an SMTP server isn't specifically set for a trading partner, this is the server that will be used. You can use *F4* to select one of the existing server configurations.

Maximum number of send attempts: This is the number of attempts that are made to successfully deliver a mail item to the immediate destination site. For AS2 and AS3 this means delivery to the URL specified on the trading partner definition. For AS1 it means delivery to the relevant SMTP server. If an unsuccessful attempt is made to deliver a mail item then it will be re-tried the next time a send command is run until this number of attempts has been exceeded, whereupon the mail item will then be marked in error.

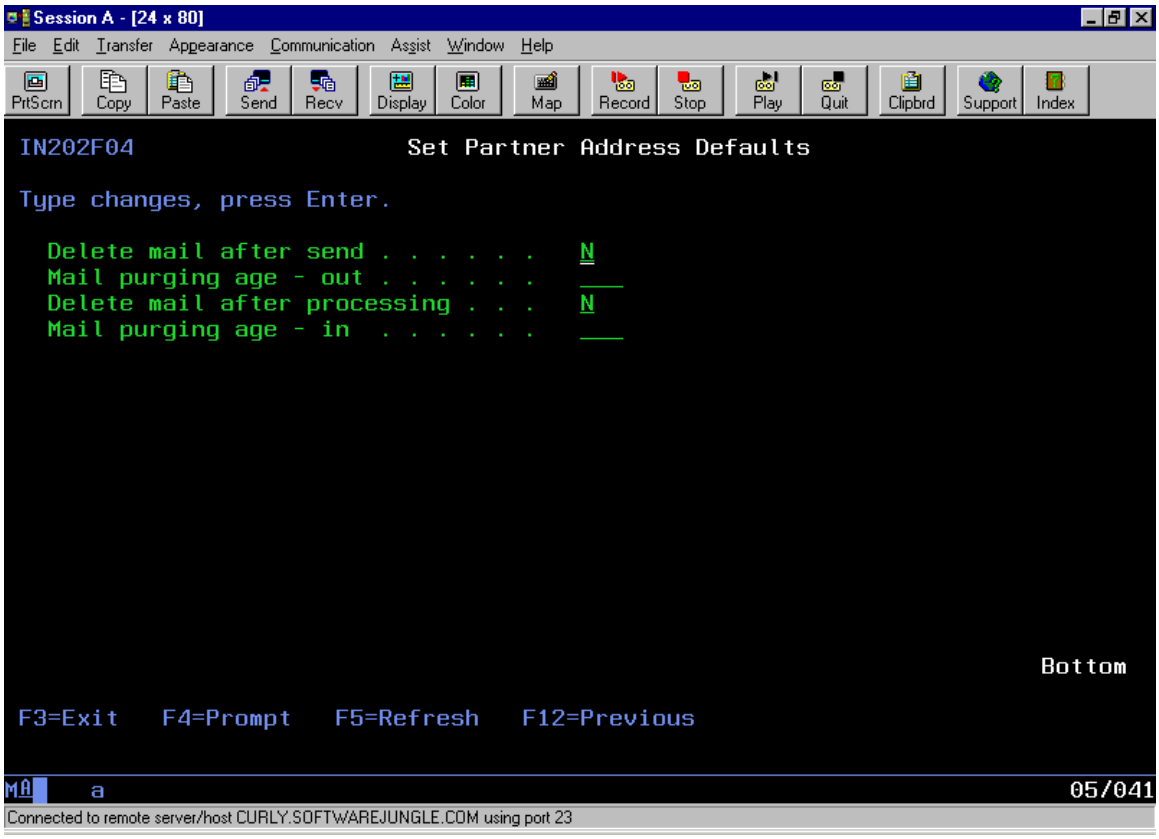
MDN is late period: Use this value to provide a time period after which EDI/IP will consider any MDN not yet received to be late. Some transport mechanisms (AS2) have the option of requesting that the MDN be returned immediately in which case this value would be moot. Enter either *HOURS or *MINS to provide a measure for the "MDN is late period" parameter above.

Auto re-send mail if late MDN: This action is only triggered if the ZCHKACKSTS (check acknowledgement status) command encounters a sent message for which an MDN has not been received within the specified period. If a message is found with a late MDN then that message will be automatically released for re-send up to the number of times given here after which a permanent "MDN late" error will be signaled.

Press rollup for more options

Delete mail after send: You have two options when it comes to purging mail that has been sent. This one controls whether the actual mail data is automatically deleted after the message has been sent. The mail header record will remain in the log. Setting this value to 'Y' means that the mail data will be permanently removed as soon as the message has been sent. You will not be able to re-send this message unless you re-create it from the original EDI/file data. The alternative is to use the **PRGSMTPDTA** command to schedule purge requests at appropriate intervals.

Mail purging age (out): An alternative to immediately deleting sent data is to use the PRGSMTPDTA command. This parameter allows you to set an aging period, expressed in days, after which SENT mail is eligible for purging by this command. This value can be overridden on the PRGSMTPDTA command.

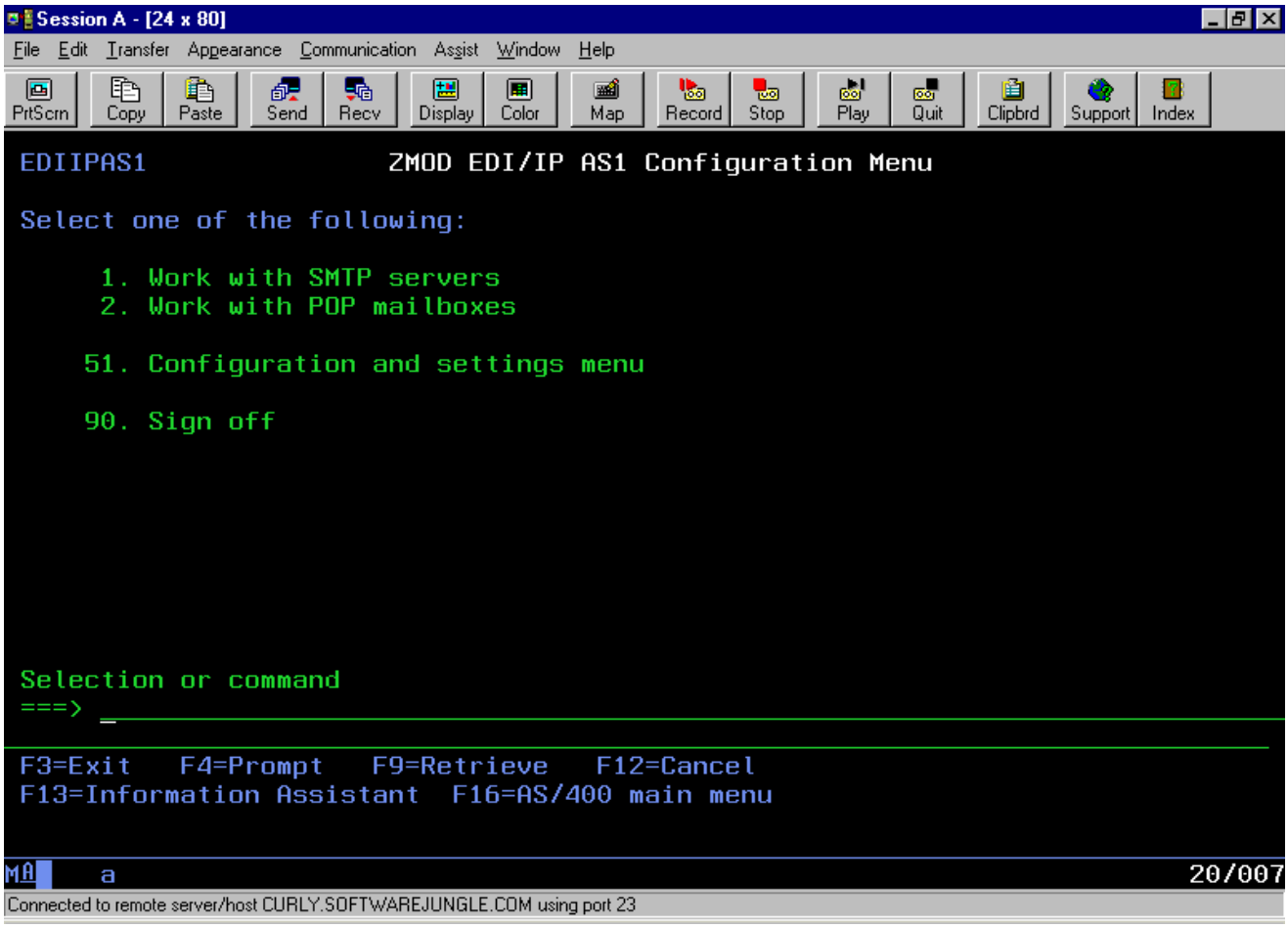


Delete mail after processing: See the "delete mail after sending" parameter above. This value provides the same function for received-and-processed mail.

Mail purging age (in): An alternative to immediately deleting received and processed data is to use the PRGSMTPDTA command. This parameter allows you to set an aging period, expressed in days, after which received and processed mail is eligible for purging by this command. This value can be overridden on the PRGSMTPDTA command.

The AS1 Menu.

The EDI/IP AS1 menu will allow you to do the static setup that relates to sending messages using SMTP. It can be reached by using the command **GO EDIIPAS1**.

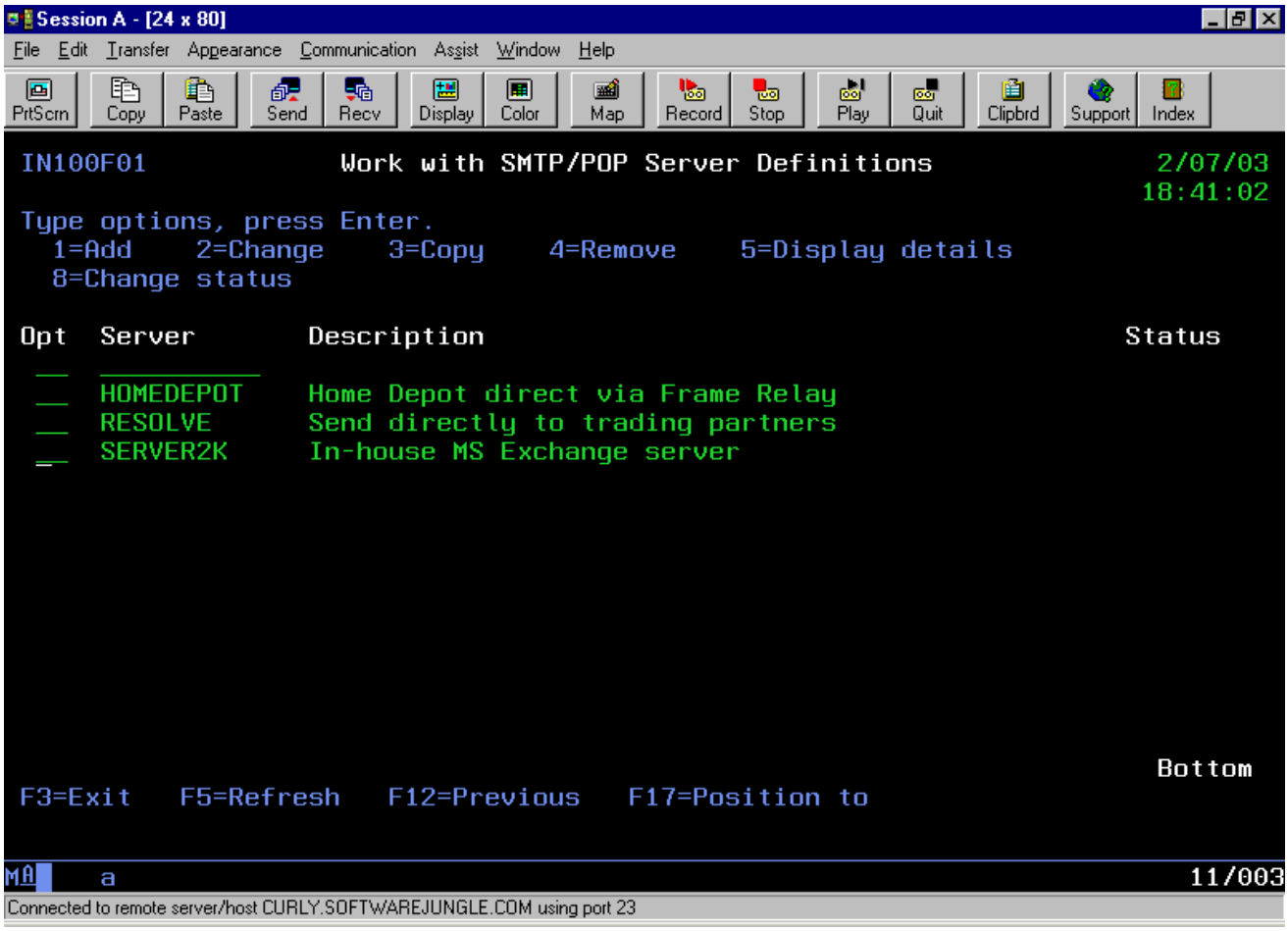


Options available from this menu are:

- *Work with SMTP servers.* To send messages via SMTP you'll need to configure at least one SMTP server.
- *Work with POP mailboxes.* Unless you're running the AS1 server, received email messages will be stored in one or more POP mailboxes on the server(s) you defined above. This option lets you set up the POP mailboxes that you'll use.

1. Work with SMTP Servers

This option is where you define the SMTP servers that will be used to send messages. You have a number of options when considering how you'd like to architect your AS1 system. AS1 messages created by EDI/IP on the iSeries need to be delivered to an SMTP server. That server can be an interim mail server that receives mail and then forwards it to the correct destination or it can be the final server that handles mail for a particular email address. You can either have the EDI/IP SMTP client do the email address resolution and deliver each message directly to the trading partner's mail server or you can deliver all these messages to your own mail server and have that server forward the mail. In order for the first option to work, the iSeries machine will need to be able to reach the Internet (if your trading partner(s) have mail servers accessible through a private network then this, of course, isn't necessary). For the second option to work your iSeries machine will need to be able to connect to your in-house mail server and the mail server will have to accept mail from the iSeries for SMTP forwarding. For mail for different partners a combination of options can be setup too.



The options available from this screen are:

1=Add. Create a new SMTP server.

2=Change. Make changes to the SMTP server definition.

3=Copy. Copy an existing server definition.

4=Delete. Delete a server definition.

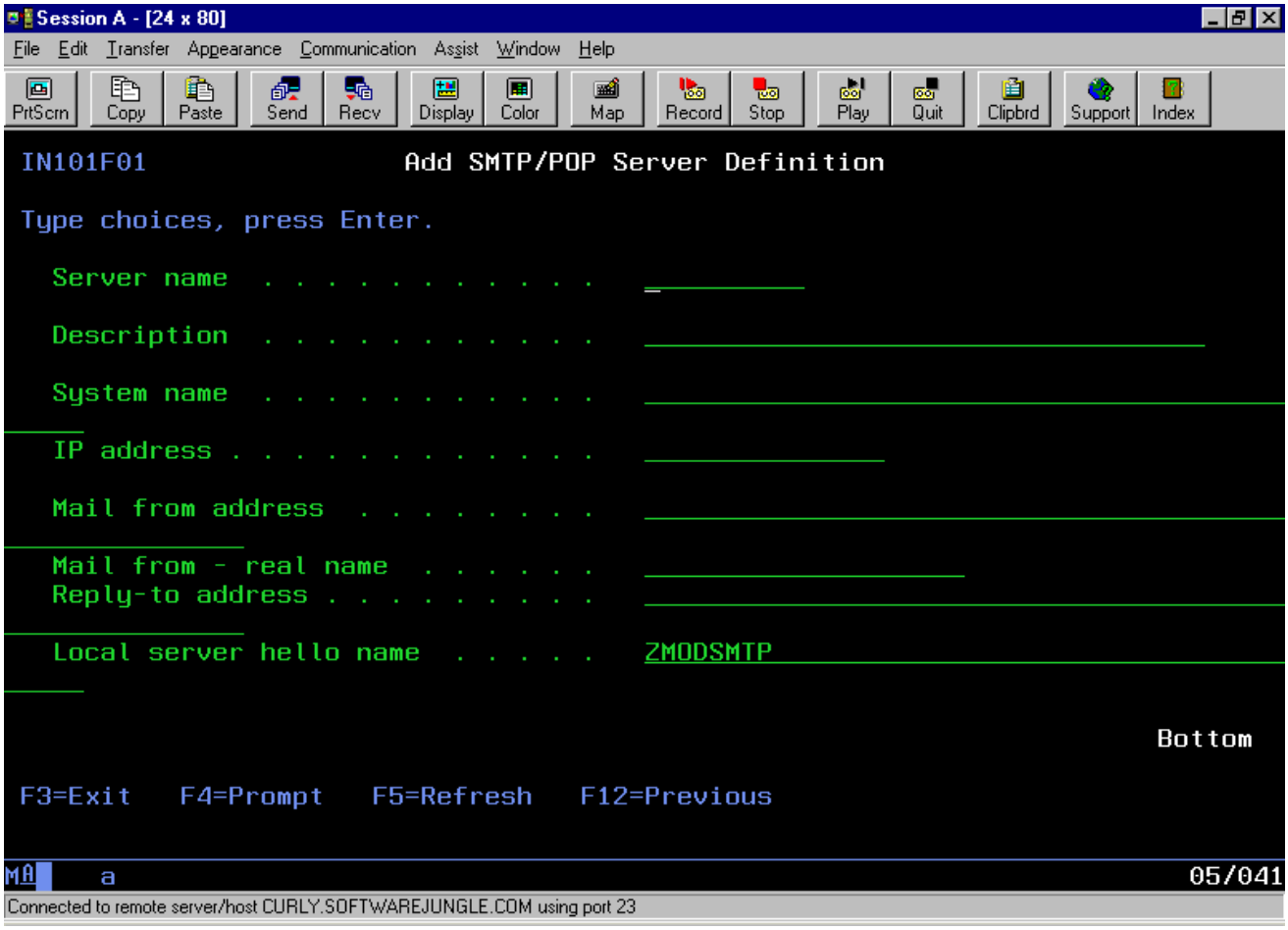
5=Display. Display the SMTP server configuration.

8=Change status. Toggle the server status between *"released"* and *"held"*. Mail for a held server will not be delivered till the server is released.

To create a new SMTP server put a '1' in the top option column and press <enter>.

Creating a New SMTP Server Definition

If you're going to use your in-house mail server as the SMTP server for AS1 messages then you'll need to know either the host name or IP address of the mail server before you can create the definition. If you intend to avoid your local mail server and send AS1 messages directly to your trading partner SMTP servers then you'll need to create an SMTP server definition using the special value of *RESOLVE in the "system name" field *(see below).*



Server name: This is an internal name that you want to assign this server. Any name will do but it should start with an alphabetic character and shouldn't contain embedded blanks.

Description: Any text that further describes this server or its' usage.

System name: This is the host name of the mail server. This host name must be resolvable to an IP address from this machine. Either the system name *OR* IP address field should be supplied. If both are given, the IP address value will be used. If you want EDI/IP to do the mail name resolution and deliver mail directly to the trading partners' SMTP servers then use the special value of ***RESOLVE** here.

IP address: This is the IP address of the SMTP server being described. Use either the system name *OR* IP address fields. If both are given, the IP address value will be used.

Mail From address: You need to provide an email address that will be used as your AS1 email identifier on outgoing messages (i.e. the "From" value). In most cases this is the email address that your trading partners will be sending AS1 messages to.

Mail From - real name: This optional value is an identifying name that is supplied in the outgoing email along with the email address.

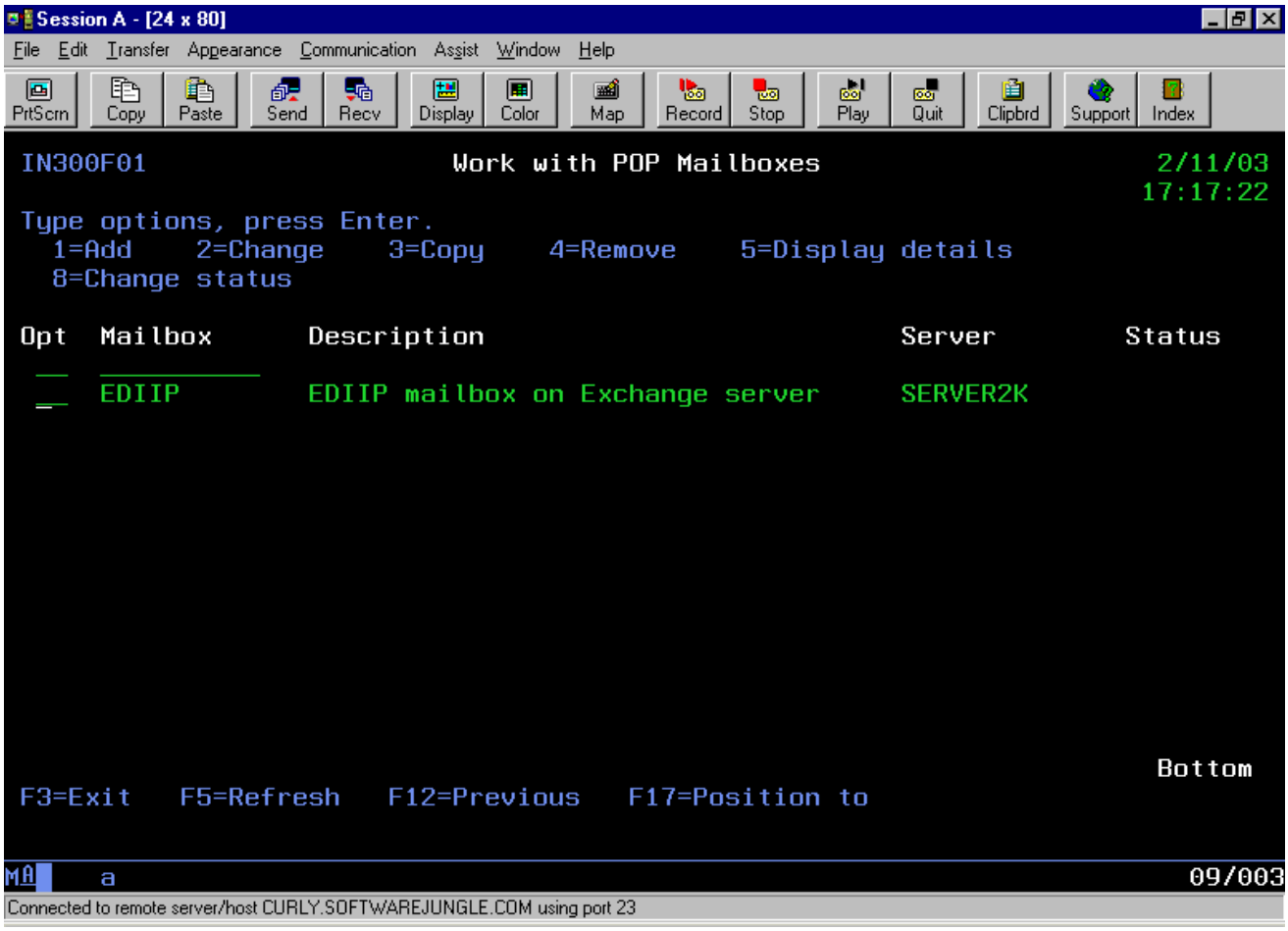
Reply-to address: This optional value allows you to provide an email address that will be used as the reply-to address for all outgoing mail to this server. If supplied this is usually the email address of the administrator responsible for this system.

Local server hello name: This is the name that the EDI/IP SMTP client will use to identify itself to the SMTP server when attempting to forward messages. This value is then recorded in the mail history. Unless you need to use some specific text leave the default value.

This completes the setup of an SMTP server. If this is the SMTP server definition you intend to use for most of your trading partners then you should define it as the default SMTP server under the "Set trading partner defaults" option. To send AS1 mails that you've created to this server use the SNDMSGSMTP command.

2. Work with POP Mailboxes

A POP Mailbox is a mailbox on a POP server where inbound email for a certain email address is stored until it's picked up. In an EDI/IP environment where you're using AS1 as the transport mechanism you'll probably have at least one email account whose address you'll give to your trading partners. These trading partners will send their secure mail to that address. On your local mail server you'll establish this account as a POP account. As such, when mail arrives for this address it will be stored until a POP client logs in to pickup that mail. What you're going to define here is the information about the POP mailboxes you've set up and how they'll be accessed.



The options available from this screen are:

1=Add. Create a new POP mailbox.

2=Change. Make changes to the POP mailbox definition.

3=Copy. Copy an existing POP mailbox definition.

4=Delete. Delete a POP mailbox definition.

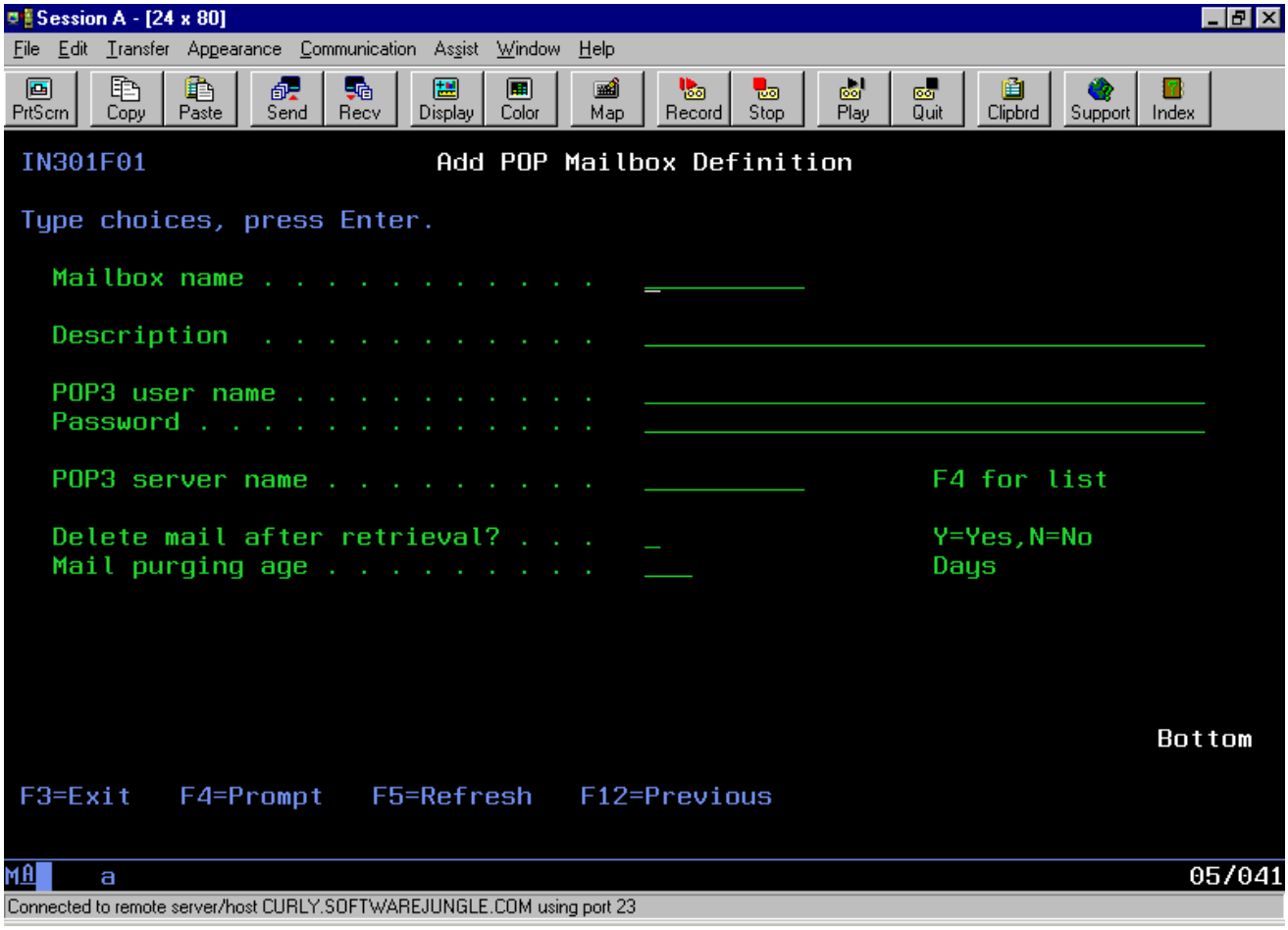
5=Display. Display the POP mailbox definition..

8=Change status. Toggle the mailbox status between *"released"* and *"held"*. POP mailboxes that are held will not be polled for mail until they're released.

To create a new POP mailbox put a '1' in the top option column and press <enter>.

Creating a New POP Mailbox Definition

Before you can define a POP mailbox you must have created a definition for the SMTP server that the POP mailbox resides on.



Mailbox name: This is an internal name that you want to use to refer to this POP mailbox by. Any valid name will do but it must begin with an alpha character and cannot contain embedded blanks.

Description: Any text that defines the use for this mailbox.

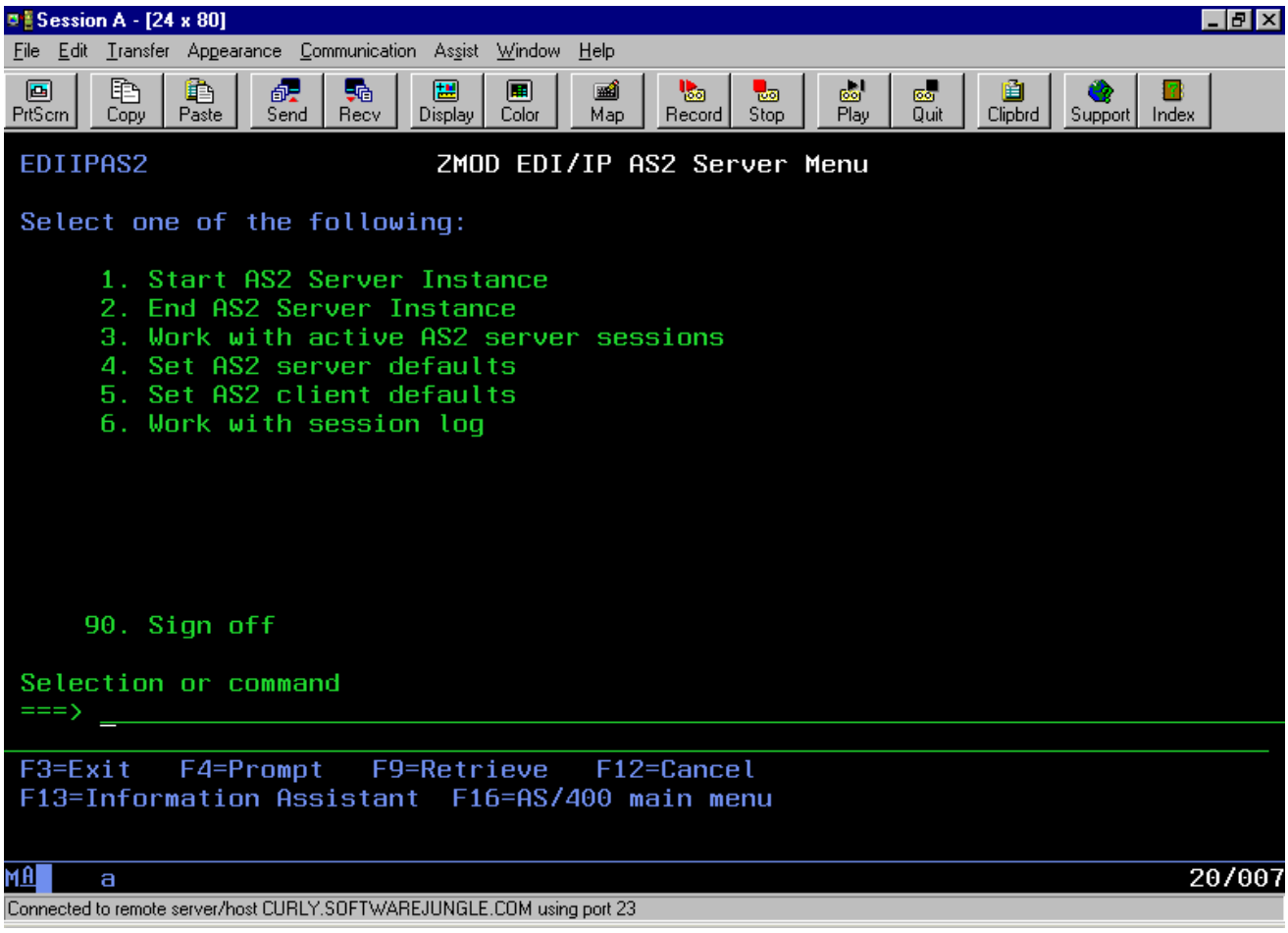
POP3 User Name: Every POP mailbox that you define on your mail server will have a user name. Usually this is the name of the user profile associated with this account.

Password: The POP mailbox will be secured by a password. This could be the password of the user profile linked to this account. The correct POP user name and password will be required in order to log into his mailbox and retrieve mail

This completes the setup of a POP mailbox. To retrieve mail from this and other POP mailboxes use the command RTVPOPMSG.

The AS2 Menu

The EDI/IP AS2 menu will allow you to do the static setup that relates to sending messages using the AS2 variant of HTTP. It also has the options required to start, end and show AS2 servers. It can be reached by using the command **GO EDIIPAS2**. An AS2 server must be started before your trading partners can post AS2 messages to you although an active server isn't required for you to be able to send AS2 messages. You may have more than server active at any one time, perhaps each with different configurations (e.g. one server running HTTP and another using SSL). Before attempting to start an AS2 server or client request for the first time you should review the AS2 server and AS2 client default settings.

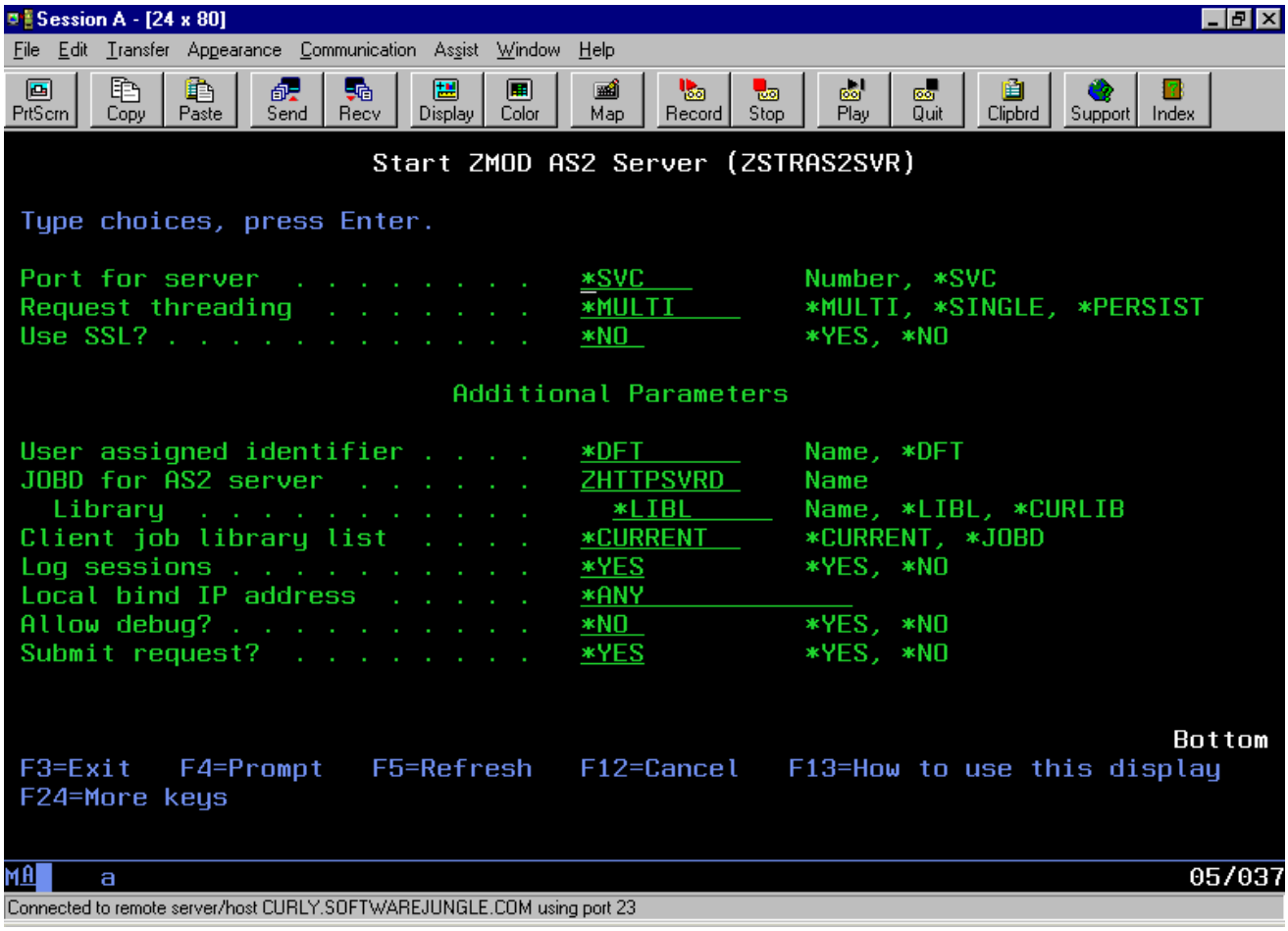


Options available from this menu are:

- *Start AS2 Server Instance.* This option uses the command ZSTRAS2SVR to start an AS2 server. The AS2 server will start listening on the specified port for requests from trading partners.
- *End AS2 Server Instance.* Use this option to stop an active AS2 server. It invokes the command ZENDAS2SVR.
- *Work with active AS2 server sessions.* Take this option to see a list of the currently active AS2 servers and any active sessions.
- *Set AS2 server defaults.* Here's where you can set the parameters for the AS2 servers including any SSL setup details.
- *Set AS2 client defaults.* Parameters relevant to outbound AS2 send functions are defined here.
- *Work with session log.* All inbound AS2 requests from trading partners are logged as sessions. From this log you can see who connected, when and what the status of that session was.

1. Start an AS2 server instance

In order to be able to receive and process secure messages using AS2 you must have a server running. A single server may process multiple incoming requests at a time up to the maximum number of concurrently active sessions that you configure. You will provide your trading partner with a URL that will allow them to connect to this server. This URL will include the TCP/IP port number that this server is listening on for incoming connections. The normal way to end this server is by using the command ZENDAS2SVR. If this server is ended by any other means, either by the job being canceled, TCP/IP being ended or a program abort, a message will be issued to the broadcast system alerting you that the server is down. You may have more than one AS2 active at a time subject to the restriction that only one server may be active on a given port for a particular host (or specific IP address if the host is multi-homed).



Port for server: Enter the TCP/IP port number that this server will listen on. If any other server (AS2 or not) is currently active on this port for this host then this AS2 server job will not start and a message will be issued to the broadcast system. You can use the command **NETSTAT** to see what servers are active on which ports. The special value of *SVC can be used to have the TCP/IP services table used to look up the port number associated with the EDI/IP service. The EDI/IP AS2 service is called "zmodhttp" and on install is set with a port value of 10180.

Request threading: The choices here are *MULTI, *SINGLE and *PERSIST. This parameter controls how incoming requests are treated. If *MULTI is used then every incoming AS2 request will spawn a separate process to deal with that request so long as the number of concurrently active requests has not been exceeded. Connections that exceed these limits will be refused. Once the AS2 session is complete the process will end. This is the most common setting. *PERSIST allows you to pre-start a number of AS2 service jobs that will stay active waiting for incoming requests. The incoming requests will be assigned to the service jobs that are free. If the number of incoming requests exceeds the available number of service jobs then those requests will be queued. If you expect to receive large numbers of AS2 requests in a short time frame and those AS2 sessions are likely to be short then this option provides better performance.

Use SSL: If you want this AS2 server to operate as an SSL server then answer *YES here. You must have allowed SSL on the AS2 server-defaults option and defined an SSL certificate path to use.

Use <F10> to get more options

User assigned identifier: To assign a name to the AS2 server so that you can better identify it then enter a value here. If you leave the default of *DFT then the name used will consist of the text AS2_ (or AS2S_ if SSL) combined with the port number.

JOBD for AS2 server: If the *"submit request"* parameter is set to *YES, the default, then this AS2 server instance job will be submitted. This is the job description that will be used to determine the job parameters for this AS2 server job. The default job description provided is ZHTTPSVRD.

Client job library list: This value controls whether the library list used for the AS2 server jobs is taken from the job description or is the same as current job's library list. Valid values are *CURRENT and *JOBD.

Log sessions: You have the option of logging every incoming AS2 request as a truExchange session. This job description will be used to determine the job parameters for this AS2 server job.

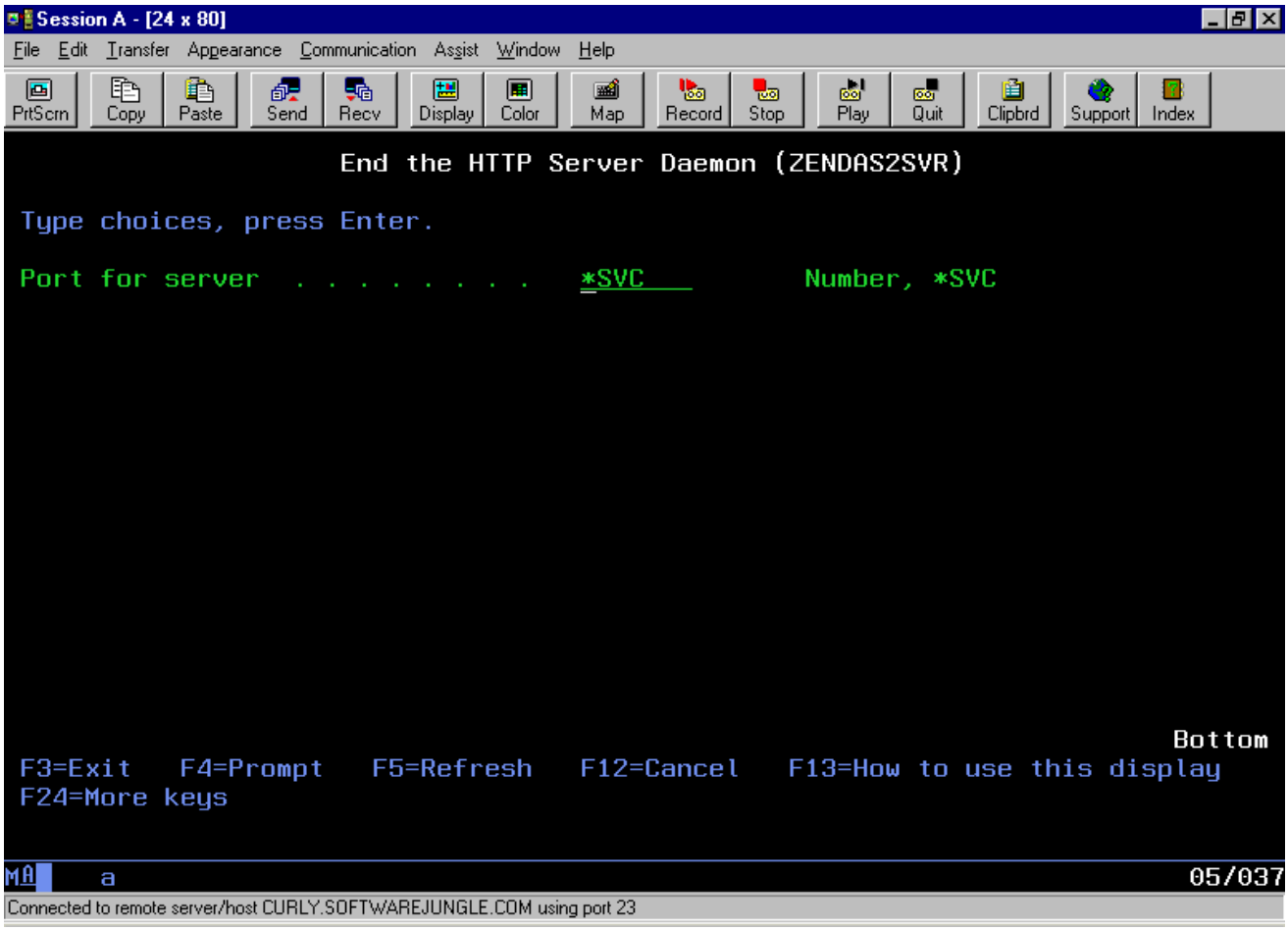
Local bind IP address: This parameter only has relevance if your AS/400 is multi-homed (i.e. has more than one IP address). When you submit an AS2 server it will listen on the specified port for all IP addresses associated with this machine. If you want to restrict this server to listening for requests on just one of the IP addresses defined for this host then give that IP address here.

Allow debug: This parameter should only be used at the request of your EDI/IP support team. It allows for very specific debugging of issues.

Submit request: This value controls whether or not the AS2 server job is submitted from the current job or whether it runs in the current job. Valid values are "*YES and *NO. Usually this would be *YES.

2. End an AS2 server instance

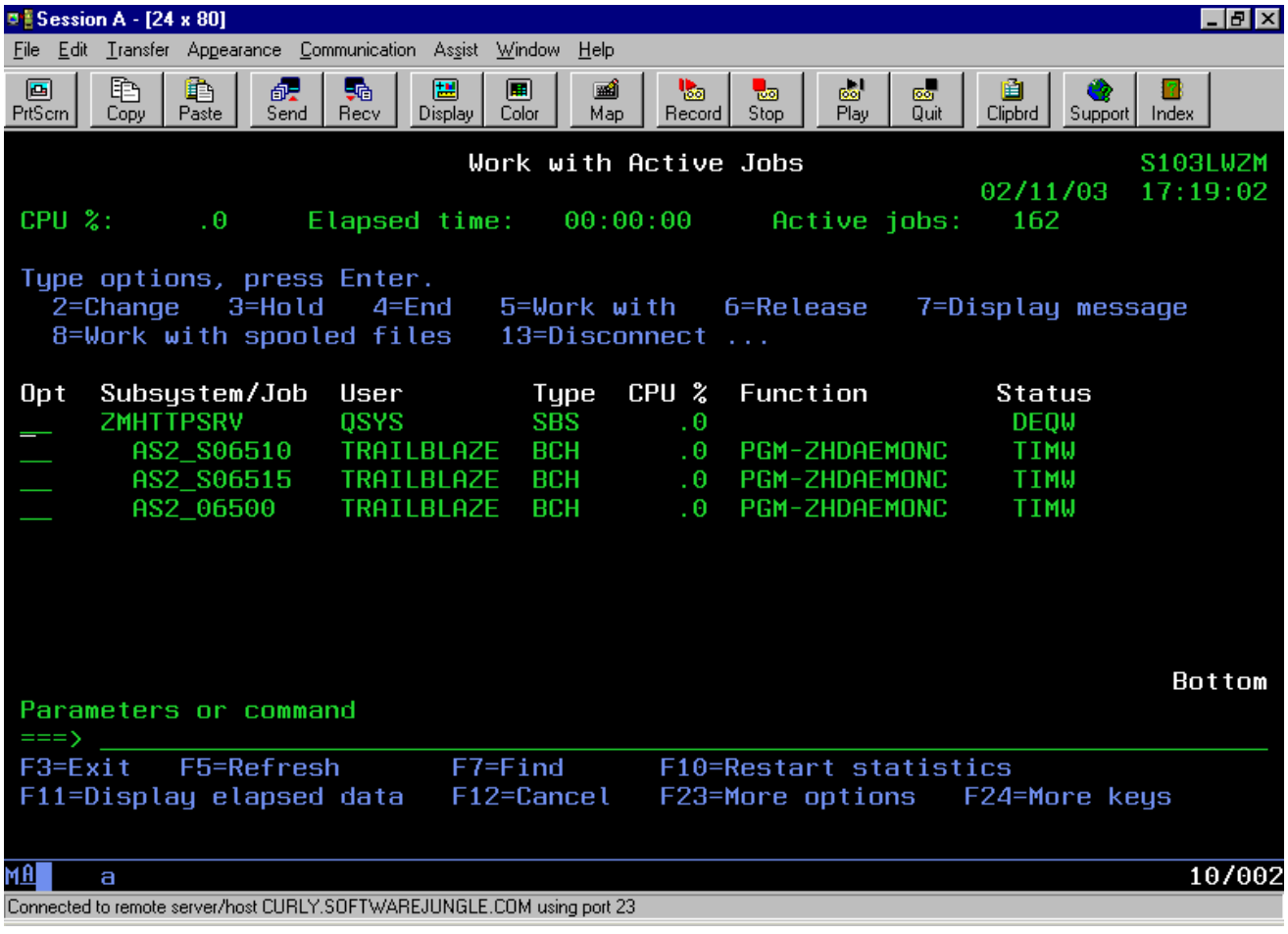
To end an AS2 server that's currently listening on a port, choose this option from the menu or use the command ZENDAS2SVR. When you end a server, any remote sessions that are active with that server will be allowed to complete. No new connections will be accepted. Ending the AS2 servers in the ZMHTTPSRV subsystem will not end the subsystem - if you need to end the ZMHTTPSRV subsystem you'll have to use the ENDSBS command. If you're trying to include a shutdown of the AS2 servers in a CL process then you can use the ZENDAS2SVR command to stop the active servers. Once any currently active connections from trading partners have completed you can end the subsystem. To programmatically check for active jobs in a subsystem you can use the EDI/IP utility command **ZCHKSBSJOB** that can be used in your CL programs.



Port for server: Enter the port number that the server you wish to end is listening on. If you started the server using the special value of *SVC then the port number specified in the TCP/IP service table for the service "zmodhttp" will be used.

3. Work with active AS2 server jobs

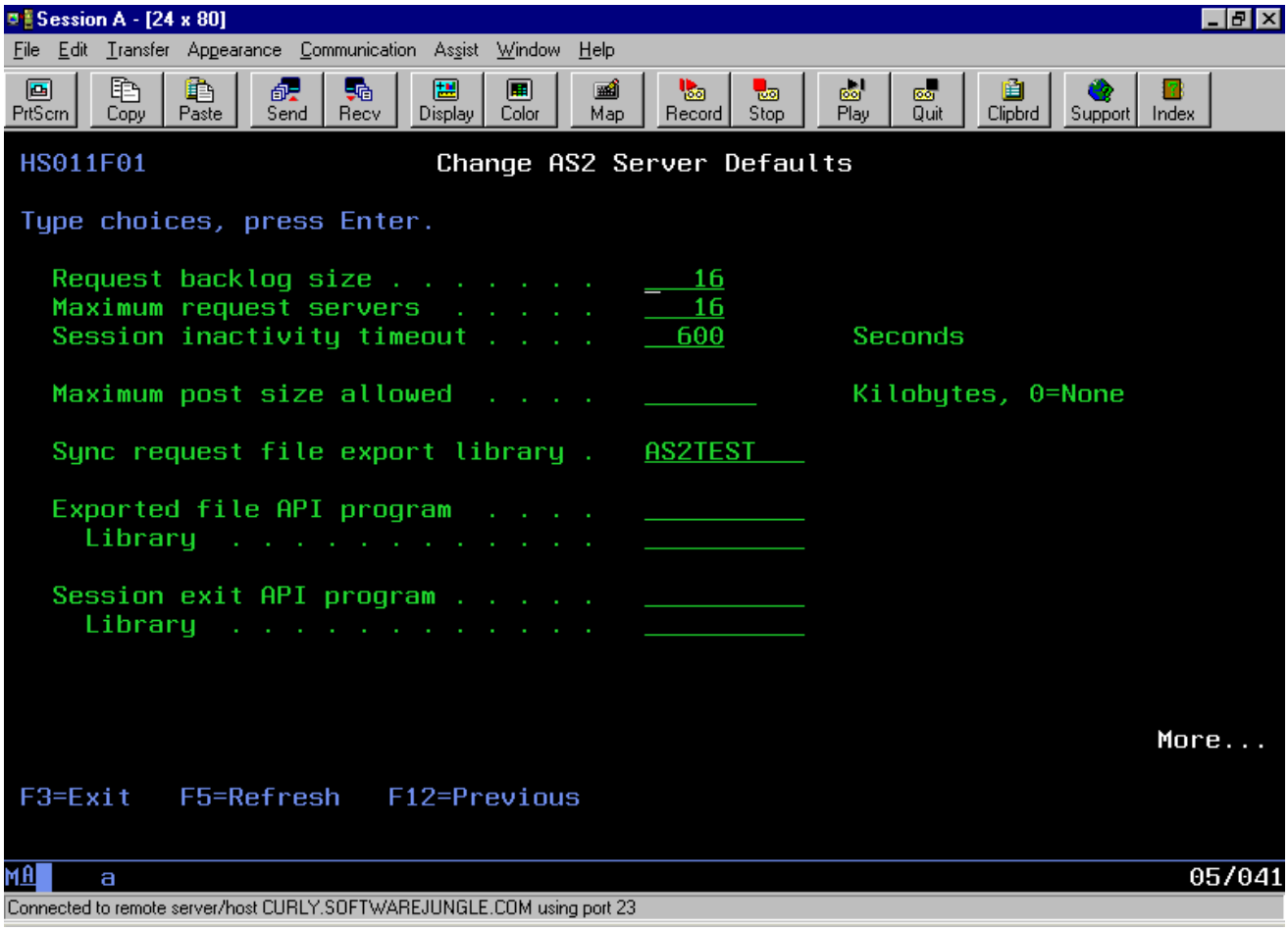
When AS2 server jobs are started they will run in the subsystem determined by the job queue in the job description used on the ZSTRAS2SVR command. The default job description provided (ZHTTPSVRD) will cause the AS2 server jobs to go active in the subsystem ZMHTTPSRV. This subsystem will be started when the ZSTRAS2SVR command is issued if it's not already active. Option 3 from the AS2 menu gives a display of the AS2 server jobs currently active in the subsystem. The job name will be the user identifier value given on the start command which by default is the text "AS2_" (or "AS2_S if this is an SSL server) with the listening port number attached.



The above example shows three active AS2 servers listening on ports 6510, 6515 and 6500 respectively. The servers listening on ports 6510 and 6515 are SSL servers. If the *PERSIST option was chosen for the threading parameter on the server start command then a number of sub-processes will be shown under the server job. The "Maximum request servers" parameter on the AS2 server default settings controls the number of persistent connections that are available. If the default value of *MULTI was chosen for the threading parameter then the sub-processes will be seen only when a remote system has made a connection.

4. Set AS2 Server Defaults

The parameters that control the operation of the AS2 servers are set using option 4 from the AS2 menu. Changes made to these default settings are not effective for currently active AS2 servers until the server is stopped and restarted.



Request backlog size: This value controls the number of incoming requests that are backlogged till a process becomes free. If the number of backlogged requests exceeds this value then new connections are refused. If the threading method is *MULTI then new processes are started for each incoming request so the time to wait for a process is negligible. However if connections are received at the same instant then some backlog will occur while the first request is started. For the *PERSIST option backlog will occur if all the pre-start processes are active with other requests. The default value is 16.

Maximum request servers: The maximum number of request servers is the maximum number of concurrently active AS2 sessions. The default value used here is 16. This number should be changed to suit your expected AS2 connection volume. If the threading method used on the AS2 server start command is *PERSIST then this will be the number of processes that will pre-start to deal with expected incoming requests. If the number of incoming requests exceeds the number of free processes available then the new requests will be backlogged (see above) till a process becomes free. If the threading method is *MULTI then new processes will start for each incoming connection request so long as this number of processes isn't exceeded. If this value is exceeded then new connections will be refused.

Session inactivity time-out: This is the amount of time in seconds that the server will wait for data from the remote trading partner connection before declaring a timeout and abandoning the session. The default is set to 10 minutes (600 seconds).

Maximum post size allowed: The maximum post size value allows you to control the size of files received from your trading partners and so prevent remote sites from sending you excessively large amounts of data. If no number is given then there's no limit set and a trading partner can send you files of any size. If there's a value given then it's expressed in kilobytes.

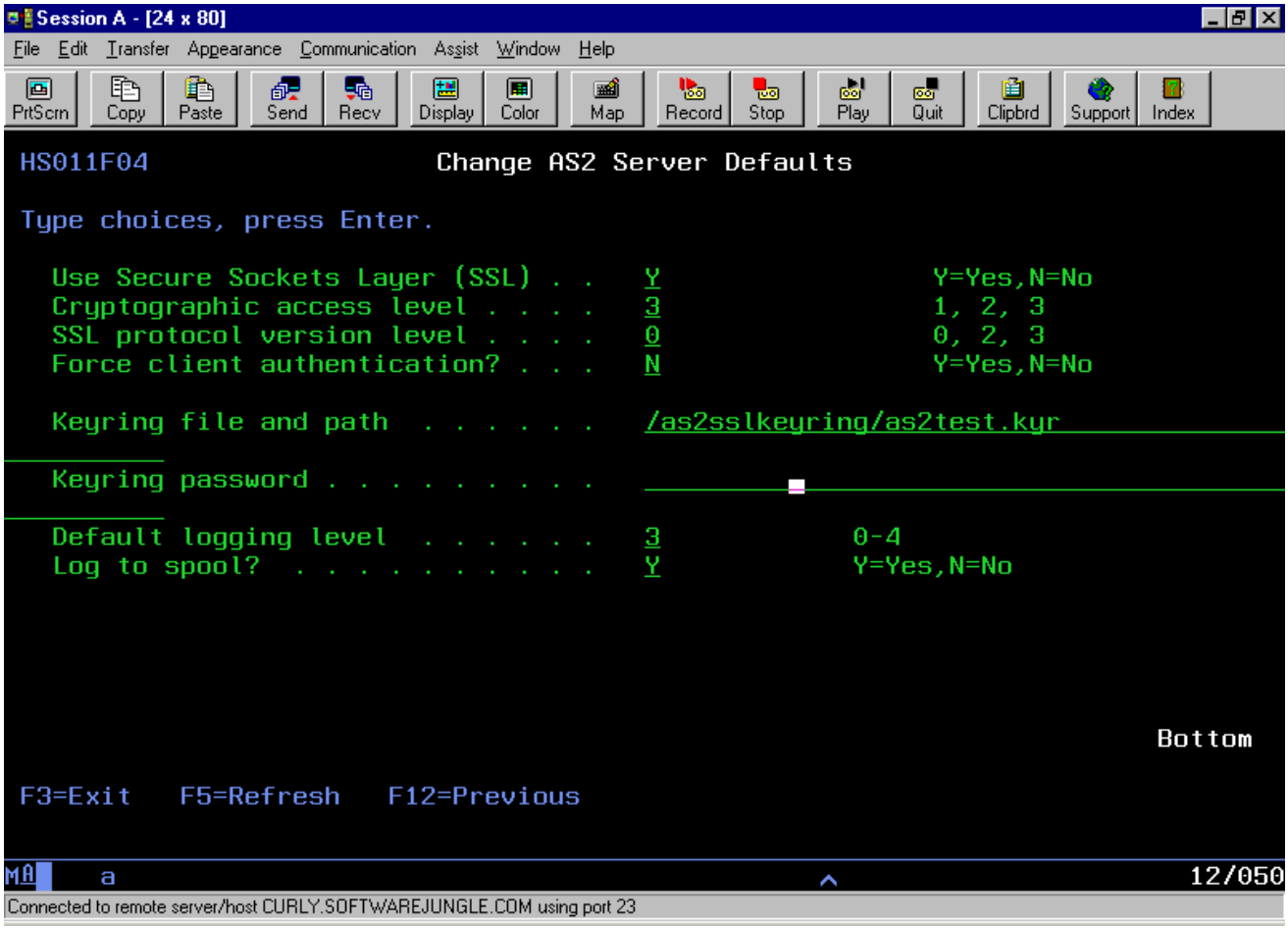
Sync request file export library: Trading partners that send data to you via AS2 have the option of requesting a synchronous MDN. A synchronous MDN is an immediate confirmation of delivery that requires that the message be decrypted, decompressed and verified if needed. In these cases EDI/IP will write the decrypted data files immediately to a library of your choosing. That's the library named here. The files will be created with a naming scheme that consists of the letter Z followed by a 9 digit sequence number (e.g. Z00000001). The *"exported file API"* described below can be used to notify you that data has been exported or you could use the **SCNPRCDBF** or **IMPEDIDBF** commands to process these files by schedule.

Exported File API program: If a received file is immediately exported in the manner described in the *"Sync request file export library"* parameter above then there's an optional API that can be called that will be passed details about the file and trading partner. This API could then, if needed, take care of processing this data. The API program is a CL or RPG program written by you (perhaps using the example code as a guide) for this purpose. The example CL code for this API can be found in member A2FILEAPI in the source file ZMODEDIIP/QCLAPIS.

Session exit API program: This is not currently used.

Press <Rollup> for further values.

Use Secure Sockets Layer (SSL): If you intend to start any AS2 server as an SSL server then you must enter 'Y' here and configure the other SSL related parameters. Putting a 'Y' here does not automatically start any AS2 server as an SSL server (that's controlled on the ZSTRAS2SVR command) but simply allows SSL as an option for this command.



The following parameters are relevant only if you're using SSL.

Cryptographic access level: This number corresponds to the access level of the IBM cryptographic software loaded on your iSeries. If you have AC3 loaded then use 3.

SSL version level: This controls the SSL version level you want to operate on. The numbers '2' and '3' correspond to SSL versions 2 and 3 respectively. Using a '0' here allows the system to automatically switch between versions 2 and 3 depending on the capability of the remote client.

Force client authentication: This option is rarely used for AS2 however that may change in the future. Setting this value to 'Y' forces the client to authenticate itself to you and requires that you pre-load their certificate into DCM and mark it as trusted.

Keyring File and Path: To do SSL you must first have set up a key and obtained a certificate using DCM. In this process you will have created a keyring file in a folder somewhere on your iSeries machine. You now need to provide the name of the keyring file (with path) for SSL to use for your AS2 server.

Keyring password: Leave this value blank.

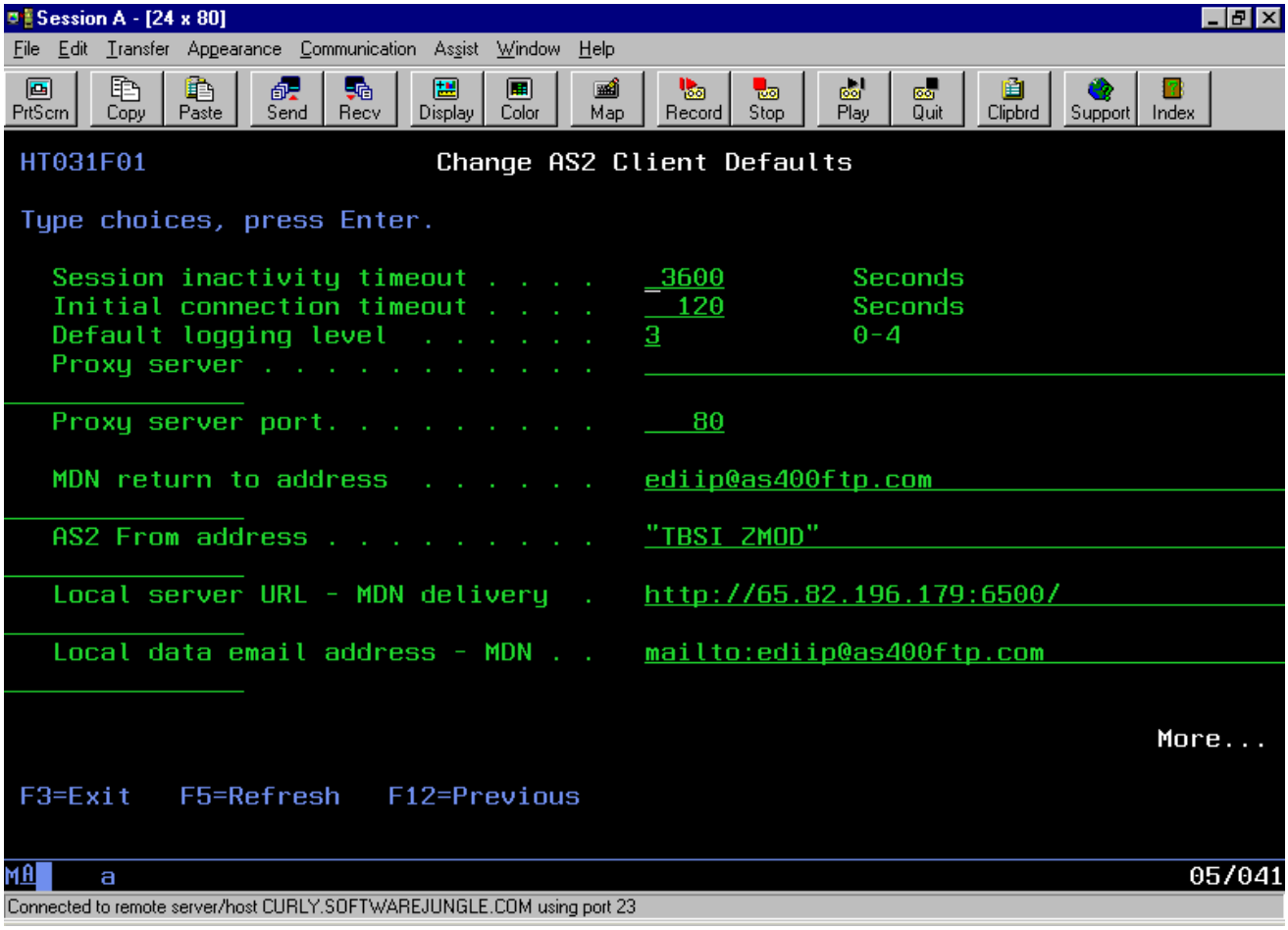
Default logging level: Your AS2 server will log messages to the ZMOD log. The type and number of messages logged is controlled by this value. Enter a value of 1-4 that specifies from least to most the level of the logging you want to do.

Log to spool: No longer used.

This completes the setup of the AS2 server defaults.

5. Set AS2 Client Defaults

The parameters that control the operation of the AS2 client are set using option 5 from the AS2 menu.



Session inactivity time-out: In a normal session the AS2 client will connect, send data to the remote AS2 server and then wait for a response. This time-out value controls the length of time that the client will wait for a response from the remote server before declaring a time-out and ending the session abnormally. It's given in seconds.

Initial connection time-out: This time-out value controls the length of time the client will spend attempting to establish connection to the remote host. If the remote host is not found with this time the client will abandon the session. The time-out is expressed in seconds.

Default logging level: Your AS2 client will log messages to the ZMOD log. The type and number of messages logged is controlled by this value. Enter a value of 1-4 that specifies from least to most the level of the logging you want to do.

Proxy server: Not used at this time.

Proxy server port: Not used at this time.

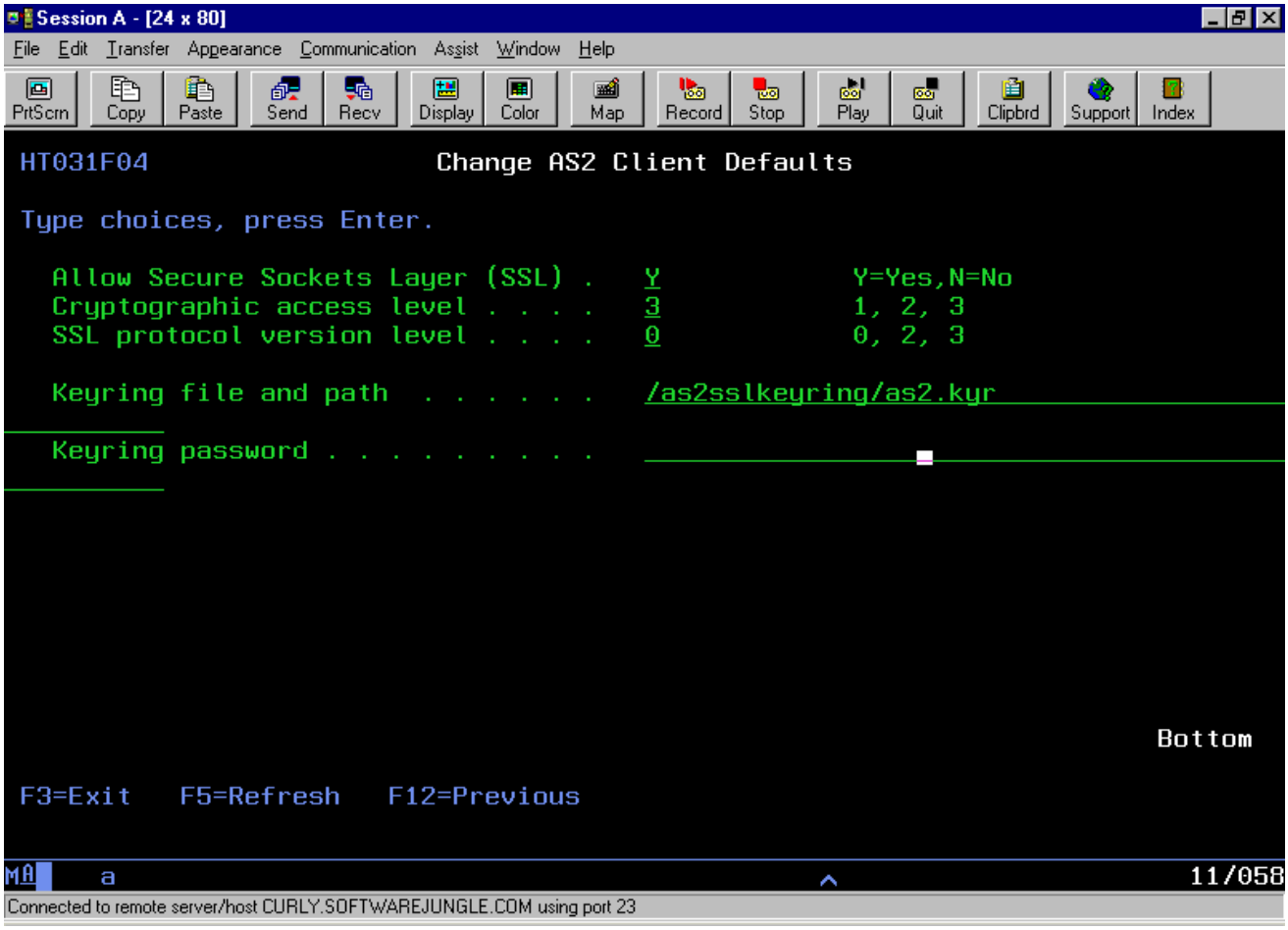
MDN return to address: This value is used for the "Disposition-notification-to" line in messages where an MDN is requested. For AS1 this value is set to the email address that the MDN is to be delivered to. For AS2 it currently doesn't really have a purpose but it's included here in case it does in the future. Since you still have to provide a value you should use either the "AS2 From Address" value (same as below) or the URL for the AS2 server.

AS2-From Address: Both parties in an AS2 relationship will have an AS2 identifier. This is your AS2 id. Unless this is overridden by a value specific to the trading partner relationship then this id will be used in all outgoing AS2 messages.

Local server URL - MDN delivery: This parameter is used only if you're requesting asynchronous MDNs from your trading partners. Asynchronous MDNs are sent by the trading partner directly to your AS2 server. This value therefore should be the URL needed to connect to your AS2 server.

Local data email address - MDN: This parameter is used only if you're requesting asynchronous MDNs via email from your trading partners. Asynchronous email MDNs are sent by the trading partner to you via AS1. This value therefore should be the email address used by your trading partner to send you AS1 messages.

Press <Rollup> for further values.



Use Secure Sockets Layer (SSL): If you need to connect to any AS2 trading partners that use an SSL URL (i.e. one that begins HTTPS:// rather than HTTP://) you will need to enter 'Y' here and configure the other SSL related parameters.

The following parameters are relevant only if you've set the above parameter value to 'Y'.

Cryptographic access level: This number corresponds to the access level of the IBM cryptographic software loaded on your iSeries. If you have AC3 loaded then use 3.

SSL version level: This controls the SSL version level you want to operate on. The numbers '2' and '3' correspond to SSL versions 2 and 3 respectively. Using a '0' here allows the system to automatically switch between versions 2 and 3 depending on the capability of the remote server.

Keyring File and Path: To do SSL you must first have set up a key and obtained a certificate using DCM. In this process you will have created a keyring file in a folder somewhere on your iSeries machine. You now need to provide the name of the keyring file (with path) for SSL to use for your AS2 client.

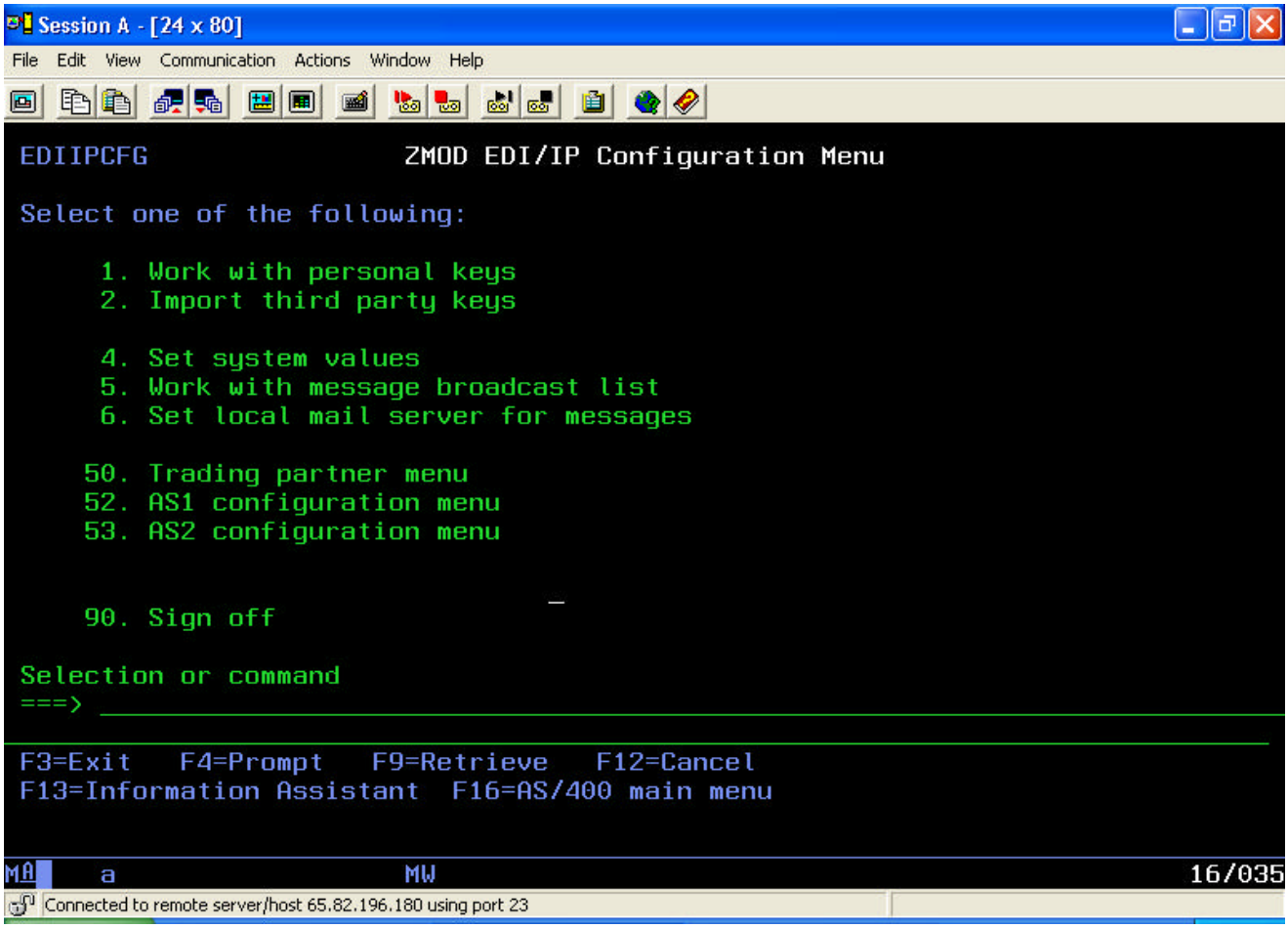
Keyring password: Leave this value blank.

This completes the setup of the AS2 client defaults.

6. Work with session log

The Configuration Menu.

The EDI/IP Configuration menu will allow you to do the static setup for your EDI/IP system. Most of the options on this menu are for first time configurations of general objects that are not specific to any transport protocol. It can be reached by using the command **GO EDIIPCFG**.



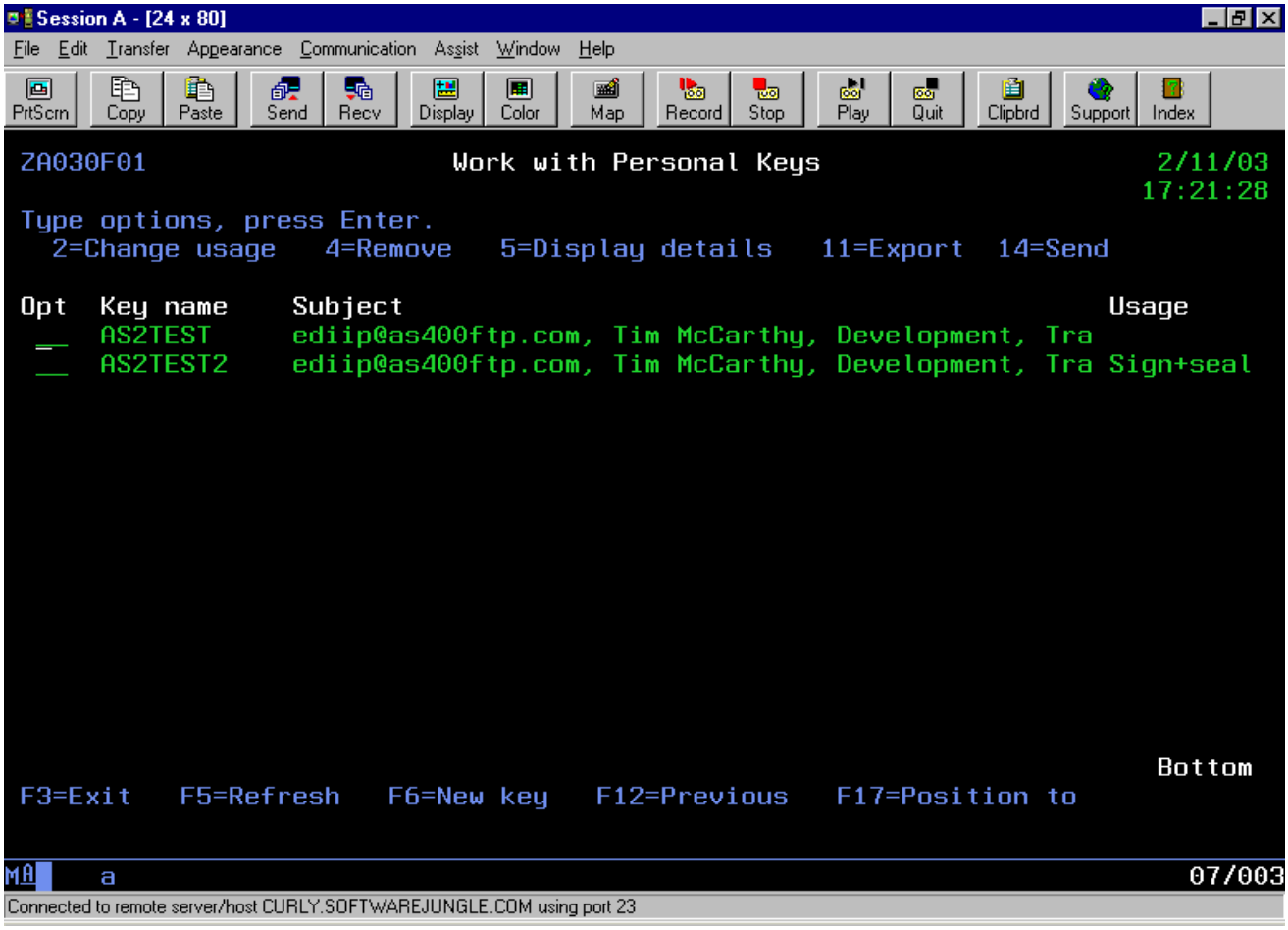
Options available from this menu are:

- *Work with Personal Keys.* This option lets you create and work with your company's local keys.
- *Import third party keys.* To load a key generated by a third party use this import function.
- *Set system values.* The system administrator's email address and various API program names are defined here.
- *Work with message broadcast list.* Messages issued by EDI/IP are logged to the message broadcast system. Use this option to work with users and email addresses in the broadcast list.

- *Set local mail server for messages.* The local mail server configuration is used by EDI/IP to send certificates and informational messages via email.
- *Trading partner menu.* Displays the trading partner menu.
- *AS1 Configuration menu.* Displays the AS1 configuration menu.
- *AS2 Configuration menu.* Displays the AS2 configuration menu.

1. Work with Personal Keys

In order to perform cryptographic functions such as encryption and digital signing you're going to need a personal key. This option allows you to create and work with your company personal keys. A personal key consists of a key pair, one part called the private key and the other the public key. The public key part of your key pair is distributed to your trading partners in a file called a certificate, which incorporates identifying details of the company and encapsulates the public key. Your trading partners will use this part of the key pair to encrypt data to send to you and to verify signed documents sent by you. You have the option of creating your own keys and certificates (self-signed certificates) through this EDI/IP function or you can obtain your keys and certificates from a third party source and import them into EDI/IP. You only need to have one personal key to exchange secure messages with all of your trading partners but you have the option of having several keys and using each for a specific use or trading partner.



The options available from this screen are:

2=Change Usage. Of all the keys that you have available to you, one will be designated as your default signing-key. In order to change the key usage and mark a key as the default key you would use this option.

4=Remove. Delete a personal key and certificate. Once a key has been deleted it cannot be recreated. You may create a new key with the same company details but it will be a different key and its' associated certificate will need to distributed to your trading partners.

5=Display details. To display more details about a particular key such as its' certificate file, serial number and validity dates use option 5.

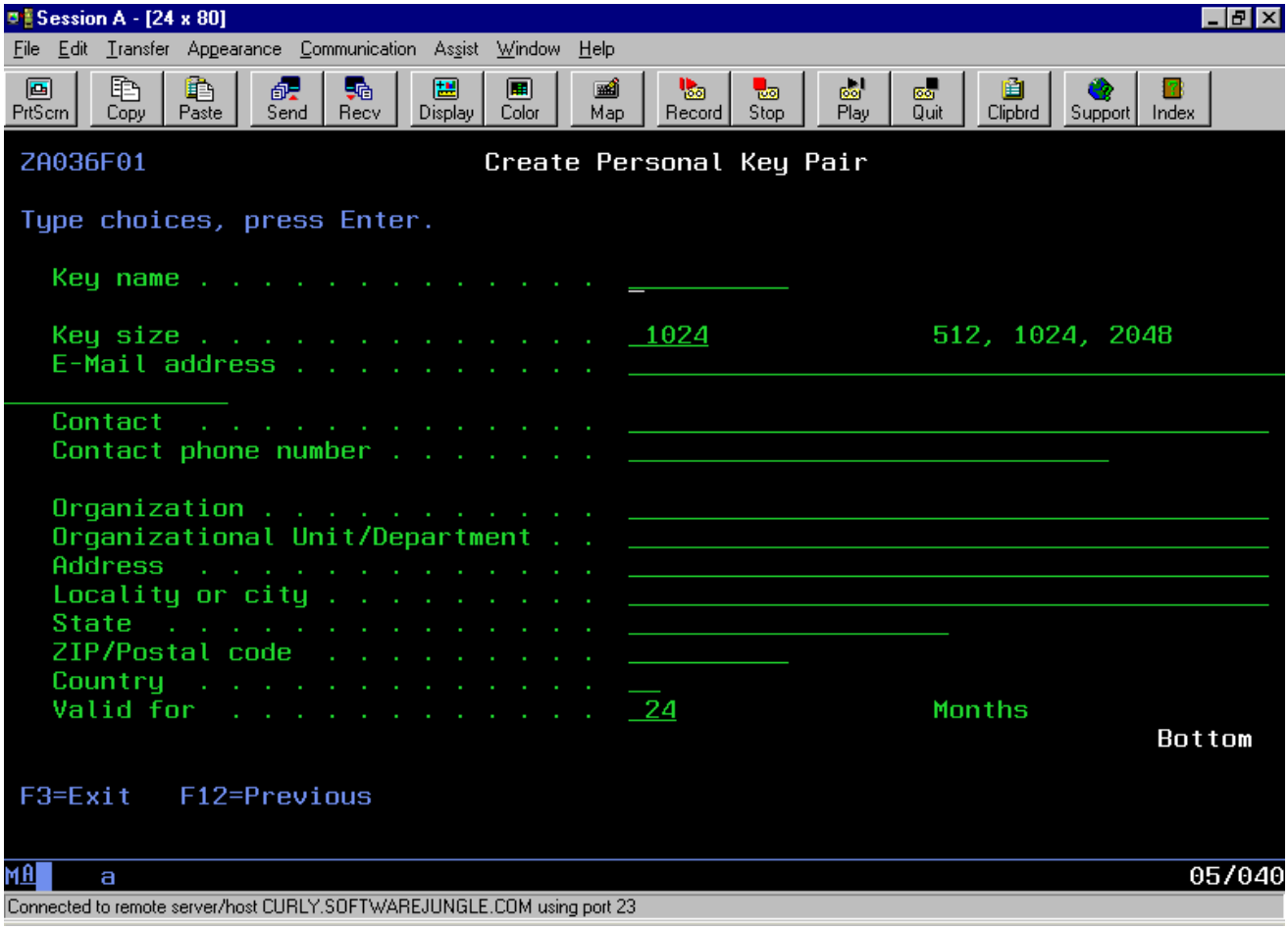
11=Export key. The export option exports this key pair so that it can be imported into another application.

14=Send. The send option will allow you to email a copy of the certificate associated with this key to one or more trading partners.

To create a new personal key use <F6-New key>.

Creating a New Personal/Local Key

This should be a task that you perform as part of your initial EDI/IP setup and perhaps only rarely after that. Your keys do not last forever - they have an in-built expiration date - so you will have to create new ones and distribute the new certificate from time to time. The validity period of your key is controlled by you on creation so you can decide how often you want to create keys and distribute certificates.



Key name: Your keys are given a name that you define so that you can identify them more easily. This name is relevant only to your EDI/IP system and is not part of the certificate that your trading partners receive. Any name will do but it should start with an alphabetic character and shouldn't contain embedded blanks.

Key size: As a general rule, the bigger the key size the more secure it is. However, bigger keys take much longer to process in encryption and signing routines so a balance is necessary. Our default of 1024 bits is the most currently accepted secure size.

Email address: This email address is used as one way to identify the owner of the certificate that will be generated and as such any email address can be used. A few AS1 systems however need this email address to match the email address that you will use for AS1 messaging, so, although the specifications don't require that it does, it would be a good email address to use. If you're not using AS1 then the email address of the system administrator will do.

The values from "Contact" through to "Country" are used to identify your company in the certificate that's generated and should be self-explanatory.

Valid for: A certificate has both a valid-from and a valid-to date. This value, given in months from the present date, is the length of time that the certificate generated with this key pair is valid for. After this time the certificate will expire and must be replaced.

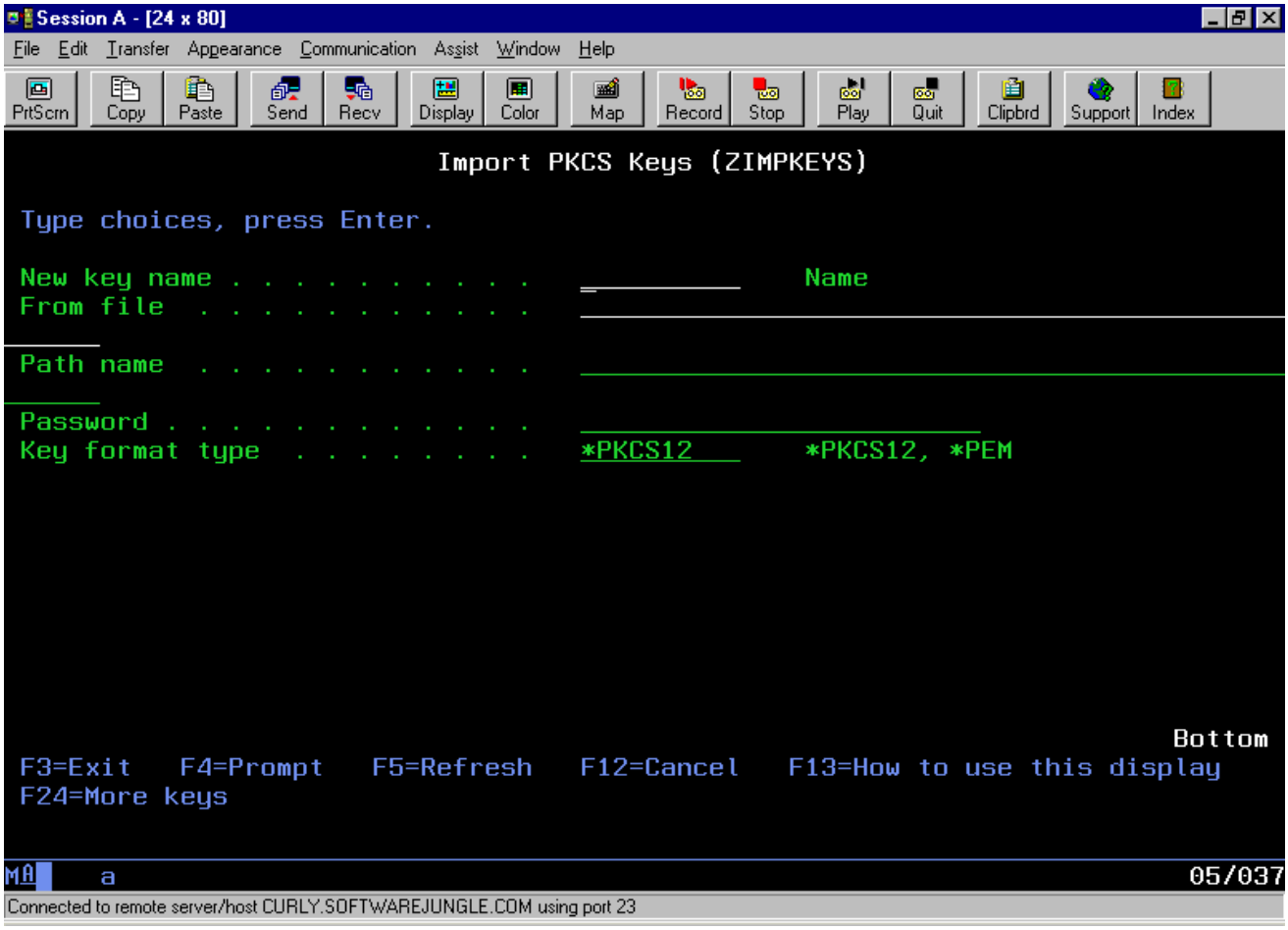
Once you press <enter> the key generation process will begin. This process might take a few minutes depending on the key size you chose and the resources available on the machine. If this is the first key you've generated then it will automatically be flagged as the default signing-key, otherwise you might need to mark it as such.

Important: Your private keys are stored in encrypted format in the IFS folder '/zmodediip/certs'. They will be named in the format 'zpkey*.zpv'. You cannot re-create a key once it has been deleted. You should ensure that these keys are saved somewhere safe should they ever be deleted and need to be restored.

2. Import third party keys

If you've obtained your personal key from a third party issuer or if you've exported a key from another product to use in EDI/IP then this import function will load the key and certificate into EDI/IP. In order to import the key into EDI/IP it should be in either **PKCS12** or **PEM** format (recognizable by the *.p12*, *.pfx* or *.pem* extensions). This menu option runs the command **ZIMPKEYS**.

If you're trying to import a trading partner's certificate then do not use this option - use the certificate import option on the trading partner menu instead.



Key name: This is the name you want to give to this key when it's imported into EDI/IP. Any name will do but it should start with an alphabetic character and shouldn't contain embedded blanks.

File name: Name the file that contains the key you want to import. You can give the file name with the full path or use the path parameter below for the path part of the name.

Path name: Enter the path to the file name given above unless the path is already included in the name.

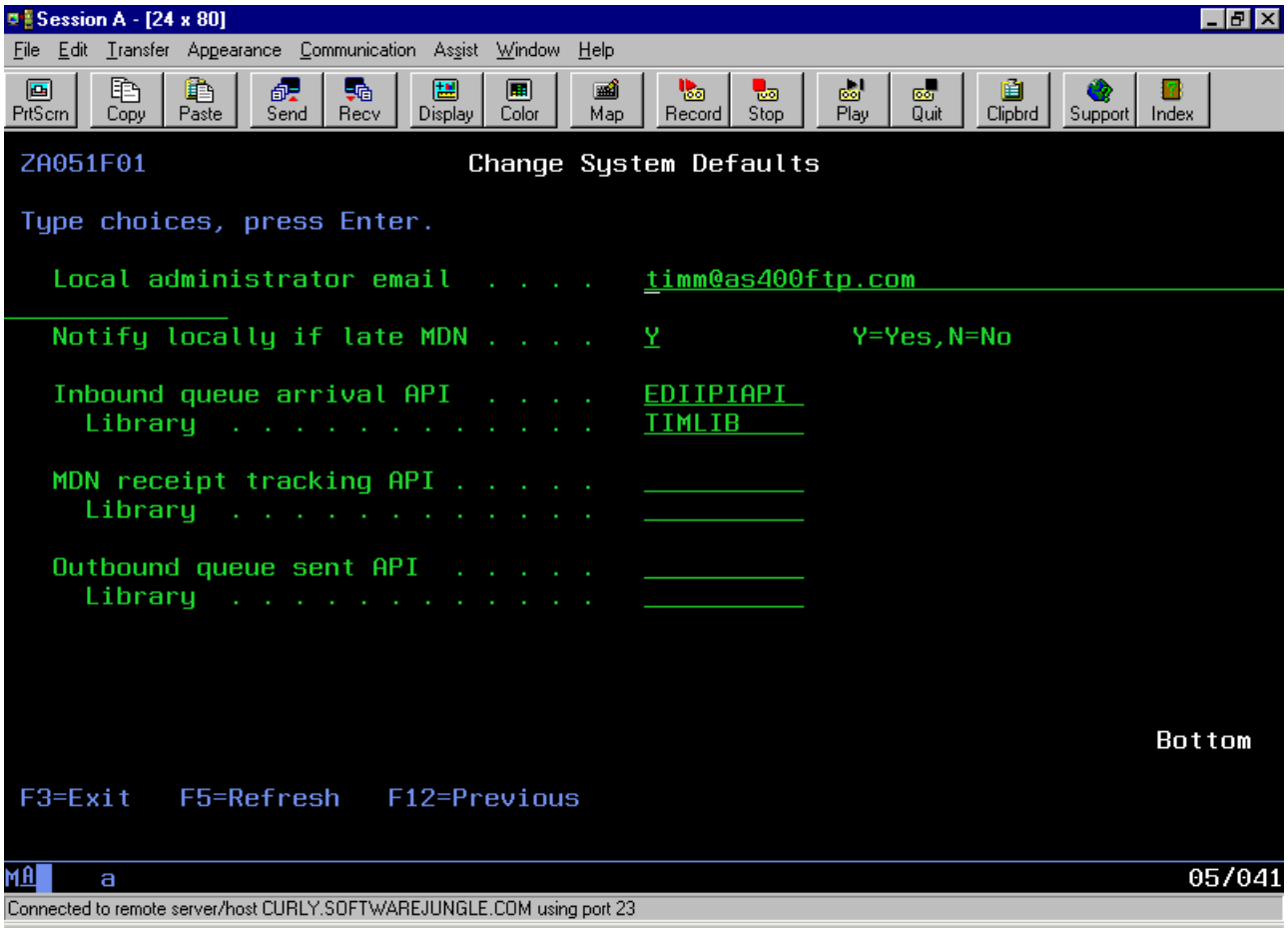
Password: The file that contains the key will have been encrypted with a password. Enter the password here.

Key format type: If the file is in a PKCS12 format (i.e. if the extension on the file name is .p12 or .pfx) then enter *PKCS12 here. If the key file is in PEM format enter *PEM.

Press <enter> and the key will be imported. Use the "Work with Personal Keys" option to get information on the key you've just loaded.

3. Set System Values

General system-wide parameter values are set here.



Local administrator email: If you want the system administrator to receive messages via email regarding the status of messages sent and received (e.g. late MDN errors) then provide that email address here. You should also have set up the local email server configuration.

Notify locally if late MDN: If an MDN is not received for a sent message within the specified time period it will be deemed late. In the case of a late MDN a message will be prepared and sent to the trading partner's system administrator notifying them of that fact so long as a contact email address is given. You can also have a similar email message sent to the local administrator if you enter a 'Y' here.

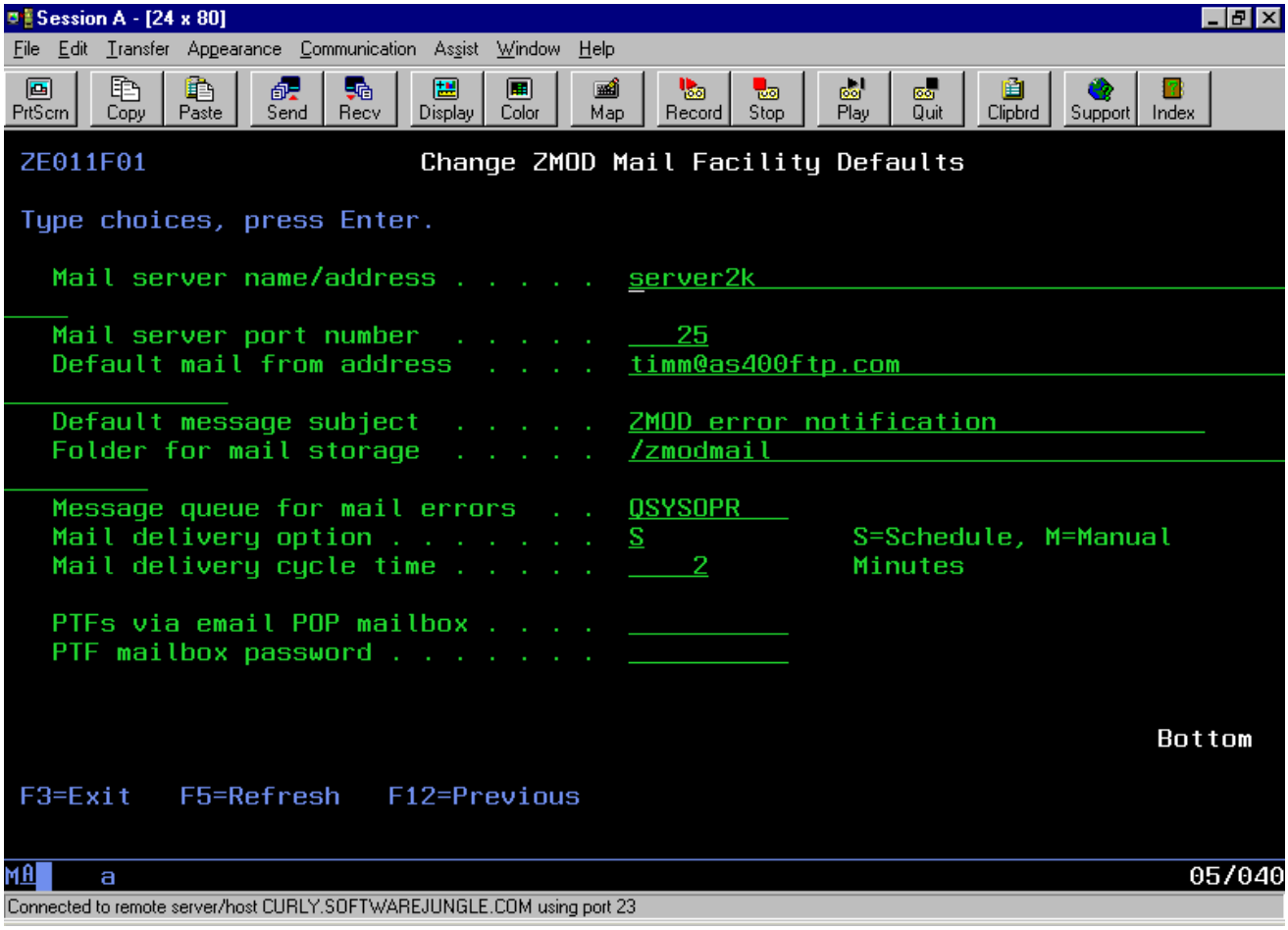
Inbound queue arrival API: If you would like a user program called when a new message arrives on the inbound queue then enter your program name and library here. The program must already exist. This program should be based on the sample Inbound Queue API provided. The API will not be called for received AS2 messages that are automatically processed because a synchronous MDN was requested. For these messages the AS2 Exported Files API will be called instead.

MDN receipt tracking API: If you would like a user program called when an MDN arrives on the inbound queue then enter your program name and library here. Details on the original sent message will also be given to the API. The program must already exist. This program should be based on the sample MDN Tracking API provided.

Outbound queue sent API: If you would like a user program called when a new message arrives on the inbound queue then enter your program name and library here. The program must already exist. This program should be based on the sample Inbound Queue API provided.

4. Set Local Mail Server for Messages

This option involves defining a mail server that can be used by certain EDI/IP functions such as the message broadcast system (if any email addresses are in the broadcast list), the certificate send options and local administrator notification messaging. This mail server setup is not related to any AS1 configurations you might make although both might use the same local mail server.



Mail server name/address: This value is the host name or IP address of your local mail server. If a host name is used it should be resolvable to an IP address.

Mail Server Port Number: If your mail server is using a port number other than the standard port 25 then you can change it here.

Default Mail-From Address: This is the email address that will be used as the default email "From:" address if no alternative is provided. This email address will need to be acceptable to the mail server.

Default message subject: This will be the default subject line on messages issued automatically by the system.

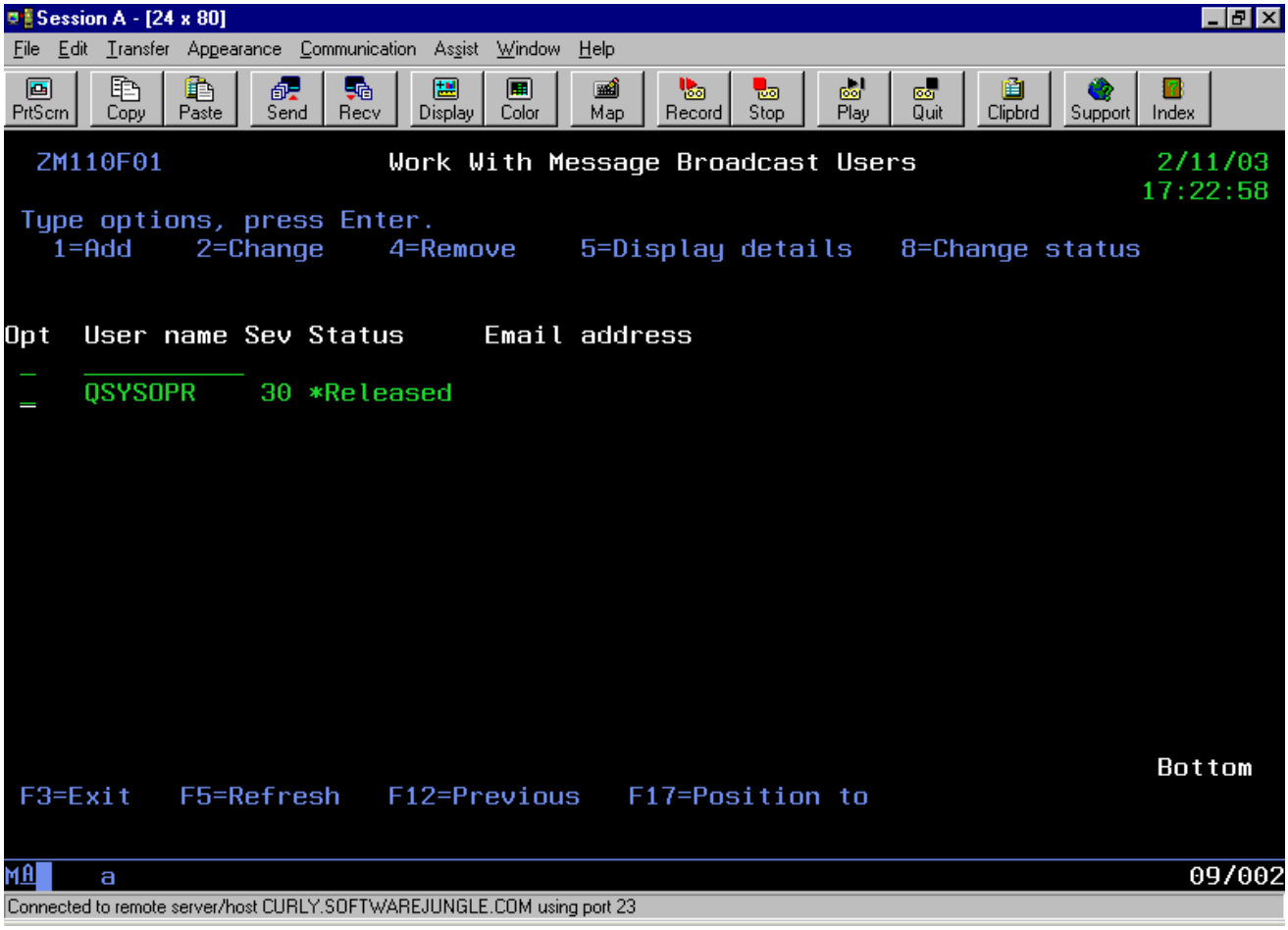
Folder for mail storage: Email messages are temporarily stored in an IFS folder before delivery to the mail server. The folder '/zmodmail' is automatically created on installation. If you want to use a different folder then you must create it and enter the name here.

Message Queue for Mail Errors: If errors are found in generating or sending these emails then error messages will be sent to the message queue named here.

Mail Delivery Option: When email messages are created they can either be sent to the mail server immediately or stored temporarily until a user job using the command XXXXXXX delivers the mail. Choose either 'S' for scheduled delivery or 'M' if a user program will initiate the delivery. The manual option is intended for configurations where a dial out needs to done to contact the mail server.

Mail Delivery Cycle Time: If the mail delivery option above is set to 'S' for scheduled delivery then this is the cycle time, in seconds, between mail delivery attempts.

PTF's via email POP mailbox: Not used.

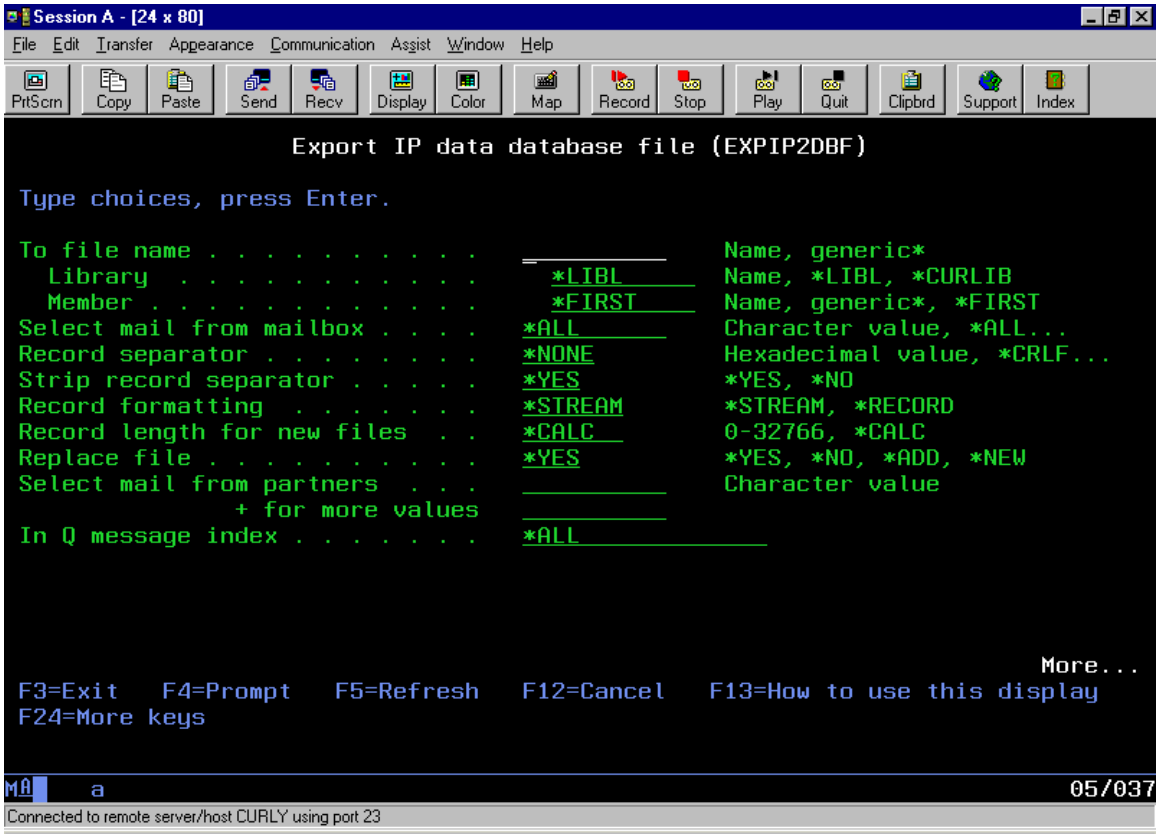

Session A - [24 x 80]
File Edit Transfer Appearance Communication Assist Window Help
PrtScrn Copy Paste Send Recv Display Color Map Record Stop Play Quit Clipbrd Support Index
ZM110F01 Work With Message Broadcast Users 2/11/03
17:22:58
Type options, press Enter.
1=Add 2=Change 4=Remove 5=Display details 8=Change status
Opt User name Sev Status Email address
QSYSOPR 30 *Released
Bottom
F3=Exit F5=Refresh F12=Previous F17=Position to
MA a 09/002
Connected to remote server/host CURLY.SOFTWAREJUNGLE.COM using port 23

Command Reference

EXPIP2DBF – Export Received EDI/IP Data into Database File.

You can use this command to export data from the EDI/IP inbound queue into a database file. Released mail items that match the selection criteria will be chosen. Any decryption and signature verification will be done at the time. You can either pre-create the database file you want the data in or have this command create the file for you. Generic file and member names can be used to have the command generate new files or members based on a sequential naming pattern. If a generic naming pattern is used you can have each mail item exported to a new file/member or have all the selected data exported to a single new file/member.

Here are the command parameter definitions and their usage.



To file name: This should be the name of the file you want to receive data into. Either a specific or generic file name can be given. If a specific file name is supplied then either the file will be created if it doesn't exist or data will be written to the existing file. If a single member or *FIRST is specified then new data will either replace or append to data already in the file depending on the "replace" parameter value. If a generic file name is used (i.e. the asterisk character * is used in the file name portion) then a new file will be created each time. The new file name will be based on the pattern given with a sequence

number value replacing the generic ("*") component. EXPIP2DBF creates file names from a generic name by taking the generic prefix value and adding a sequence number to make up a ten character file name. The sequence number used will be the lowest number needed to generate a file name that does not already exist. If the file name already exists in the library given then the REPLACE parameter cannot be *NO or a exception message (UZM5033) will be generated.

example: to receive a remote file into a new local file that has a name beginning with the letters INVF you would specify a receive-to file name of INVF. EXPIP2DBF will generate file names of the pattern INVF000000, INVF00001 etc. Useful tools for finding and processing files with a certain pattern are the commands SCNPRCDBF and SCNDBFPTRN.*

Library name: This is the name of the library that contains or will contain the To-File named above. You can use the special value of *CURLIB to use the job's current library.

Member: Enter either the name of the member you wish to receive data into or *FIRST to use the first member in the file. If the member does not already exist it will be created. If you did not use a generic file name you have the option of specifying a generic member name here. EXPIP2DBF will build member names in the same way it builds file names from generic values.

Select mail from mailbox: If you received this data via an **AS1 POP** mailbox then you can select the data to process based on the POP mailbox name it was pulled from. The default of *ALL will select all data regardless of whether it was received via POP. If you would like to export only **AS2** data then you can use the special value of **AS2DATA* for this parameter.

***Record separator:** If the data you're exporting has record separator characters (e.g. carriage return/line feed) that you need to identify because you either want to strip them off or you want the data written in record format then identify those characters here. Typically this value would be a carriage return/line feed combination (*CRLF). Valid values are *NONE, *CRLF, *CR only, *LF only or any character value(s) entered in hexadecimal notation. To enter a single hex character as the record separator then key that characters hex equivalent followed by '00'. Most EDI data does not have these record separator characters.*

Strip record separator: To have any characters identified above stripped from the data stream answer *YES here.

Record Formatting: Valid options are *RECORD and *STREAM. The record option dictates that each record received (as determined by the record separator characters) will be written as a record in the file. Received records that are longer that the maximum record length of the file (assuming the file already exists) will be wrapped into multiple records. The stream option will cause the data records received to be written to the file as a single stream of data with each record's data following the previous. If the data you're exporting is **EDI** data then **STREAM* is probably the option you want.

Record length for new files: If the file does not already exist or a generic file name is used, you can specify the record length you want used when the file is created. If you do not specify a record length then the record length will be calculated as follows:

- Where the record format option is *STREAM then the default record length will be 1024 bytes.
- The first 32k bytes of data received will be examined and the longest record length calculated. This record length will be the record length of the file. If no record separators are found in the received stream then the file will be created with a record length equal to the number of bytes in the received file with a maximum record length of 32766 bytes.

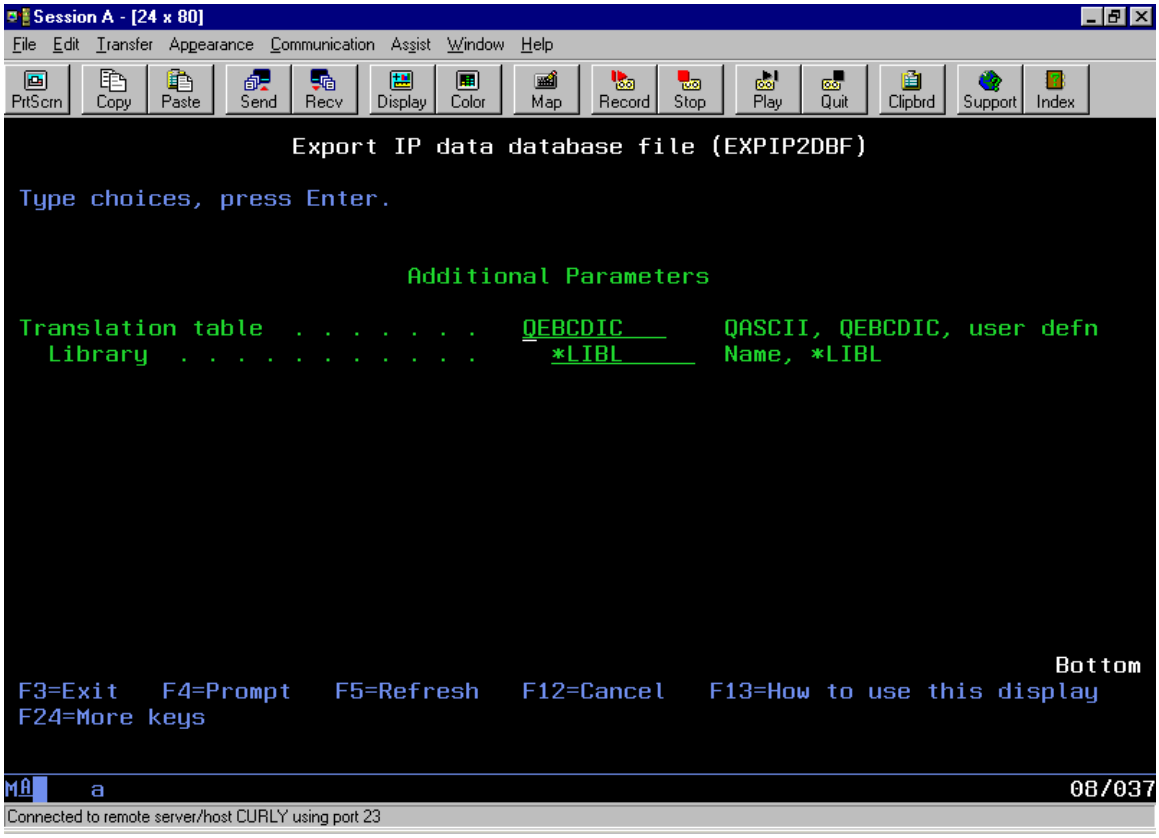
Replace file: Valid options are *YES, *NO, *ADD and *NEW. The REPLACE(*YES) option will over-write any data currently in the file. The *ADD option will add any data records received to the end of the file. If the file already exists, you must use either *YES or *ADD. If you use either a generic file or member name then you can use the *NEW option here to specify that each mail item selected to export will have its' data written to a new file. For all other options all the mail item data exported will be written to a single file.

Select mail from partners: If you want to select specific messages from the inbound queue based on the sending trading partner then you can use this parameter to specify the names of up to 10 trading partner names.

In Q message index: Every message on the EDI/IP inbound queue has a unique index number. If you want to export a specific message then you can do that by entering its' index number here. You can find the message index by looking at the message details (option 2). If you want to select specific messages from the inbound queue based on the sending trading partner then you can use this parameter to specify the names of up to 10 trading partner names.

Press <F10> to get more values.

Translation table: This parameter controls the translation table used to convert the EDI/IP data before outputting to the database file. The default is QEBCDIC which assumes that the data being exported needs to be converted to the EBCDIC character set on export.



Informational Messages

USM0019 – Export of EDI/IP messages to file ended.

Exception Messages

CPF0001 – Error in command. Problem will be reported via diagnostic messages such as:

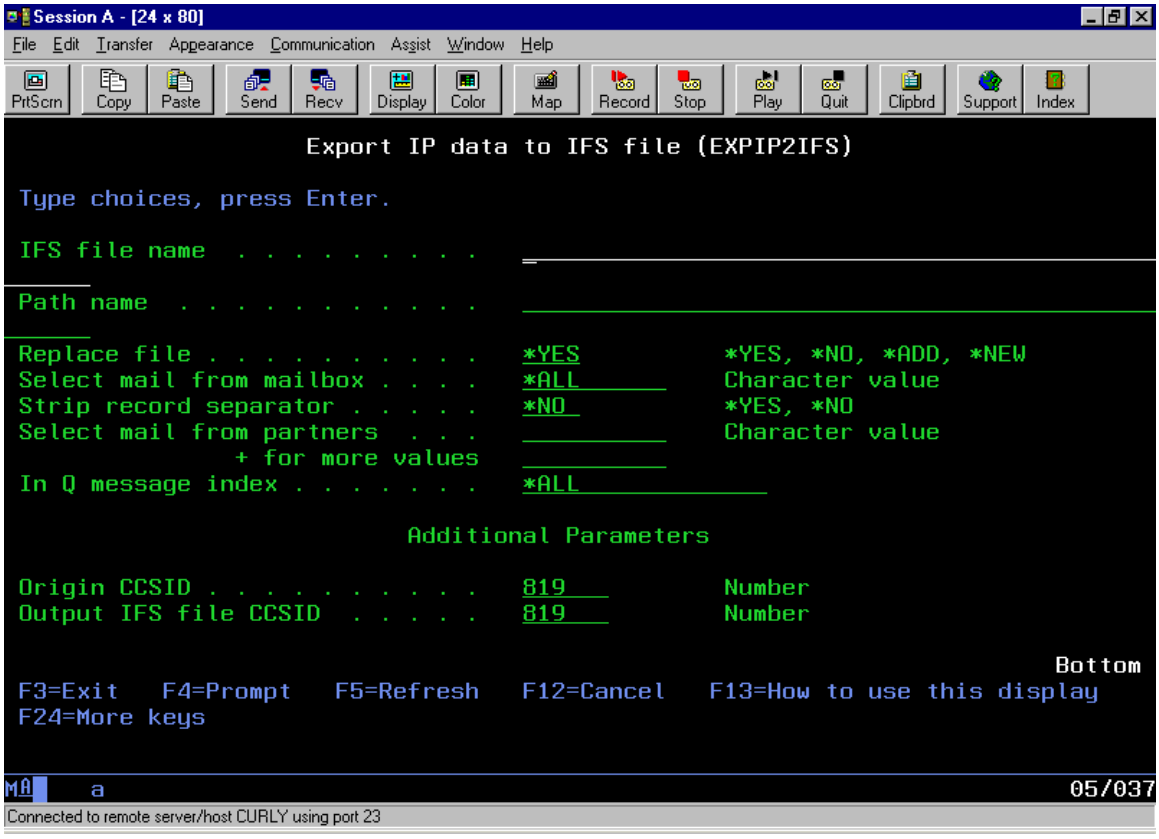
- UZM0515 – Name or generic name not valid.
- UZM0522 – Name entered not a valid name. The file name entered was not valid.
- UZM0523 – Cannot have both generic file and member.

USM9024 – No messages exported.
USM9025 – Export of EDI/IP messages ended in error. Check joblog.
UZM0152 – File must exist if *LIBL used for library name.
UZM5033 – File &1 in library &2 already exists.

EXPIP2IFS – Export Received EDI/IP Data into IFS File.

You can use this command to export data from the EDI/IP inbound queue into one or more files in the IFS file system. Released mail items that match the selection criteria will be chosen. Any decryption and signature verification will be done at the time. You can either pre-create the IFS file you want the data in or have this command create the file for you. Generic file names can be used to have the command generate new files based on a sequential naming pattern. If a generic naming pattern is used you can have each mail item exported to a new file or have all the selected data exported to a single new file.

Here are the command parameter definitions and their usage.



IFS file name: This is the name of the IFS file(s) that you want to export the data to. If you enter a specific file name then the data will be written to that file. If you've set the option REPLACE(*****YES) then any data currently in the file will be overwritten. If you use REPLACE(*ADD) then the new data will be appended to the file. You can also use a generic or wildcard format to enable the command to generate a new file for this stream . To use this generic format place an asterisk after any file name prefix you care to use.

E.g. If you enter the file name "EDI*.TXT" then the command will generate new files names of the form "EDI1.TXT", "EDI2.TXT" etc. where the sequence number used is

the lowest number that can be used to create a file that doesn't already exist in the IFS path named.

Path name: This is the name of the IFS path or folder name that the file exists in or will be created in.

Replace file: Valid options are *YES, *NO, *ADD and *NEW. The REPLACE(*YES) option will over-write any data currently in the file. The *ADD option will add any data records received to the end of the file. If the file already exists, you must use either *YES or *ADD. If you use a generic file name then you can use the *NEW option here to specify that each mail item selected to export will have its' data written to a new file. For all other options all the mail item data exported will be written to a single file.

Select mail from mailbox: If you received this data via an **AS1 POP** mailbox then you can select the data to process based on the POP mailbox name it was pulled from. The default of *ALL will select all data regardless of whether it was received via POP. If you would like to export only **AS2** data then you can use the special value of **AS2DATA* for this parameter.

Strip record separator: To have any record separator characters stripped from the data stream answer *YES here. The record separator characters are defined in the RECSEP parameter.

Record separator: *If the data you're exporting has record separator characters (e.g. carriage return/line feed) that you need to identify because you want to strip them off then identify those characters here. Typically this value would be a carriage return/line feed combination (*CRLF). Valid values are *NONE, *CRLF, *CR only, *LF only or any character value(s) entered in hexadecimal notation. To enter a single hex character as the record separator then key that characters hex equivalent followed by '00'. Most EDI data does not have these record separator characters.*

Select mail from partners: If you want to select specific messages from the inbound queue based on the sending trading partner then you can use this parameter to specify the names of up to 10 trading partner names.

In Q message index: Every message on the EDI/IP inbound queue has a unique index number. If you want to export a specific message then you can do that by entering its' index number here. You can find the message index by looking at the message details (option 2). If you want to select specific messages from the inbound queue based on the sending trading partner then you can use this parameter to specify the names of up to 10 trading partner names.
Press <F10> to get more values.

Origin CCSID: Data written to the IFS file will be translated between character set ID's if the origin and output CCSID values differ. This value determines the CCSID of the data in the received message.

Output IFS file CCSID: If the IFS file doesn't exist then this is the CCSID attribute that will be assigned to the file at creation. Data translation will be done with this CCSID value as the output character set type.

Informational Messages

USM0019 – Export of EDI/IP messages to file ended.

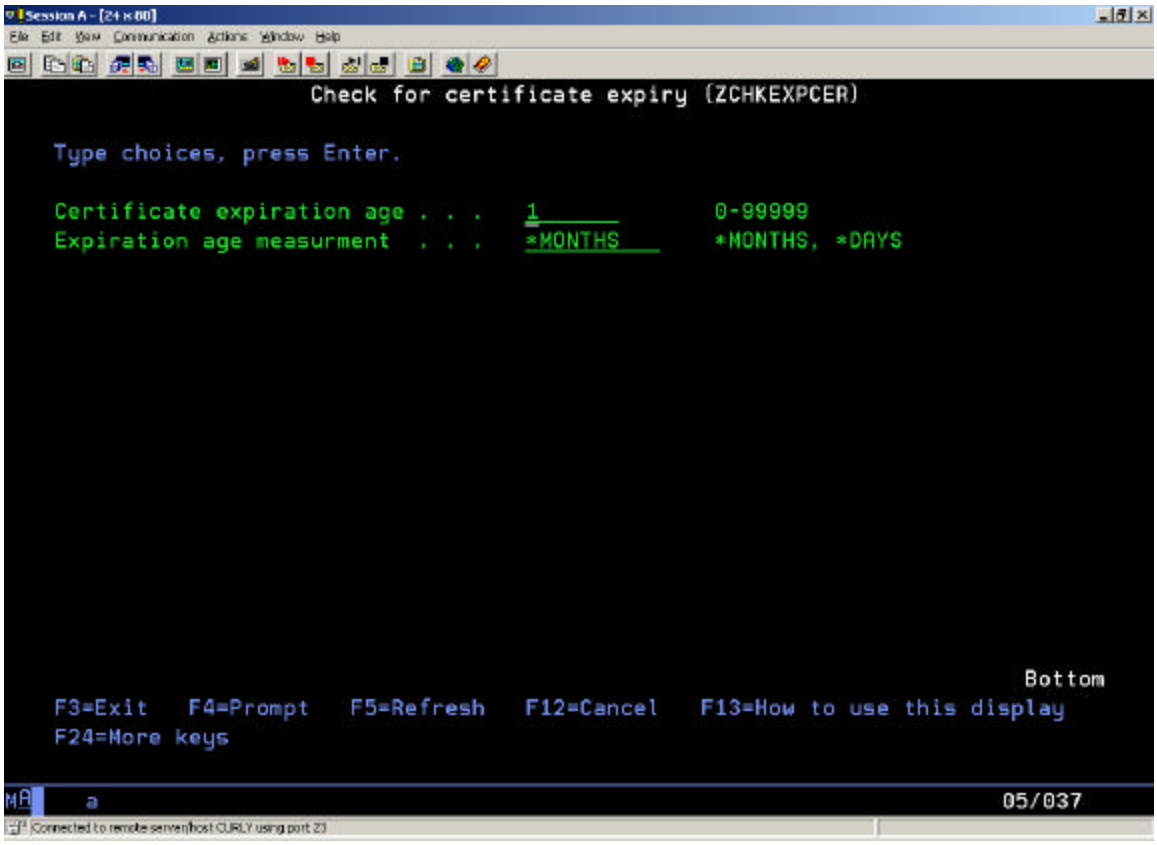
Exception Messages

CPF0001 – Error in command. Problem will be reported via diagnostic messages.
UIF9002 – No such IFS file or directory.
USM9024 – No messages exported.
USM9025 – Export of EDI/IP messages ended in error. Check joblog.

ZCHKEXPCER – Check for certificate expiry.

*This command should be used to check that your local keys and partner certificates are still valid and to warn you before any of these certificates are about to expire. Usually this command would be placed in an EDI/IP scheduled service program that would probably also include the EDI/IP purge process (**PRGSMTPDTA**). You can specify the period in the future within which you'd like to know of certificates that are about to expire. This period should be chosen to provide sufficient time for you to:*

- *Generate a new key and certificate and distribute it to your trading partners if it's a local key that's about to expire.*
- *Contact your trading partner about getting a new certificate if their certificate is about to expire.*



Messages giving details about the keys and certificates that are not valid or that will expire will be written to the truExchange broadcast system. In addition this command will issue an exception message to notify you that you'll need to look in the log and do something.

This command will verify the following:

- *That you have a default signing key set.*
- *That the default signing key and any local keys specified as overrides on a trading partner profile are valid and will not expire within the selected timeframe.*
- *That every trading partner set to receive encrypted data has at least one certificate attached and that at least one of these certificates is valid and will not expire within the selected timeframe.*

Here are the command parameter definitions and their usage.

Certificate expiration age: This is where you set the period within which all assigned certificates are valid. The measure of this value is set in the following parameter. The default is to check one month ahead.

Expiration age measure: Choose either ****DAYS*** or ****MONTHS*** as the measurement value of the *"Certificate expiration age"* parameter above.

Informational Messages

USM0103 – Assigned certificates appear valid for the specified period.

Exception Messages

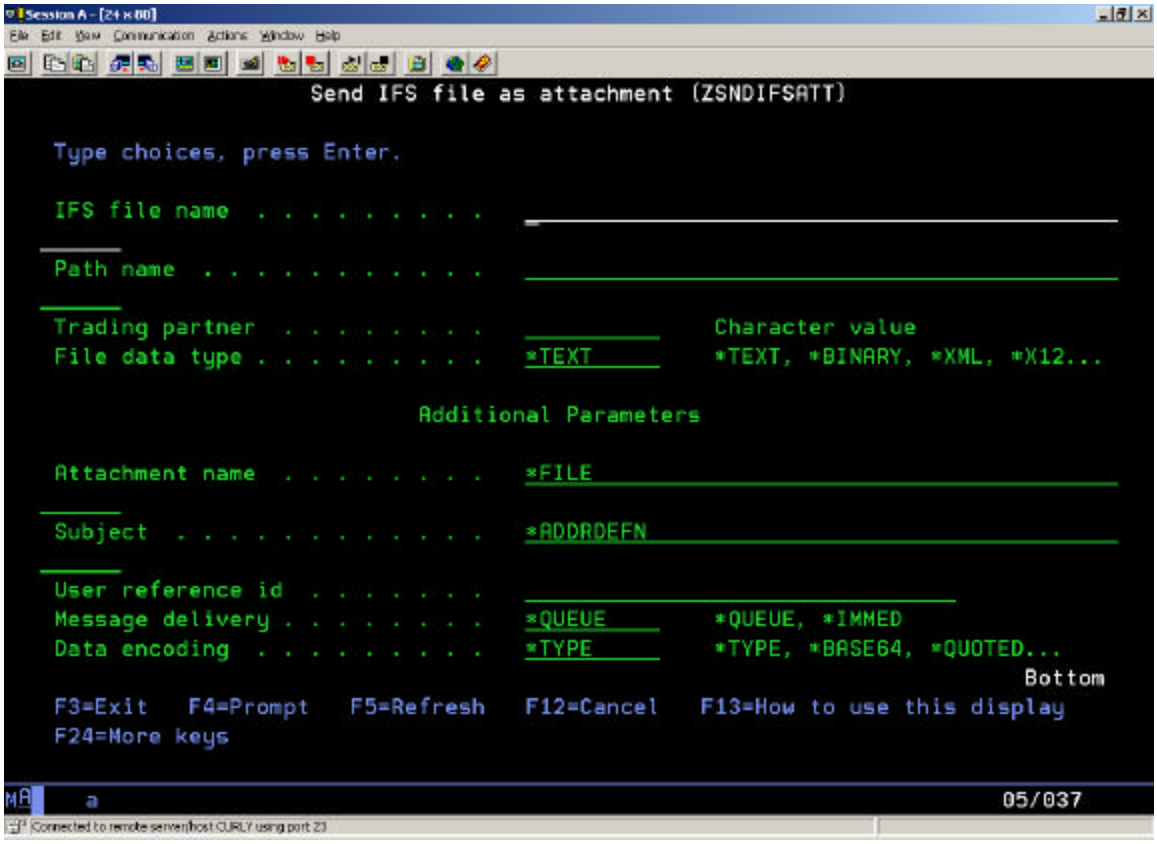
CPF0001 – Error in command. Problem will be reported via diagnostic messages.

USM9050 – Check the ZMOD log. Some certificates are about to expire.

ZSNDIFSATT – Send an IFS File to a Trading Partner.

This EDI/IP command is used to send an IFS file to a single trading partner. The data content is not parsed or changed in any way. The security settings and primary delivery method configured for the recipient will be used to secure, package and deliver the message.

Here are the command parameter definitions and their usage.



IFS file name: This is the name of the IFS file that you want to send. You can either specify the full file name with path or just fill in the file name here and give the path name in the parameter below. This file must exist at the time this command is run or an exception message will be issued.

Path name: The name of the IFS path or folder name that the file exists in.

Trading Partner: Enter the EDI/IP name of the trading partner you want to send this file to.

Press <F10> to get more values.

File data type: This value identifies the content type to the trading partner and can affect the content type encoding. Choose one of either *TEXT, *BINARY, *XML, *X12,

*EDIFACT, or *EDIOTHER. The file data type controls the content-type value of the message.

Content type: This parameter appears if *USER is chosen as the file data type. Enter the value you want to supply as the content type on the message.

Attachment name: This is the name that will be given to the file attachment in the email. The default of *FILE means that the actual file name will be used.

Subject: Type the text that will be used as the subject line in the secure message. If *ADDRDEFN is left as the default then the subject line in the trading partner definition will be used.

User reference id: Up to 30 characters can be used as a reference value that will be stored with this message. The reference value is displayed on the outbound queue message details and will be passed to the message-sent API program.

Message delivery: If you want the message sent immediately then use *IMMED here. If you choose *QUEUE then the message will be added to the outbound queue in a released state and will be sent when one of the delivery commands such as SNDMSGSMTP (send AS1 messages) or SNDMSGHTTP (send AS2 messages) is run.

Data Encoding: This parameter controls the data encoding method used. If the transport method is AS2 then no content type encoding is performed. If the default of *TYPE is used then the file data type will control the encoding. File data will be base64 encoded except if it's marked as *TEXT in which case it will be encoded as quoted-printable. Other options are *BASE64, *QUOTED and *NONE.

Informational Messages
None.

Exception Messages
CPF0001 – Error in command. Problem will be reported via diagnostic messages.

UIF9001 – Not authorized to this operation on this file
UIF9002 – No such directory or path.
UIF9007 – Could not open file requested.
UIF9009 – No IFS file matches the name given.

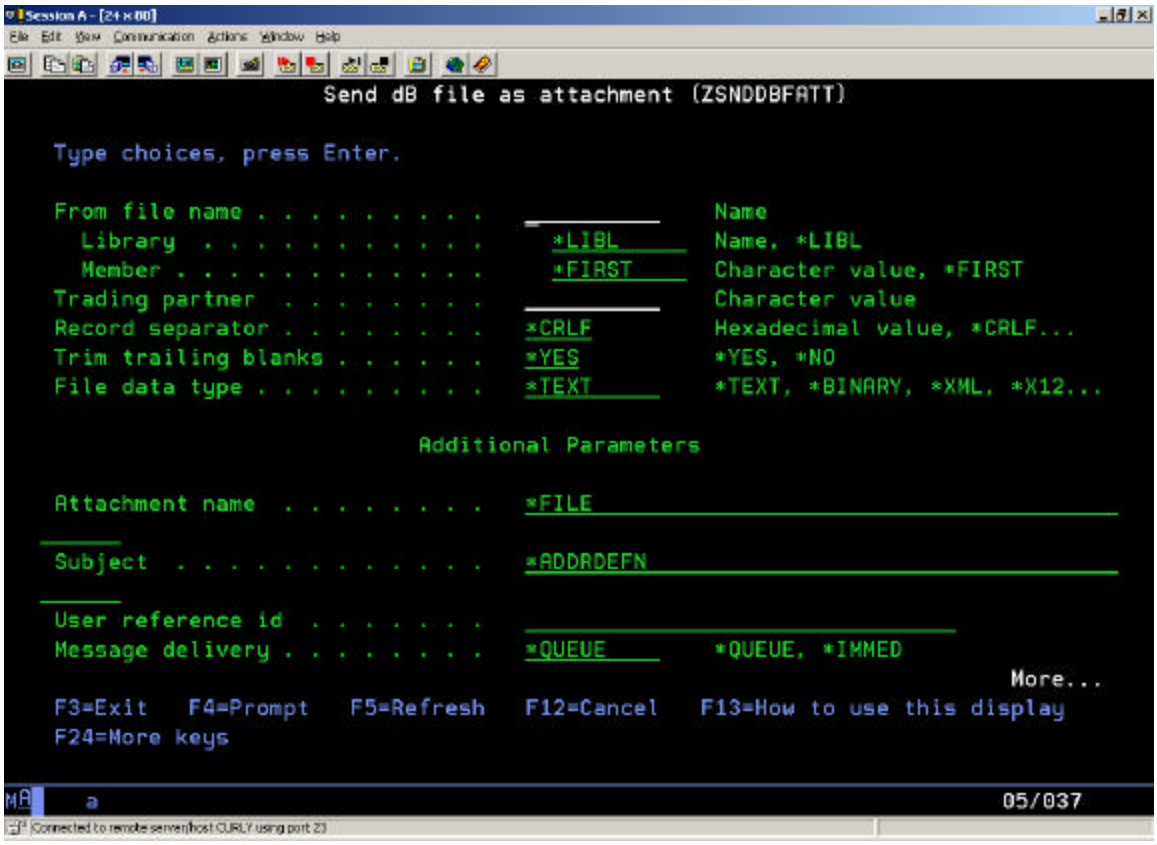
USM0213 – Receiver name/trading partner not defined.
USM9109 – Could not create mail item header. See log.

Note: If you've requested message delivery of *IMMED then the messages from the appropriate delivery request (either SNDMSGSMTP for AS1 and SNDMSGHTTP for AS2) should also be monitored for.

ZSNDDBFATT – Send a Database File to a Trading Partner.

This EDI/IP command is used to send a database file to a single trading partner. Apart from optional record separator characters, record trimming and character set translation the data content is not parsed or changed in any way. The security settings and primary delivery method configured for the recipient will be used to secure, package and deliver the message.

Here are the command parameter definitions and their usage.



From file name: This is the name of the database file that you want to send. This file must exist at the time this command is run or an exception message will be issued.

Library: The library name that the file resides in. The special value of *LIBL can be used.

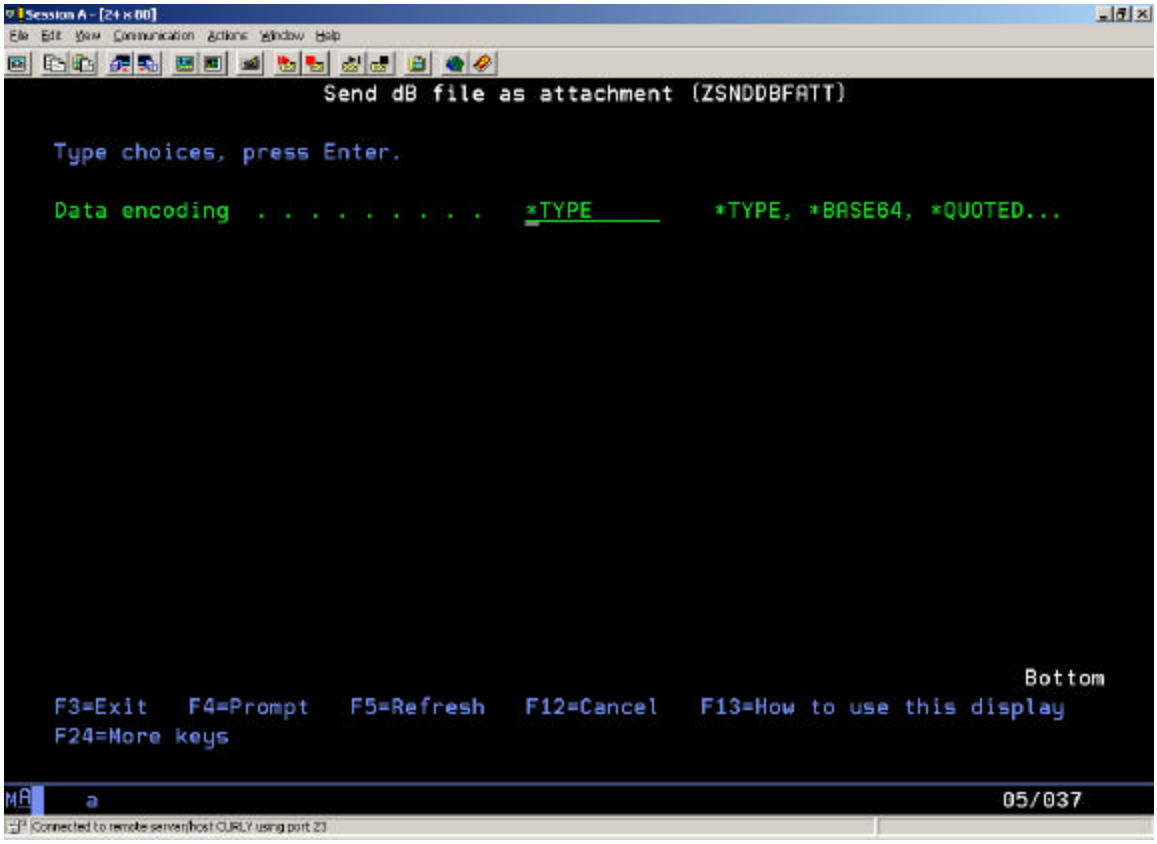
Member: The member name in the file to send. The default will select the data in the first member.

Trading Partner: Enter the EDI/IP name of the trading partner you want to send this file to.

Record separator: *If this is a file of text that's being sent then you probably want to add a record separator between database file records. Typical record separators would be *CRLF (carriage return/line feed) or just *LF (line feed). If you don't want record separators inserted then specify *NONE here.*

Trim Record Blanks: *Your file might contain records with a number of blanks trailing each record. To trim these blanks off before a record separator is added then use *YES here.*

Press <F10> to get more values.



File data type: This value identifies the content type to the trading partner and can affect the content type encoding. Choose one of either *TEXT, *BINARY, *XML, *X12, *EDIFACT, *EDIOTHER or *USER. The file data type controls the content-type value of the message.

Content type: This parameter appears if *USER is chosen as the file data type. Enter the value you want to supply as the content type on the message.

Attachment name: This is the name that will be given to the file attachment in the email. The default of *FILE means that the actual file name will be used.

Subject: Type the text that will be used as the subject line in the secure message. If *ADDRDEFN is left as the default then the subject line in the trading partner definition will be used.

User reference id: Up to 30 characters can be used as a reference value that will be stored with this message. The reference value is displayed on the outbound queue message details and will be passed to the message-sent API program.

Message delivery: If you want the message sent immediately then use *IMMED here. If you choose *QUEUE then the message will be added to the outbound queue in a released state and will be sent when one of the delivery commands such as SNDMSGSMTP (send AS1 messages) or SNDMSGHTTP (send AS2 messages) is run.

Data Encoding: This parameter controls the data encoding method used. If the transport method is AS2 then no content type encoding is performed. If the default of *TYPE is used then the file data type will control the encoding. File data will be base64 encoded except if it's marked as *TEXT in which case it will be encoded as quoted-printable. Other options are *BASE64, *QUOTED and *NONE.

Informational Messages
None.

Exception Messages
CPF0001 – Error in command. Problem will be reported via diagnostic messages.

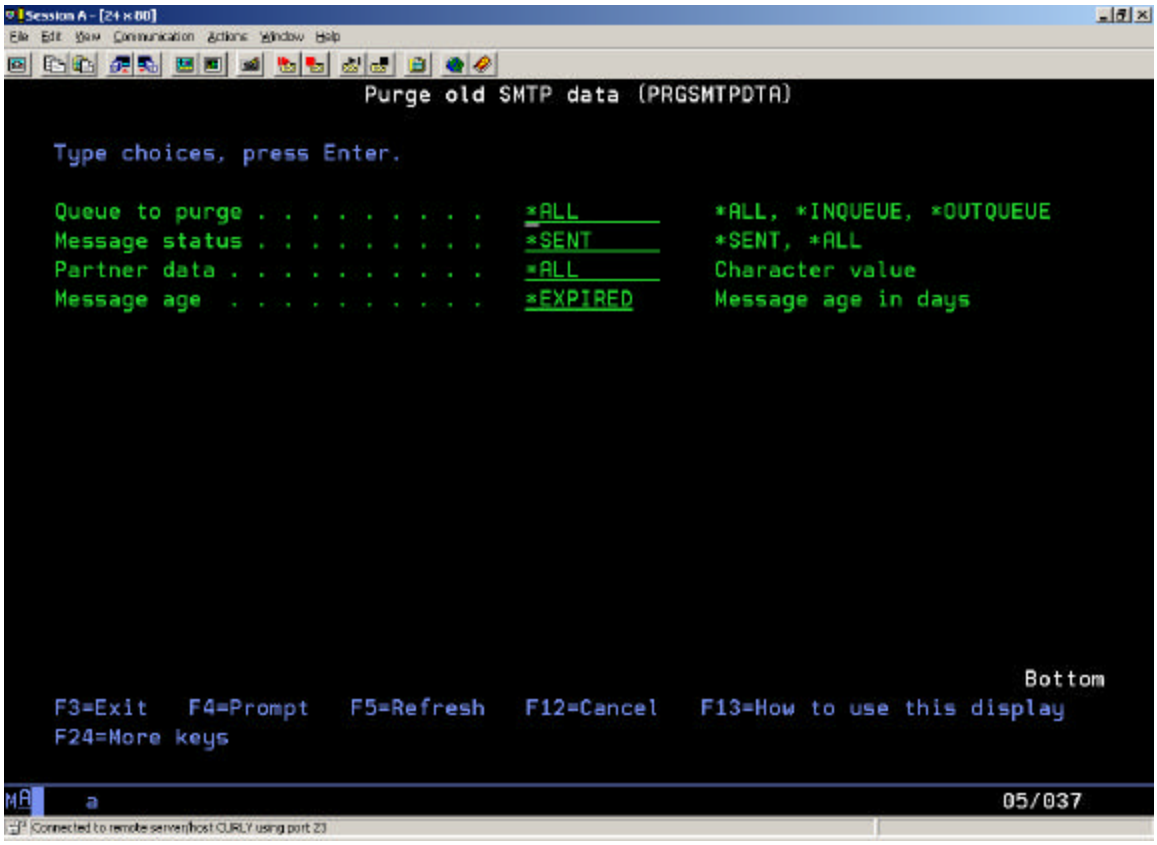
USM0213 – Receiver name/trading partner not defined.
USM9110 – Could not access database file.
USM9109 – Could not create mail item header. See log.
USM9111 – Memory allocation problem.

UZM0605 – File/member is empty

Note: If you've requested message delivery of *IMMED then the messages from the appropriate delivery request (either SNDMSGSMTP for AS1 and SNDMSGHTTP for AS2) should also be monitored for.

PRGSMTPDTA – Purge Data From EDI/IP System.

Use this command to removed old and unwanted data from your EDI/IP inbound and outbound queues. Usually this command would be placed in a CL service routine that is scheduled to run at intervals – perhaps monthly. This purge program can be run at any time and will not effect the generation and processing of inbound or outbound messages.



Here are the command parameter definitions and their usage. A message must meet all the selection criteria to be purged.

Queue to Purge: You can choose to separately purge your inbound and outbound queues. Select one of either *INQUEUE, *OUTQUEUE or *ALL to purge both.

Message status: This value determines whether the date and partner selection options are applied to outbound-sent and inbound-processed messages or all messages regardless or current state. Select *SENT to attempt to purge outbound messages that have been sent and inbound messages that have been purged.

Partner data: If you only want to select messages for a certain trading partner only then enter the partner name here.

Message age: This parameter determines the age of messages to be purged. The age is calculated from the date the message was created or received depending on whether it's an outbound or inbound message. If the default value of *EXPIRED is chosen then the message age is taken from the trading partner setup. Messages that are older than the number of days specified in the trading partner definition ***when the message was created*** will be eligible for purging. If a number is used then any messages older than that number of days can be purged regardless of the trading partner message aging parameter.

Informational Messages

None.

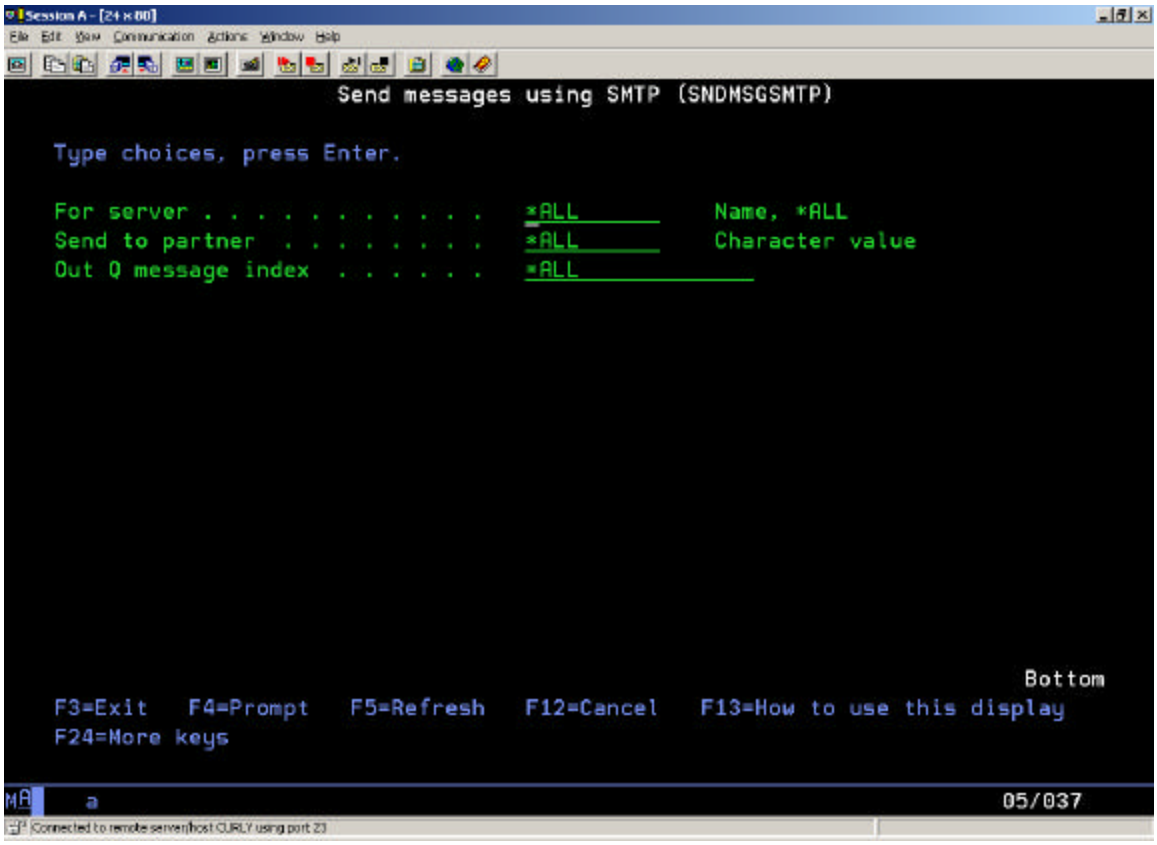
Exception Messages

None.

SNDMSGSMTP – Send AS1 Messages.

This command is used to deliver AS1 messages from the outbound queue to the SMTP server that's defined to handle messages for each trading partner. Unless a specific message index is used, only messages in a released status will be attempted. If a message cannot be delivered it will left in a released state for a future retry when this command is run again unless delivery has already been attempted for the maximum number of tries defined for that trading partner. In that case it will be marked as "failed". If the response from the SMTP server indicates that the message is faulty and could never be delivered it will be immediately marked as "failed". If the "outbound queue sent API" is defined it will be called after the delivery attempt of each message.

Here are the command parameter definitions and their usage.



For server: To deliver outbound messages selectively to a single SMTP server then name that server here. This value is typically used if a dial-up needs to be initiated before a certain server can be reached but it could be used in any circumstance when multiple SMTP servers are defined and they need to be contacted separately.
Send to Partner: To deliver a single partner's messages at this time enter the trading partners name here.

OutQ message index: Each mail entry on the EDI/IP outbound queue has a unique index number. To send a specific message enter the message index here or choose option 14 from the outbound queue.

Informational Messages

USM0008 – Send of EDI/IP messages ended normally.

Exception Messages

CPF0001 – Error in command. Problem will be reported via diagnostic messages.

USM9008 – Not all SMTP sessions were successful.
USM9009 – No SMTP messages sent.

If partner name used:
USM0200 – Partner name not defined.

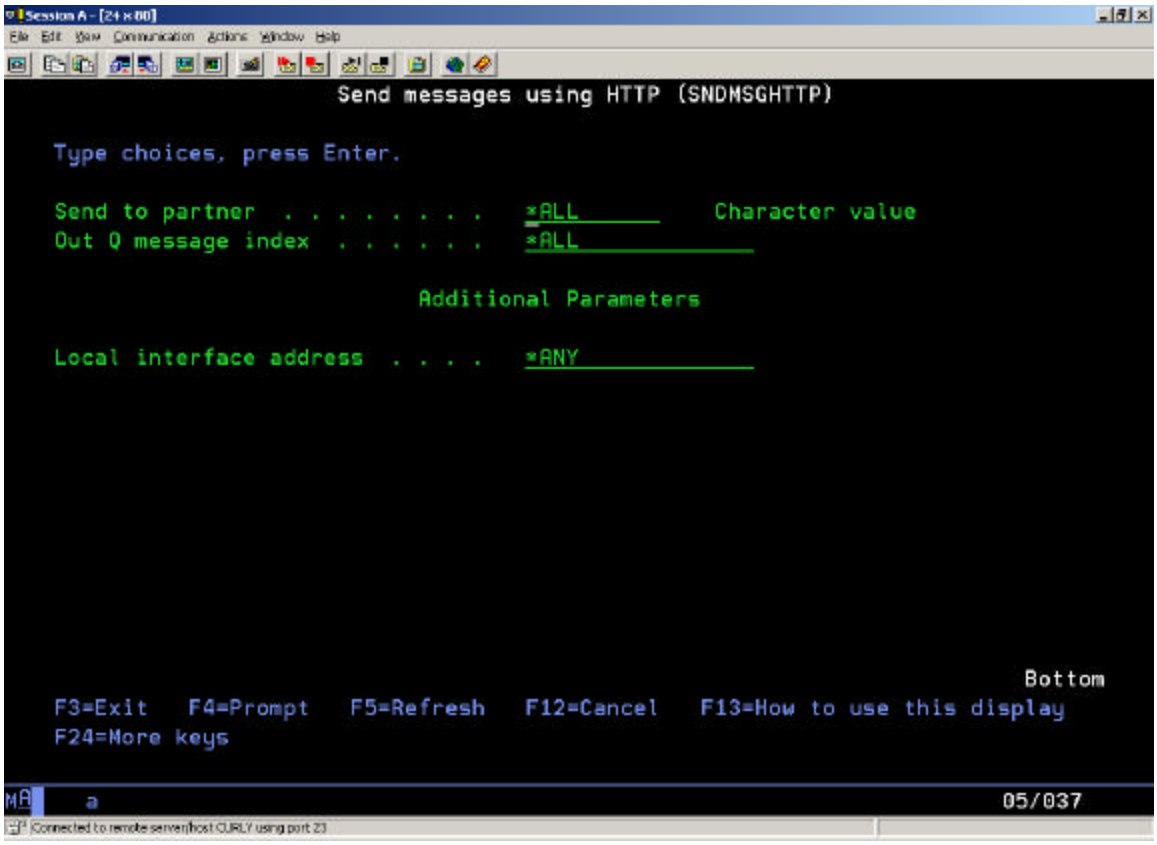
If SMTP server name used:
USM0306 – Server not defined.

If a mail index is used:
USM9202 - Mail index not found.
USM9203 – Mail index is busy.
USM9204 – Mail with this index has wrong protocol for this command.

SNDMSGHTTP – Send AS2 Messages.

This command is used to deliver AS2 messages from the outbound queue to the AS2 server at the URL that's defined for each trading partner. Unless a specific message index is used, only messages in a released status will be attempted. If a message cannot be delivered it will left in a released state for a future retry when this command is run again unless delivery has already been attempted for the maximum number of tries defined for that trading partner. In that case it will be marked as "failed". If the response from the AS2 indicates that the message is faulty and could never be delivered it will be immediately marked as "failed". If the "outbound queue sent API" is defined it will be called after the delivery attempt of each message.

Here are the command parameter definitions and their usage.



Send to Partner: To deliver a single partner's messages at this time enter the trading partners name here.

OutQ message index: Each mail entry on the EDI/IP outbound queue has a unique index number. To send a specific message enter the message index here or choose option 14 from the outbound queue.

Local interface address: If the local iSeries host is multi-homed (i.e. has multiple IP addresses) and you need to pick a specific IP address from the ones available to you for this delivery then enter that IP address here.

Informational Messages

UHT9104 –&1 messages delivered.

Exception Messages

CPF0001 – Error in command. Problem will be reported via diagnostic messages.

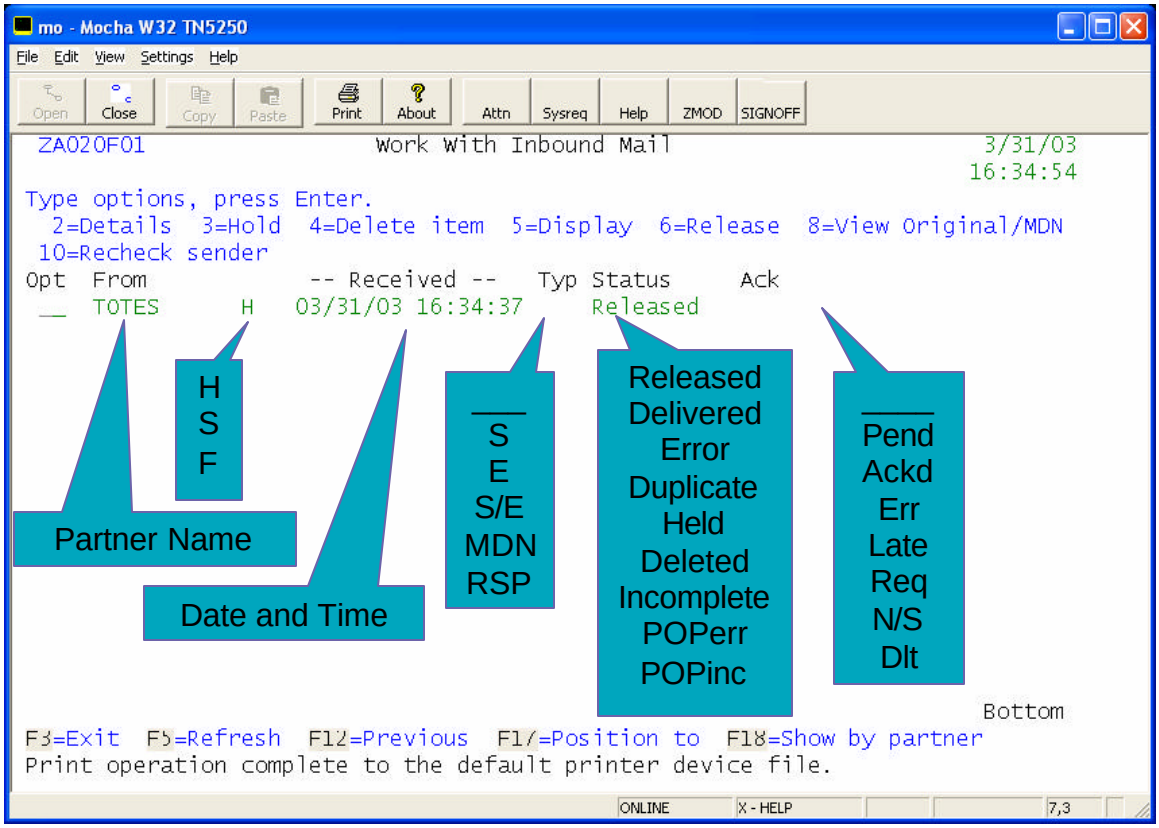
UHT9101 – No AS2 messages sent.
UHT9102 – Attempted to deliver &1 messages. &2 marked undeliverable, &3 not yet sent.
UHT9103 – Attempted delivery of &1 messages. &3 not yet sent. Try later.

If partner name used:
USM0200 – Partner name not defined.

If a mail index is used:
USM9202 - Mail index not found.
USM9203 – Mail index is busy.
USM9204 – Mail with this index has wrong protocol for this command.

ZWRKINMAIL – Work with received mail.

This command is used to show you all received Mail independent of the type of protocol which was used (AS1, AS2, and AS3).



From: The Trading Partner from whom you received the transaction. *UNKNOWN is shown when the partner could not be identified.

H stands for HTTP (AS2)
S stands for SMTP (AS1)
F stands for FTP (AS3)

Type: This column shows you what type of message you received

BLANK	Unknown
S	Signed
E	Encrypted
S/E	Signed and Encrypted
MDN	MDN
RSP	Response.

Status: Different stats are explained below

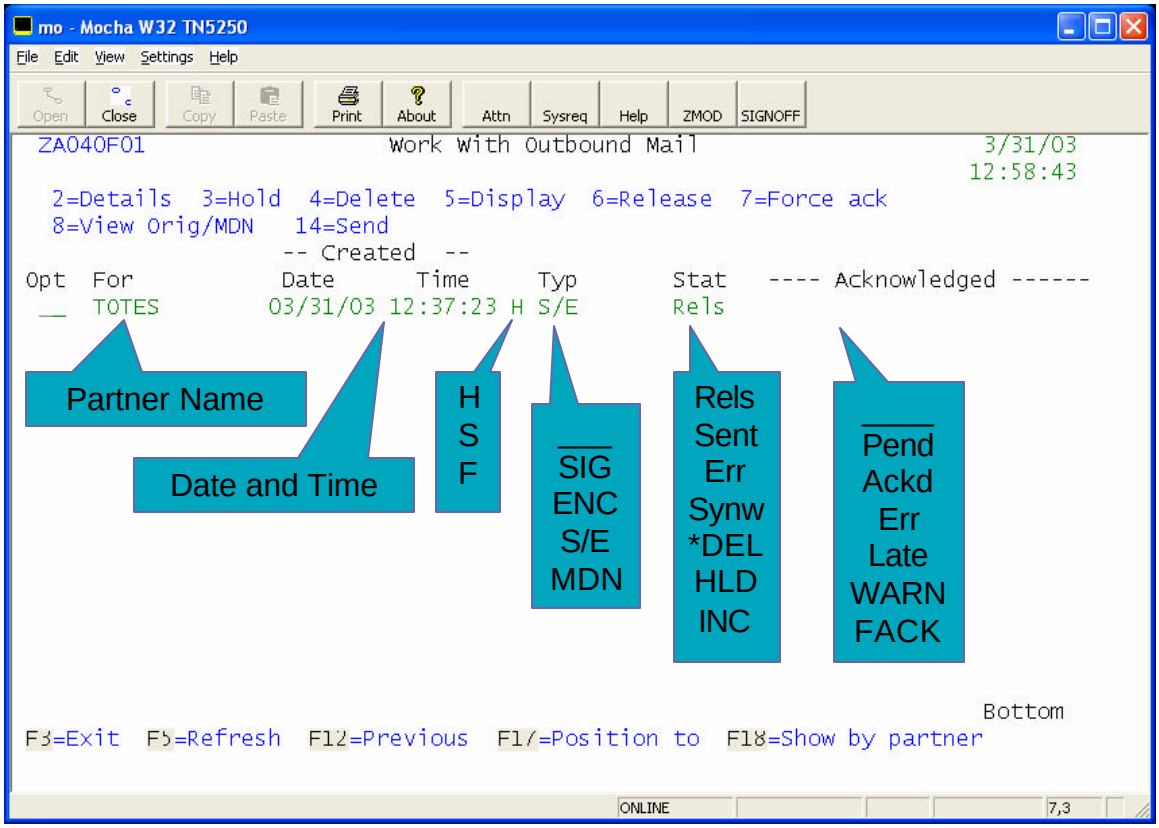
Released	Message was received and is ready to be processed
Delivered	Message was received and processed (decrypted / exported to a file or Translator)
Error	indicates that there is a problem with this transaction
Duplicate	indicates that you received the same message (identical message Identifier) twice
Held	Message is on hold and will not be processed
Deleted	Message was deleted
Incomplete	Message was not completely received or is in the status of being received.

ACK: Acknowledgment status of your received transaction

Pend	Message was received and MDN is pending
Ackd	Message was received and processed and MDN was created
Err	Error with MDN
Late	MDN is late
N/S	MDN is not send
Dlt	MDN is deleted

ZWRKOTMAIL – Work with outbound mail.

This command is used to show you all transactions in your outbound mailbox independent of the type of protocol which was used (AS1, AS2, and AS3).



For: The Trading Partner to whom you will send or sent the transaction.

H stands for HTTP (AS2)
S stands for SMTP (AS1)
F stands for FTP (AS3)

Type: This column shows you what type of message you will send or sent.

SIG	Signed
ENC	Encrypted
S/E	Signed and Encrypted
MDN	MDN

Status: Different stats are explained below

Rels	Message is queued but not send yet.
Sent	Message was sent to your Trading Partner
Err	indicates that there is a problem with this transaction
HLD	Message is on hold and will not be send
*DEL	Message is deleted
INC	Message is not completed, is in the process of being generated.

ACK: Acknowledgment status of your received transaction

Pend	Message was received and MDN is pending.
Ackd	Message was send and MDN was received
Err	Error with MDN
Late	MDN is late

Readers' Comments — We'd Like to Hear from You

Reference Manual
EDI/IP

Overall, how satisfied are you with the information in this book?

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Overall Satisfaction	❑	❑	❑	❑	❑

How satisfied are you that the information in this book is:

	Very Satisfied	Satisfied	Neutral	Dissatisfied	Very Dissatisfied
Accurate	❑	❑	❑	❑	❑
Complete	❑	❑	❑	❑	❑
Easy to find	❑	❑	❑	❑	❑
Easy to understand	❑	❑	❑	❑	❑
Well organized	❑	❑	❑	❑	❑
Applicable to your tasks	❑	❑	❑	❑	❑

Please tell us how we can improve this book:

Thank you for your responses. May we contact you? ❑ Yes ❑ No

When you send comments to nuBridges, you grant nuBridges a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Name _____ Address _____

Company or Organization _____

Phone No. _____