

## Chapter 8a

# Encryption and Advanced Electronic Signature

This chapter contains information about the configuration and the functions of i-effect®'s \*CRYPT module. This chapter is divided into different sections **“Signing/verification of PDFs”**, **“Encryption/Description of Files”**, and **“Qualified Signing of Files”**.

Enter **12** in i-effect®'s main menu to reach the Signature and Encryption tasks. This option displays a menu in which all \*CRYPT module tasks are summarized. These functions can be run from this menu.

### Here are a few key terms:

#### Keystore

A keystore is a protected database which contains keys and certificates.

Access to the keystore is gained by a password, which must be determined by the user at the creation of a new keystore. A password, which is in use, can only be changed, if it has been entered for authentication already.

#### Key

A key is a character string of bits, which is used in cryptography. A key makes encryption, decryption, and other mathematical operations possible.

#### Private/Public Key Pair

A public/private key pair is a mathematical combination of two character strings, which are called “private key” and “public key”. The public key is the member of the key pair that typically is accessible to all partners, who are involved in encrypted communication. The private key is the sensitive part of the key pair and should only be accessible to its owner.

Data, which has been encrypted with a public key, can only be decrypted with the corresponding private key. The reverse is also true. Data that has been encrypted with a public key cannot be decrypted with the same public key.

<b>Private Key</b>	In an asymmetric cryptosystem the private key is a key which is only known to its owner. In a symmetric cryptosystem trusted communication partners also know the private key.
<b>Public Key</b>	In cryptosystems a public key is one that can be known by all and is used to encrypt messages, which are intended for the owner of the corresponding private key.
<b>Symmetrical cryptosystem</b>	A symmetrical cryptosystem is a cryptosystem, which, unlike an asymmetrical cryptosystem, uses the same key for encryption and decryption.
<b>Asymmetrical cryptosystem</b>	An asymmetrical cryptosystem is a system that, unlike a symmetrical cryptosystem, uses different keys for encryption and decryption. These are called the public and private keys.
<b>Certificate</b>	<p>In an asymmetrical cryptosystem the certificate is proof that a public key belongs to a particular person, institution, or machine. The authenticity, confidentiality, and integrity of data can then be guaranteed.</p> <p>A certificate contains information about the name of its owner, the owner's public key, a serial number, the validity, and the name of the Certificate Authority. This data is usually signed with the private key of the Certificate Authority and can be verified with the public key of the Certificate Authority. Certificates for keys that are no longer secure can be blocked over a Certificate Revocation List.</p>
<b>Certification Chain</b>	A certification chain is a list of certificates from the user's certificate to the root certificate of a CA (Certificate Authority). The certification chain can be tested to see if the certificate came from a specific Certificate Authority, which then verifies the user's identity.
<b>Certificate Authority (CA)</b>	A Certificate Authority is an organization that issues certificates. A digital certificate is the electronic equivalent of an ID, and is used to assign a specific public key to a person or organization. The Certificate Authority certifies the assigning of keys by signing them with their own digital signature. The certificate contains "keys" and other information that is used to authenticate as well as to encrypt and decrypt sensitive or confidential documents, which are transmitted over the internet or other networks. Extra information that the CA can add to the certificate are lifespan, references to blocked lists, etc.

## i-effect®'s Standard Keystore

After installation is complete, i-effect®'s standard keystore can be found in the directory `/i-effect/<version>/crypt` under the name certificates. P.12 (VERSION is the version of i-effect® that is currently installed, e.g. v1r4m0)

It is recommended that the password be changed before the initial use of the keystore. The tool "**i-effectKeyManager**" can be used for this purpose. The tool is in the directory `/i-effect/<version>/CRYPT/tools/i-effectKeyManager.jar`.

All functions for importation and exportation of keys, certificates, and additional functions of the keystore are found in the "**i-effectKeyManager**".

The use of the i-effectKeyManager is described in chapter 12 "Additional Graphical Applications"

## Basic Functions of the \*CRYPT Module

The basic configuration of \*CRYPT can be found in chapter 10 "Administration in i-effect®": The sub point "Additional Parameters for the \*CRYPT Module" explains the basic settings for \*CRYPT.

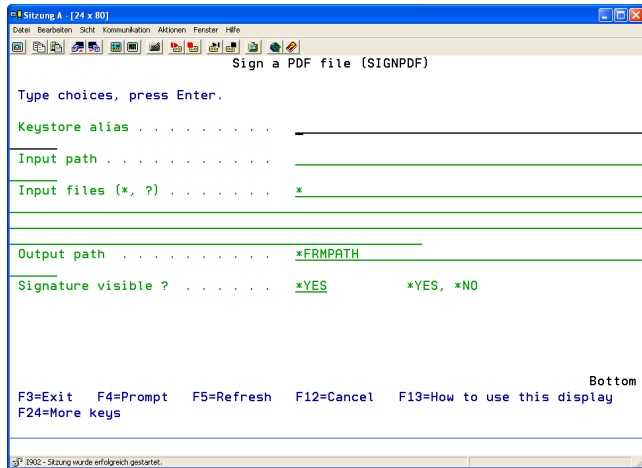
## Sign PDF File (SIGNPDF)

The command **SIGNPDF** is used to sign PDF files. This can be an existing PDF document or it can be a PDF that will be created by i-effect® from an IBM Power Systems spooled file. The command uses the functions of the \*CRYPT module to calculate an advanced electronic signature. The use of this command requires the i-effect® modules \*BASE and \*CRYPT.

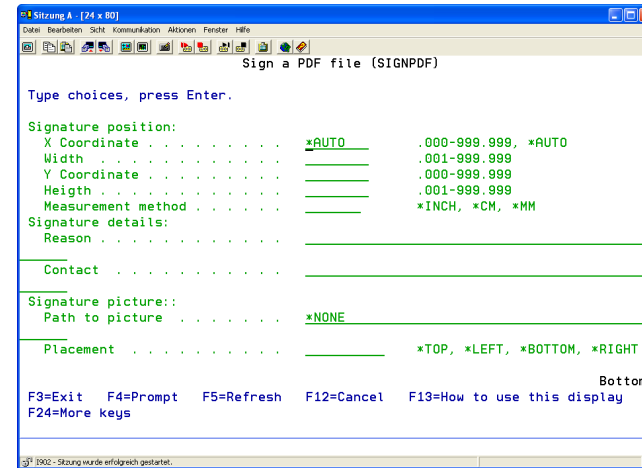
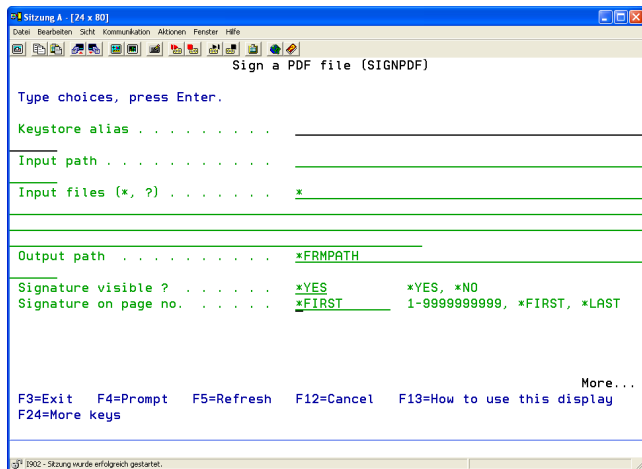
To reach the menu to sign PDF files, choose 12 from the i-effect® main menu and then menu **item 1 "Sign PDF File(s)"**.

The command **SIGNPDF** and **F4** can also be entered directly from the i-effect® menu.

**The following program interface will be displayed:**



The possible parameters for signing PDF files take up two pages. Press **F9** in order to display all possible parameters. To switch between the menu pages use the page up and page down keys.



#### Description of the parameters:

##### Keystore Alias (ALIAS)

The alias in i-effect®'s keystore which is used to create signatures. A private key must be filed in the keystore under the alias.

##### Input Path (FRMPATH)

Enter the path in the IFS file system from which the PDF files will be taken for the creation of a digital signature.

##### Input File(s) (FRMIFSFIL)

Determine the PDF documents which are to be read from FRMPATH and then be used. By entering the wildcards "\*" and "?" any number of files corresponding to the desired search pattern can be controlled.

*	All files of the chosen directory will be edited.
<b>generic*</b>	Part of the file name can be substituted with "*". If the first part of the name is entered followed by "*", then all files whose name begins with the desired part will be listed.

#### The following formats are possible for generic names:

<b>ABC*</b>	All files having the characters ABC at the beginning will be used, e.g. ABC, ABCD, ABCTEST.
-------------	---

<b>a*</b>	All files whose name is in quotation marks and begins with a will be used, e.g. "a," "aB," and "aD."
<b>*.pdf</b>	All files having the suffix ".pdf" will be used.

### Output Path (TOPATH)

Name of the output path for the signed PDF documents. The file name corresponds to the original file name from the elements of the input path. If files with the same name already exist in the target directory, then the original names will be kept and an add-on (number) will be inserted.

### Visible Signature (VISIBLE)

The embedded signature can be visualized as a picture or a signature graphic. With this parameter it can be determined if a visual graphic will be embedded in the PDF file.

<b>*YES</b>	The signature will be visualized in the PDF document.
<b>*NO</b>	The PDF document contains a signature, but it is not visualized as a graphic.

### Signature on Page No. (SIGPAGE)

If a signature graphic is to be embedded, this parameter determines on which page it will appear.

<b>*FIRST</b>	The signature graphic will appear on the first page of the PDF document.
<b>*LAST</b>	The signature graphic will appear on the last page of the PDF document.
<b>1-999999999</b>	The signature graphic will appear on the designated page.

### (POSITION)

The POSITION parameter (Signature position) will appear if VISIBLE (\*YES) is chosen.

The parameter allows the positioning of the signature's visualization in the PDF document. Coordinates must be entered from top to bottom and from left to right to specify where on the page the visualization will appear.

#### Example:

```
SIGNPDF      ALIAS(,as2.menten.com')
             FRMPATH(/tmp')
             FRMIFSFIL(*.pdf')
             TOPATH(*FRMPATH)
             VISIBLE(*YES)
             SIGPAGE(*FIRST)
             POSITION(0 100 0 100 *MM)
```

With the certificate from "as2.menten.com", this will create a digital signature for all PDF files in the directory "/tmp". A visual signature graphic 110x100 mm will appear on the first page in the upper left hand corner.

There are five elements of this parameter.

#### X-Coordinate

The first element is the column position or x-coordinate, i.e. the position from the left margin where the signature graphic will be positioned. Whether measurements in inches or millimeters will be used, depending on the value of the measurement method option below (see element 5).

#### Width

The second element is the width of the visual signature area. Whether measurements in inches or millimeters will be used, depending on the value of the measurement method option below (see element 5).

#### Y-Coordinate

The third element is the line number or the y-coordinate, i.e. the position from the top of the page where the signature graphic will be positioned. Whether measurements in inches or millimeters will be used, depending on the value of the measurement method option below (see element 5).

#### Height

The fourth element is the height of the visual signature area. Whether measurements in inches or millimeters will be used, depending on the value of the measurement method option below (see element 5).

#### Measurement Method

The fifth element is the measurement system, which is to be used. It determines the units in which the four previous elements will be rendered

#### Possible Special Values:

<b>*INCH</b>	The vertical and horizontal position, as well as length and breadth will be specified in inches.
<b>*MM</b>	The vertical and horizontal position, as well as length and breadth will be specified in millimeters.
<b>*CM</b>	The vertical and horizontal position, as well as length and breadth will be specified in centimeters.

## Signature Details (DETAILS)

Along with the signature other details can be specified, which will be displayed to the receiver when the file is opened.

This parameter consists of two elements:

### Reason

The first element allows the indication of a reason for signature creation. This can contain any order of characters and will be displayed in the field "Reason."

### Contact

With the second element contact information can be entered. This can contain any order of characters and will be displayed in the field "Contact."

## Signature Graphic (PICTURE)

Along with the signature a graphic can be embedded in the PDF file that will be displayed to the recipient. This graphic can be a personal signature or a picture.

Possible Special Values:

**\*NONE** No graphic file will be embedded.

The parameter consists of two elements:

### Path of the Image File

Enter the complete IFS path of the image file.

### Alignment

Determine the area in which the image should appear in the signature field.

Possible Special Values:

<b>*TOP</b>	The image appears at the top of the signature field (the signature will be underneath it).
<b>*LEFT</b>	The image will appear on the left side of the signature field (the signature will be on the right).
<b>*BOTTOM</b>	The image will appear at the bottom of the signature field (the signature will be above it).
<b>*RIGHT</b>	The image will appear on the right side of the signature field (the signature will be on the left).

## Verify PDF Signature (VERIFYPDF)

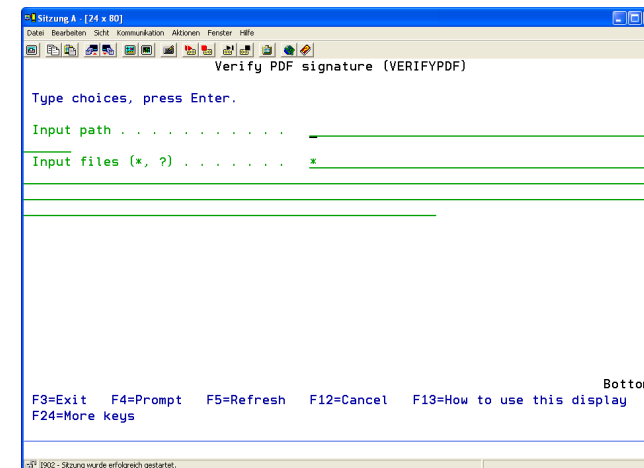
The command **VERIFYPDF** verifies the signed PDF files. The electronic signature of the PDF file will be checked against the public key in the i-effect® keystore. If the entry is present in the keystore, the signature is verified, and integrity and authenticity of the document are guaranteed.

In order to use this command, licenses for the i-effect® modules \*BASE and \*CRYPT are required.

The menu for PDF file verification can be found by entering 12 in the i-effect® menu and then menu item 2 "Verify PDF File(s)."

The command VERIFYPDF and F4 can also be entered directly from the i-effect® menu, if desired.

The following program interface will be displayed:



Description of the parameters:

### Input Path (FRMPATH)

Determine the path in the IFS file system from which the PDF files should be taken in order to verify the digital signatures.

### Input File(s) (FRMIFSFILE)

Determine the PDF documents which are to be read from FRMPATH and then be used. By entering the wildcards "\*" and "?" any number of files corresponding to the desired search pattern can be controlled.

**\*** All files of the chosen directory will be edited.

**generic\*** Part of the file's name can be substituted with "\*". If the first part of the name is entered followed by "\*", then all file names beginning with the desired part will be listed.

The following formats are possible for generic names:

**ABC\*** All files having the characters ABC at the beginning will be used, e.g. ABC, ABCD, ABCTEST.

**a\*** All files whose name is in quotation marks and begins with a will be used, e.g. "a," "aB," and "aD."

**\*.pdf** All files having the suffix ".pdf" will be used.

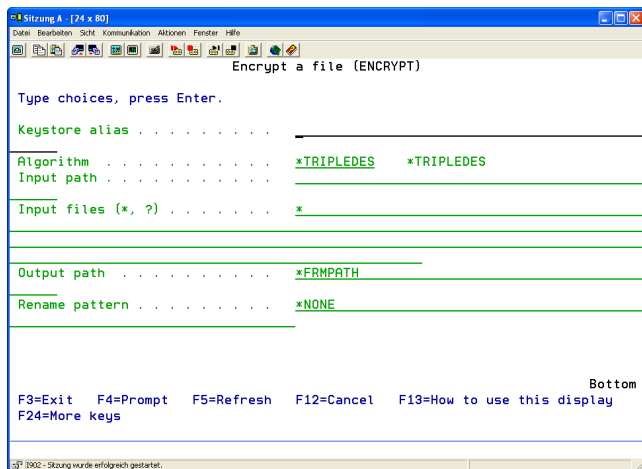
## Encrypt Files (ENCRYPT)

The command **ENCRPT** is used to encrypt files. The i-effect®\*CRYPT functions encrypt files by using a PKI infrastructure, which results in an output file in the standard PK7 format. On the one hand, this can be decrypted from any PKI system, and on the other hand, it presents a standardized way of displaying encrypted contents. To use this function, it is required that a public key exists in the recipient's i-effect® keystore, and that the alias of the key entry is known. Furthermore, the i-effect® modules \*BASE and \*CRYPT are required in order to use this command.

To reach the encryption menu, enter **12** from the i-effect® menu and then **menu item 4 "Encrypt Files"**.

The command **ENCRPT** and **F4** can also be entered from the i-effect® menu directly.

The following program interface will be displayed:



Description of the parameters:

### Keystore Alias (ALIAS)

The alias in i-effect®'s keystore, which will be used to encrypt the data of the input file. A public key (certificate) must be filed in the keystore under the alias.

### Encryption Algorithm (ENCRYPT)

This parameter determines the type of encryption to be used for the specified input file.

\*TRIPLEDES 3DES encryption

The Data Encryption Standard (DES) is a widespread symmetric encryption algorithm with a key length of 3DES (=168 bits), which is three times as much as with DES encryption (=56 bits). The key's complexity is increased by a factor of  $2^{112}$ .

### Input Path (FRMPATH)

Determine the path in the IFS files system in which the files that are to be encrypted are stored.

### Input File(s) (FRMIFSFIL)

Determine the PDF documents which are to be read from FRMPATH and then be used. By entering the wildcards "\*" and "?" any number of files corresponding to the desired search pattern can be controlled.

**\*** All files of the chosen directory will be selected and processed.

**generic\*** Part of the file's name can be substituted with "\*". If the first part of the name is entered followed by "\*", then all file names beginning with the desired part will be listed.

The following formats are possible for generic names:

**ABC\*** All files having the characters ABC at the beginning will be used, e.g. ABC, ABCD, ABCTEST.

**a\*** All files whose name is in quotation marks and begins with a will be used, e.g. "a," "aB," and "aD."

**\*.pdf** All files having the suffix ".pdf" will be used.

### Output Path (TOPATH)

Determine the output path for the encrypted files. The file name corresponds to the original file name from the elements of the input path. In addition to the name, the suf-

fix .pk7 will be added. If files with the same name already exist in the target directory, the original name will be kept and an add-on (number) will be inserted.

### Rename File as (RENAME)

If it is desired that the encrypted files be renamed or given a new name or name pattern, this can be indicated here. With this parameter the standard settings for naming encrypted files can be changed.

The following option is possible:

**\*NONE** The encrypted files will not be renamed. The name will be created from the original file's name plus the new suffix (pk7).

### Name

A new name or name pattern can be entered here. The name pattern contains the wildcard "\*" for the original file name IN FRONT OF the suffix and a second "\*" for the suffix in the original file name.

E.g.:

**\*.DONE** Converts "file1.txt" to "file1.DONE"  
**\*\_DONE.\*** Converts "file1.txt" to "file1\_DONE.txt"  
**\*\_1055am** Converts "file1.txt" to "file1.txt\_1055am"

## Decrypt Files (DECRYPT)

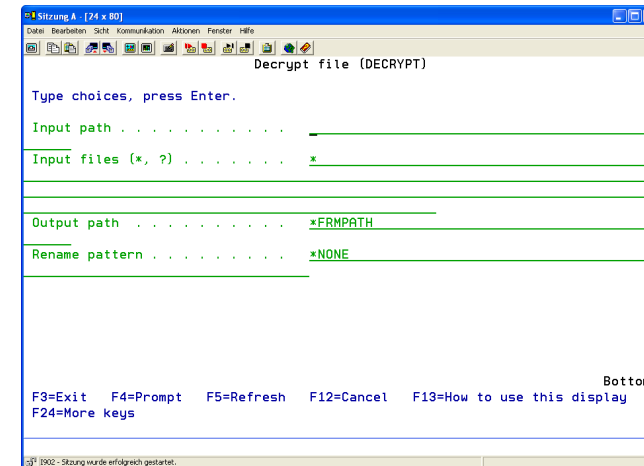
The command **DECRYPT** is used to decrypt files. The i-effect® \*CRYPT functions decrypt files by using a PKI infrastructure. The result of this process is a file that, as long as the corresponding key is present in the keystore, contains the original data.

To use the functions, it is required that the private key, which corresponds to the public key used to encrypt, exists in the i-effect® keystore. Licenses for the modules \*BASE and \*CRPT are required to use this command.

To reach the encryption menu enter **12** and then **menu item 5 "Decrypt File(s)"** from i-effect®'s main menu.

The command **DECRYPT** followed by **F4** can also be entered directly from the i-effect® menu.

The following program interface will be displayed:



Description of the parameters:

### Input Path (FRMPATH)

Determine the path in the IFS files system in which the files to be decrypted are stored.

### Input File(s) (FRMIFSFIL)

Determine the PDF documents which are to be read from FRMPATH and then be used. By entering the wildcards "\*" and "?" any number of files corresponding to the desired search pattern can be controlled.

<b>*</b>	All files of the chosen directory will be selected and processed.
<b>generic*</b>	Part of the file's name can be substituted with "*" If the first part of the name is entered followed by "*", then all file names beginning with the desired part will be listed.

The following formats are possible for generic names:

<b>ABC*</b>	All files having the characters ABC at the beginning will be used, e.g. ABC, ABCD, ABCTEST.
<b>a*</b>	All files whose name is in quotation marks and begins with a will be used, e.g. "a," "aB," and "aD."
<b>*.pdf</b>	All files having the suffix ".pdf" will be used.

### Output Path (TOPATH)

Determine the output path for the decrypted files. The file name corresponds to the original file name from the input path. The suffix "pk7", if present, will be removed. If files with the same name already exist in the target directory, then the name will be kept and an add-on (number) will be inserted.

### Rename File as (RENAME)

If it is desired that the encrypted files be renamed or given a new name or name pattern, this can be indicated here. With this parameter the standard settings for naming encrypted files can be changed.

The following option is possible:

<b>*NONE</b>	The decrypted files will not be renamed. The name will be created from the original file's name. If the suffix "pk7" is present, it will be removed.
--------------	--

#### Name

A new name or name pattern can be entered here. The name pattern contains the wildcard "\*" for the original file name IN FRONT OF the suffix and a second "\*" for the suffix in the original file name.

E.g.:

<b>*.DONE</b>	Converts "file1.txt" to "file1.DONE"
<b>*_DONE.*</b>	Converts "file1.txt" to "file1_DONE.txt"
<b>*_1055am</b>	Converts "file1.txt" to "file1.txt_1055am"

## General Commands & Tools

This section explains commands and tools that are available in the \*CRYPT module.

### Keystore Tools

Keystore tools are supplemental JAVA programs for the i-effect® keystore. They are located in `/i-effect/<version>/CRYPT/tools` under the name `KeystoreTools.jar`. JAR files are JAVA archives that contain JAVA programs.

Keystore tools can be run with the command `RUNJAVA` and the required parameters for the desired functionality. The functions are explained in the following, as well as the parameters used by them.

Call up of `KeystoreTools` looks as follows:

**`RUNJAVA CLASS(/i-effect/<version>/CRYPT/tools/KeystoreTools.jar) PARM(...)`**

This is followed by one or more parameters that are entered within the `PARAM` parameter followed by a comma.

#### Please Note:

The parameters within the `PARAM` parameter must be entered in simple single quotes (`param1, param2,....`)

### \*CHECK – Tests the Validity of Certificates in the i-effect® Keystore

If the `*CHECK` parameter is entered followed by a number (both separated by a comma) the length of validity of all the certificates in the keystore will be tested. The number corresponds to the number of days that will be tested until a certificate loses its validity.

The results of this test are saved in the file:

**`/i-effect/<version>/internal/YYYY-MM-DD-certificates_to_check.list`**

All relevant certificates are listed with their aliases and certificate information in this file.

If all certificates that will lose their validity in the next 30 days should be listed, the command will look as follows:

**`RUNJAVA CLASS(/i-effect/<version>/CRYPT/tools/KeystoreTools.jar) PARM(*CHECK,30)`**

In addition, all certificates will be listed automatically that are not yet valid or have already lost their validity. If no certificates were found that are no longer or not yet valid, the output file will be empty.

## Tip for the Automated Use of i-effect® \*SERVER

KeystoreTools \*CHECK can be automated, for example as a weekly **\*SCHEDULE** sever task and used to react promptly when a certificate or partner's certificate expires. The type of \*SEVER processing used here is **\*USERDEFINED**, and \*NONE should be entered as file type.



*A valid license for the \*SEVER module is required for these tasks.*

One further parameter must also be sent to the command when using server task: the \*SEVER specific variable "%SESSIONNUMBER%". This variable contains the session number used during the run time from the sever task. If this number is sent to KeystoreTools \*CHECK, logbook messages from KeystoreTools will be written with this number in the session.

Using the previous example, the command in the server task looks as follows:

```
RUNJVA CLASS(./i-effect/<version>/CRYPT/tools/KeystoreTools.jar) PARM(*CHECK, 30,%SESSIONNUMBER%)
```

This command is entered as a process to be carried out.

A detailed description of server tasks and their processing can be found in Chapter 8 "Process Automation".