

Use iPSRA To Add Compression and Encryption to iSeries Saves and Restores

The iSeries PKWARE Save Restore Application (iPSRA) feature integrates *PKZIP/SecureZIP for iSeries* compression and encryption technology with IBM iSeries Save and Restore APIs.. This guide explains how to use the iPSRA feature to improve efficiency and security when doing iSeries backups.

About *PKZIP* and *SecureZIP for iSeries*

PKZIP for iSeries provides powerful, easy-to-use data compression on the AS/400, iSeries and i5. *PKZIP for iSeries* Enterprise Edition adds password-based decryption of encrypted files, powered by RSA® BSAFE. Files created by *PKZIP for iSeries* use the widely adopted ZIP format and can be accessed on all major platforms, from iSeries to Microsoft® Windows.

SecureZIP for iSeries adds data protection to the feature set for *PKZIP for iSeries*. *SecureZIP for iSeries* delivers the same high-performance data compression as *PKZIP for iSeries* and additionally protects data with digital signatures and RSA BSAFE encryption, either password- or certificate-based. Like *PKZIP for iSeries*, *SecureZIP for iSeries* uses the ZIP format and creates files that can be accessed on all major platforms.

The compression/decompression and encryption/decryption functionality of *PKZIP/SecureZIP for iSeries* is controlled by two main commands: PKZIP and PKUNZIP. The PKZIP command controls compression and encryption and creates or updates ZIP archive files. The PKUNZIP command decrypts and decompresses files in an archive to restore them to their original form. Both commands offer a wealth of options.

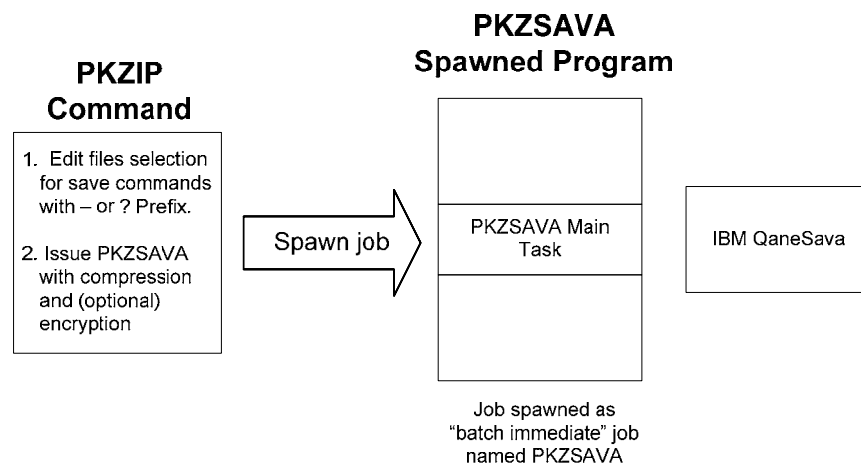
This guide describes how the iPSRA feature works with the PKZIP command to save objects and with the PKUNZIP command to restore them.

Save/Restore Command Overview

The iPSRA feature supports the following Save command types:

- Save (SAV) command
- Save Object (SAVOBJ) command
- Save Document Library Object (SAVDLO) command
- Save Library (SAVLIB) command

With the iPSRA feature, PKZIP spawns a batch immediate program named PKZSAVA that processes save command data and causes it to be compressed—and optionally encrypted—into an archive instead of being saved directly to disk. PKZSAVA uses the IBM API, which is a pre-started job in the QSYSWRK subsystem. The figure below gives an overview of the process.



Saving Data

To use the iPSRA feature to save data, you enter command strings for SAV, SAVOBJ, and/or SAVLIB in the FILES parameter of PKZIP. Prefix each command string with a hyphen (-) or question mark (?).

- A '-' tells PKZIP that the entire command is entered and is ready to be validated
- A '?' causes PKZIP to prompt for command parameters to enter before validating command syntax

The following example shows a PKZIP command to save the library MYLIB and some other objects:

➔ **PKZIP ARCHIVE('/MYpath/myarchive') FILES('-SAVLIB LIB(MYLIB) DEV(MYNAME1)' '?SAVOBJ DEV(MYNAME2)')**

The command line above creates a saved library file or iPSRA file of the library MYLIB and compresses it into the archive /MYpath/myarchive as file SAVLIB01_MYNAME1. The command line uses the '?' prefix with the command SAVOBJ to prompt for the objects to be saved. These must be specified at the prompt. The SAVOBJ command creates an iPSRA file named SAVOBJ02_MYNAME2 in the archive.

Restoring Data

The restore of a save object with a RSTLIB command works similarly except that only one file can be restored at a time to assure proper building of the saved objects.

Examples are given in the final section.

Syntax

Save command parameters must be consistent with the save command type entered and must be separated by at least one blank character. Refer to the Control Language (CL) documentation for detailed information about valid parameters when you save objects to save files.

File Names Used for Saved Data

The file name PKZIP uses for saved data in an archive is based on the type of save command and the name used in the save command DEV parameter. If the DEV parameter is *SAVF, then the name comes from the SAVF parameter. See the table below for examples.

Command	File Name
SAVOBJ DEV('MYPAYROLL')	SAVOBJnn_ MYPAYROLL
SAVLIB DEV(*SAVF)	SAVF(MYLIB/MYSAVF)
SAV DEV('MYSAVFDEVPATH')	SAVnn_ MYSAVFDEVPATH

The *nn* in the example file names is the sequence number of the command for one PKZIP run.

Be sure to avoid duplicate file names when updating an existing archive with more save commands. PKZIP removes existing files that have the same name as a new save file.

Note: When updating an existing archive, PKZIP checks to see if there are any iPSRA files in the archive. If there are, PKZIP uses the largest *nn* number plus one as the starting *nn* for iPSRA files to be added. This helps avoid any accidental loss of iPSRA files in an updated archive because of duplicate file names.

Extended Data in Archive

Each save operation creates specific data in the extended data area to tell PKUNZIP that a file to be restored is a file of the special SAVE type. The extended data also provides history information that can be displayed with VIEWOPT(*ALL) to show the original command, the job when the save data was created, the target release used for the save, and the spawned job PKZSAVA.

```

Filename: SAVLIB01_DEDSAV01
Detected File type:      SAVE Apps. Data
Created by:              PKZIP for iSeries(tm)  8.2
Zip Spec to Extract:    2.0 Or Greater
Compression method:     Deflated [Superfast]
Date and Time           2005 Aug 8 14:08:58
Compressed size:        42876 bytes
Uncompressed size:     413808 bytes
32-bit CRC value (hex): 334d1674
Extended attributes:    yes, [Length = 134]
File Save Apps Data:<savlib lib(deD) dev(dedsav01) OUTPUT(*OUTFILE)
      OUTFILE(WSS/TEM01)>
      :TGTRLS(*Current) Save Job <019627/EVWSS/PKZSAVA>
File Comment:"none"
-----
Found 1 file, 413808 bytes uncompressed, 42876 bytes compressed:   90%
SecureUNZIP extracted    0 files
SecureUNZIP Completed Successfully
    
```

```

Additional Message Information

Message ID . . . . . : AQZ0895
Date sent . . . . . : 08/08/05      Time sent . . . . . : 14:09:20

Message . . . . . : File Save Apps Data:<savlib lib(deD) dev(dedsav01)
      OUTPUT(*OUTFILE) OUTFILE(WSS/TEM01)>

This shows that the file is a Save Application data file type 4 and the
command that was used to store the save file in the archive. Save Data
created with PKZIP job 019586/EVWSS/EVWSSL01. Target System was *Current.
    
```

Notice in the output above that the spawned job (019627/EVWSS/PKZSAVA) was captured as well as the PKZIP job (019586/EVWSS/EVWSSL01).

Information on the spawned job may be needed to do a DSPLOG command.

The target release (TGTRLS) is also shown to note the target setting for the run.

Notes and Restrictions

All IBM restrictions and security requirements apply to the use of the Save and Restore commands in iPSRA. Some additional restrictions are noted below.

PKZIP

- QTEMP cannot be specified for the library name on the OUTFILE or SAVACTMSGQ parameters.
- Some parameters of the save commands that are not used by iPSRA are ignored. For example, CLEAR, DTACPK, and so on.
- Objects saved by PKZIP can only be restored using the restore from application with PKUNZIP, and they can only be restored to a release of the operating system that is the same or later than the version from which the objects were saved.
- The save parameters are only completely validated when PKZIP submits the save command for processing.
- A target release VxRxMx value prior to V5R1M0 is not valid: The iPSRA feature is not supported prior to version 5, release 1, modification level 0 of PKZIP. The version, release, and modification level depend on the save operation being performed. See the valid values for the TGTRLS parameter table in the *iSeries Backup and Recovery Manual* for a complete list of valid values.
- All compression methods except the Terse compression method are supported with iPSRA.
- Positional options on the save commands must contain the parameter. For example, the save library command

```
SAVLIB WSS DEV(*SAVF) SAVF(TESTWSS)
```

would not work to save the library WSS. The WSS is a positional parameter, where it is assumed WSS was the LIB option. The correct approach with iPSRA is to use the command:

```
-SAVLIB LIB(WSS) DEV(*SAVF) SAVF(TESTWSS)
```

- The save operation must be completely successful or it aborts. If any object is not saved for any reason, the PKZIP job assumes a failure. Do not include an object that will not save, as this object will cause a major failure.

PKUNZIP

- QTEMP cannot be specified for the library name on the OUTFILE parameter.
- To ensure that all objects are properly restored, only one restore command can be processed per run.
- Some parameters of the restore commands are not used by iPSRA and are ignored. For example, VOL, SEQNBR, and so on.
- The user must have the required security for the restore command.

Using OUTPUT and OUTFILE with the Save Commands

A save command can have an OUTFILE parameter that is used to build a file listing the objects saved with that command. When an OUTFILE is specified for a save command, PKZIP automatically archives the outfile in the same archive as the iPSRA file with the name specified. The outfile provides a record of the contents of an iPSRA file. An outfile has the format and restrictions defined by IBM for the save commands. Use of outfiles is optional.

If OUTPUT(*PRINT) or OUTFILE(*PRINT) is used with a save or restore command, the printout is produced by the IBM API job and not with the PKZIP job nor the spawned PKZSAVA job. Therefore, it appears in the special OUTQ job named QPRTJOB for each user.

How to Use the Save Application Feature

The save option is activated by entering a save command in the FILES parameter of PKZIP, prefixed with a hyphen (-) or question mark (?). Multiple save commands can be entered to select multiple sets of files in the same pass. The save commands do not need to be the same, nor do they need to use the same prefix.

If a command fails the pre-command processor, PKZIP issues the message AQQ0332, which shows the failed command. The reason for the failure appears in the job log prior to this message.

If a failure occurs during the processing of the save commands, the reason for the failure appears in the job log of the spawned job. If any errors occur in the spawned job, a job log will be forced. There is no pre-check processing on security or on the objects themselves. All data verification is handled by the save API.

For example, the following command tries to save a library (NOLIB) that does not exist:

➔ PKZIP ARCHIVE('/yourpath/BILLS/X5TESTL.ZIP') TYPARCHFL(*IFS) FILES('-SAVLIB LIB(NOLIB) DEV(TESTSAV01) OUTPUT(*PRINT)')

The log of the PKZIP output might look like this:

```
Scanning files in *DB for match ...
Found 0 matching files
1 Save Command(s) selected
Command:<SAVLIB LIB(NOLIB) OUTPUT(*PRINT)>
Compressing SAVLIB01_TESTSAV01 in SAVE Apps. Data mode
Save Operation encountered an error. See Job Log of PKZSAVA save job for further details.
iPSRA Initialization Failure has occurred
iPSRA Failed. Save command not successful.
SecureZIP Copied 1 files from input archive
SecureZIP Compressed 0 files in Archive /yourpath/BILLS/X5TESTL.ZIP
SecureZIP Completed with Errors
Press ENTER to end terminal session.
```

The job log of the PKSAVF output might look like this:

```
CPF3781 Diagnostic 30 08/08/05 14:49:10.428528 QANESERV QSYS
From module . . . . . : QANESERV
From procedure . . . . . : QaneSendPgmMsg_FP14qanec_CTLBLK_tPcT2iN24
Statement . . . . . : 19
To module . . . . . : QP0ZPCPN
To procedure . . . . . : InvokeTargetPgm_FP11qp0z_pcp_cb
Statement . . . . . : 187
Message . . . . . : Library NOLIB not found.
Cause . . . . . : The library specified for the save or restore
operation does not exist on the system. Recovery . . . : Do one of the
following and try the request again: If this is a save operation, correct
the library name on the LIB parameter. If this is a restore operation,
correct the library name specified on the SAVLIB or RSTLIB parameter, or
use the Create Library (CRTLIB) command to create the library by
specifying LIB(NOLIB). If this is a restore operation and VOL(*SAVVOL)
was specified, the save library must exist in the auxiliary storage pool
specified on the RSTASPDEV parameter. If RSTASPDEV(*SAVASPDEV) and
RSTASP(*SAVASP) are specified along with VOL(*SAVVOL), then the save
library must exist in the system ASP. To restore a library that is new to
the system, specify VOL(*MOUNTED) instead of VOL(*SAVVOL).
```

How to Use the Restore Application Feature

To restore an iPSRA file from archive, code the restore command in the RSTIPSRA parameter of PKUNZIP. The RSTIPSRA parameter is defined as a command entry, so there are no quotes around the command. Enclose the entire restore command in parentheses: RSTIPSRA (*command*). To be prompted at the command, place the cursor on the RSTIPSRA entry and press the F4 key.

If an archive contains more than one file, you must use the FILES parameter to select the file you want to match up with the RSTIPSRA parameter. PKUNZIP restores only one iPSRA file per run.

If any object is not restored, PKUNZIP issues the message AQZ1007 and creates a job log for the PKZRSTA job. You should review the log to find any object that was not restored and the reason.

If a partial restore is performed, then the CRC and/or hash calculation for authentication does not take place, and the warning message AQZ1000 is displayed. This situation can arise if the save operation was a SAVLIB, but the restore operation restores only a few objects with the RSTOBJ.

Database Considerations for Save and Restore

Below are some tips for working with the save and restore functions.

- When you save an object to a save file or using iPSRA, you can prevent the system from updating the date and time of the save operation by specifying UPDHST (*NO) on the save command.
- When you restore an object, the system always updates the object description with the date and time of the restore operation. Display the object description and other save/restore related information by using the Display Object Description (DSPOBJD) command with DETAIL (*FULL).
- To display the last save/restore date for a database file, type: DSPFD FILE (*filename*) TYPE (*MBR).

Sample Jobs

iPSRA Example 1

The following example saves the library DED and prints the output of the save. It also saves the file object TESTFILE from the library TESTLIB with

several options of the SAVOBJ. These save application files are compressed with a default setting and will be encrypted using a password.

```
➔ PKZIP ARCHIVE('/yourpath/bills/testsavx1.zip') TYPARCHFL(*IFS)
FILES(
'-SAVLIB LIB(DED) DEV(DED SAV01) OUTPUT(*PRINT) '
'-SAVOBJ OBJ(TESTFILE) LIB(TESTLIB) DEV(TESTOBJ11)
OBJTYPE(*FILE) TGTRLS(V5R1M0) UPDHST(*NO)
PRECHK(*YES) OUTPUT(*PRINT) ')
PASSWORD('bills00000') VPASSWORD('bills00000')
```

```
SecureZIP(TM) for iSeries is running under Beta release B0
Machine ID = 01045B5C, Processor Group = P05
Scanning files in *DB for match ...
Found 0 matching files
2 Save Command(s) selected
Command:<SAVLIB LIB(DED) OUTPUT(*PRINT)>
Compressing SAVLIB01_DEDSAV01 in SAVE Apps. Data mode
Add SAVLIB01_DEDSAV01 -- Deflating (90%) encrypt(BSAFE AES 256Key)
Command:<SAVOBJ OBJ(TESTFILE) LIB(TESTLIB) OBJTYPE(*FILE)
UPDHST(*NO) PRECHK(*YES) OUTPUT(*PRINT)>
Compressing SAVOBJ02_TESTOBJ11 in SAVE Apps. Data mode
Add SAVOBJ02_TESTOBJ11 -- Deflating (79%) encrypt(BSAFE AES 256Key)
SecureZIP Compressed 2 files in Archive /yourpath/bills/testsavx1.zip
SecureZIP Completed Successfully
```

IPSRA Example 2

The following commands display the contents of the archive:

```
➔ PKUNZIP ARCHIVE('/yourpath/bills/testsavx1.zip') TYPARCHFL(*IFS)
TYPE(*VIEW)
```

```
Archive: /yourpath/bills/testsavx1.zip, 1358415 bytes, 2 files, 1 Segment
```

Length	Method	Size	Ratio	Date	Time	CRC-32	Name
430192	Defl:S	43718	90%	08-09-05	08:16	ac1f8407	!SAVLIB01_DEDSAV01
6325776	Defl:S	1313814	79%	08-09-05	08:16	101311d4	!SAVOBJ02_TESTOBJ11
6755968		1357532	80%				2 files

```
➔ PKUNZIP ARCHIVE('/yourpath/bills/testsavx1.zip') TYPARCHFL(*IFS)
TYPE(*VIEW) VIEWOPT(*ALL)
```

```
Archive Comment:"SecureZIP for iSeries"
Filename: SAVLIB01_DEDSAV01
Detected File type: SAVE Apps. Data Encrypt=Strong Encrypted
Created by: PKZIP for iSeries(tm) 8.2
Zip Spec to Extract: 5.1 Or Greater
Compression method: Deflated [Superfast]
Date and Time: 2005 Aug 9 08:16:16
Compressed size: 43718 bytes
Uncompressed size: 430192 bytes
32-bit CRC value (hex): ac1f8407
Extended attributes: yes, [Length = 130]
Strong Encryption AES 256 Key (BSAFE).
Algorithm Key 256, Security type Password
Number Certificate Recipients 0
```

```

Recipient List:
File Save Apps Data:<SAVLIB LIB(DED) DEV(EDSAV01) OUTPUT(*PRINT)>
: TGTRLS(*Current) Save Job <019700/EVWSS/PKZSAVA>
File Comment:"none"
-----
Filename: SAVOBJ02_TESTOBJ11
Detected File type: SAVE Apps. Data Encrypt=Strong Encrypted
Created by: PKZIP for iSeries(tm) 8.2
Zip Spec to Extract: 5.1 Or Greater
Compression method: Deflated [Superfast]
Date and Time 2005 Aug 9 08:16:16
Compressed size: 1313814 bytes
Uncompressed size: 6325776 bytes
32-bit CRC value (hex): 101311d4
Extended attributes: yes, [Length = 217]
Strong Encryption AES 256 Key (BSAFE).
Algorithm Key 256, Security type Password
Number Certificate Recipients 0
Recipient List:
File Save Apps Data:<SAVOBJ OBJ(TESTFILE) LIB(TESTLIB) DEV(TESTOBJ11) OBJTYPE
(*FILE) TGTRLS(V5R1M0) UPDHST(*NO) PRECHK(*YES) OUTPUT(*PRINT)>
: TGTRLS(*Current) Save Job <019701/EVWSS/PKZSAVA>
File Comment:"none"
-----
Found 2 files, 6755968 bytes uncompressed, 1357532 bytes compressed: 80%
    
```

IPSRA Example 3

Now we want to restore the saved library DED to a new library called DEDNEW and then restore the object TESTFILE to the new DEDNEW library. PKUNZIP can perform only one restore at a time, so the operation requires two steps.

Step1.

➔ **PKUNZIP ARCHIVE('/yourpath/bills/testsavx1.zip') TYPARCHFL(*IFS)
 FILES('SAVLIB01_DEDSAV01') TYPE(*EXTRACT)
 RSTIPSRA(RSTLIB SAVLIB(DED) DEV(RSTDED) output(*print)
 RSTLIB(DEDNEW))
 PASSWORD('bills00000')**

```

UNZIP Archive: /yourpath/bills/testsavx1.zip
Archive Comment:"SecureZIP for iSeries"
Searching Archive /yourpath/bills/testsavx1.zip for files to extract
Command:<RSTLIB SAVLIB(DED) RSTLIB(DEDNEW) OUTPUT(*PRINT)>
Extracting file SAVLIB01_DEDSAV01
Inflating *iPSRA:SAVLIB01_DEDSAV01 iPSRA File
SecureUNZIP extracted 1 files
SecureUNZIP Completed Successfully
    
```

Step 2.

➔ **PKUNZIP ARCHIVE('/yourpath/bills/testsavx1.zip') TYPARCHFL(*IFS)
FILES('SAVOBJ02_TESTOBJ11') TYPE(*EXTRACT)
RSTIPSR(RSTOBJ OBJ(TESTFILE) SAVLIB(TESTLIB)
DEV(RSTTEST) OBJTYPE(*FILE) RSTLIB(DEDNEW)
OUTPUT(*PRINT)) PASSWORD('bills00000')**

```
UNZIP Archive: /yourpath/bills/testsavx1.zip
Searching Archive /yourpath/bills/testsavx1.zip for files to extract
Command:<RSTOBJ OBJ(TESTFILE) SAVLIB(TESTLIB) OBJTYPE(*FILE)
RSTLIB(DEDNEW) OUTPUT(*PRINT)>
Extracting file SAVOBJ02_TESTOBJ11
Inflating *iPSRA:SAVOBJ02_TESTOBJ11 iPSRA File
SecureUNZIP extracted 1 files
SecureUNZIP Completed Successfully
```

iPSRA Example 4

The following example shows that we can restore one or more objects from an iPSRA file that was created with SAVLIB.

➔ **PKUNZIP ARCHIVE('/yourpath/bills/testsavx1.zip') TYPARCHFL(*IFS)
FILES('SAVLIB01_DEDSAV01') TYPE(*EXTRACT)
RSTIPSR(RSTOBJ OBJ(MYFILE2) SAVLIB(DED) DEV(RST1FILE)
OBJTYPE(*FILE) RSTLIB(DEDNEW) OUTPUT(*PRINT))
PASSWORD('bills00000')**

```
UNZIP Archive: /yourpath/bills/testsavx1.zip
Searching Archive /yourpath/bills/testsavx1.zip for files to extract
Command:<RSTOBJ OBJ(MYFILE2) SAVLIB(DED) OBJTYPE(*FILE)
RSTLIB(DEDNEW) OUTPUT(*PRINT)>
Extracting file SAVLIB01_DEDSAV01
Inflating *iPSRA:SAVLIB01_DEDSAV01 iPSRA File
SecureUNZIP extracted 1 files
SecureUNZIP Completed Successfully
```

iPSRA Example 5

The following example demonstrates the use of OUTFILE in a save command and shows how PKZIP automatically adds the outfile to the archive.

➔ **PKZIP ARCHIVE('/yourpath/bills/iPSRA_test/x3.zip')
TYPARCHFL(*IFS) FILES(
'-SAVLIB LIB(DED) DEV(DEDSAV01) OUTPUT(*OUTFILE)
OUTFILE(ATEST/DEDSAV01)'
'-SAV DEV('IFS_testpkcs7*') OBJ('/ajunk/testpkcs7/*')'
OUTPUT('/yourpath/bills/iPSRA_test/File01_SAV') ')**

```
Scanning files in *DB for match ...
Found 2 matching files
2 Save Command(s) selected
Command:<SAVLIB LIB(DED) OUTPUT(*OUTFILE) OUTFILE(ATEST/DEDSAV01)>
Compressing SAVLIB01_DEDSAV01 in SAVE Apps. Data mode
```

```
Add SAVLIB01_DEDSAV01 -- Deflating (90%)
Compressing ATEST/DEDSAV01(DEDSAV01) in TEXT mode
Add ATEST/DEDSAV01/DEDSAV01 -- Deflating (98%)
Command:<SAV OBJ('/AJUNK/TESTPKCS7/*')
      OUTPUT('/yourpath/BILLS/IPSRA_TEST/FILE01_SAV')>
Compressing SAV02_IFS_TESTPKCS7* in SAVE Apps. Data mode
Add SAV02_IFS_TESTPKCS7* -- Deflating (63%)
Compressing /yourpath/BILLS/IPSRA_TEST/FILE01_SAV in BINARY mode
Add /yourpath/BILLS/IPSRA_TEST/FILE01_SAV -- Deflating (61%)
SecureZIP Compressed 4 files in Archive /yourpath/bills/iPSRA_test/x3.zip
SecureZIP Completed Successfully
```

Four files are stored in the archive. Two are the iPSRA files, and the other two are the outfiles in the commands.

iPSRA Example 6

The example below shows a restore error.

➔ **PKUNZIP ARCHIVE('/yourpath/bills/testsavx1.zip') TYPARCHFL(*IFS)
 FILES('SAVLIB01_DEDSAV01') TYPE(*EXTRACT)
 RSTIPSRA(RSTOBJ OBJ(TESTFILE) SAVLIB(TESTLIB)
 DEV(RSTTEST) OBJTYPE(*FILE) RSTLIB(DEDNEW)
 OUTPUT(*PRINT)) PASSWORD('bills00000')**

```
UNZIP Archive: /yourpath/bills/testsavx1.zip
Archive Comment:"SecureZIP for iSeries"
Searching Archive /yourpath/bills/testsavx1.zip for files to extract
Command:<RSTOBJ OBJ(TESTFILE) SAVLIB(TESTLIB) OBJTYPE(*FILE)
RSTLIB(DEDNEW) OUTPUT(*PRINT)>
Extracting file SAVLIB01_DEDSAV01
Inflating *iPSRA:SAVLIB01_DEDSAV01 iPSRA File
Restore Operation encountered an error. See Job Log of PKZRSTA restore
job for further details.
SecureUNZIP extracted      0 files
SecureUNZIP found         1 file(s) in Error

Additional Message Information

Message ID . . . . . : AQZ1007
Date sent . . . . . : 08/09/05      Time sent . . . . . : 09:54:57

Message . . . . . : Restore Operation encountered an error. See Job Log of
PKZRSTA restore job for further details.

DSPSPLF FILE(QPJOBLOG) JOB(019721/EVWSS/PKZRSTA) for job log and
detail on why the restore operation failed. Possible problem may be that
some or all of the objects may not have been restored due to some restore
setting.

Since OUTPUT(*PRINT) was in effect you could view the restore output:

*...+...1...+...2...+...3...+...4...+...5...+...6...+...7...
5722SS1 V5R3M0 040528 RESTORE OBJECT INFORMATION
OBJECT NAME . . . . . : TESTFILE
SAVE LIBRARY . . . . . : TESTLIB
OBJECT TYPE . . . . . : *FILE
SAVE FILE NAME . . . . : QANE019357
SAVE FILE LIBRARY . . . : QTEMP
LABEL . . . . . : *SAVLIB
OPTION . . . . . : *ALL
```

```

MEMBER OPTION . . . : *MATCH
SAVE DATE/TIME . . . :
ALWOBJDIF . . . . . : *NONE
RESTORE LIBRARY . . . : DEDNEW
RESTORE ASP . . . . . : *SAVASP
Specified file for library TESTLIB not found.
          * * * * * E N D   O F   L I S T I N G * * * * *

Or from the DSPSPLF FILE(QPJOBLOG) JOB(019721/EVWSS/PKZRSTA) the job log will
show the actual IBM Restore error messages:

CPF3806   Diagnostic                20   08/09/05  09:54:57.125128  QANESERV
QSYS
From module . . . . . : QANESERV
From procedure . . . . . : QaneSendPgmMsg__FP14qanec_CTLBLK_tPct2iN24
Statement . . . . . : 19
To module . . . . . : QPOZPCPN
To procedure . . . . . : InvokeTargetPgm__FP11qp0z_pcp_cb
Statement . . . . . : 187
Message . . . . . : Objects from save file QANE019357 in QTEMP not restored.
Cause . . . . . : The library name in the save file does not match the
library name that you specified in the SAVLIB parameter. Recovery . . . :
Use the DSPSAVF command to display the save file and to determine the
library from which the objects were saved. Specify the correct library in
the SAVLIB parameter and try the command again.

CPF3780   Diagnostic                30   08/09/05  09:54:57.125152  QANESERV
QSYS
From module . . . . . : QANESERV
From procedure . . . . . : QaneSendPgmMsg__FP14qanec_CTLBLK_tPct2iN24
Statement . . . . . : 19
To module . . . . . : QPOZPCPN
To procedure . . . . . : InvokeTargetPgm__FP11qp0z_pcp_cb
Statement . . . . . : 187
Message . . . . . : Specified file for library TESTLIB not found.
Cause . . . . . : The data in the save file or on the tape, diskette or
optical volume did not match the specified parameters. Recovery . . . :
See the previously listed messages. If the restore operation is from
diskette, tape or optical, display the contents of the volume using the
appropriate display command specifying the DATA(*SAVRST) parameter. If the
restore operation uses a save file, display the contents of the save file
(DSPSAVF command). Correct any errors and then try the request again.

```

iPSRA Example 7

The example below shows how the save information is depicted for an object that was saved with PKZIP iPSRA and UPDHST (*YES) for the save command. Notice that the save file shows the save library of QTEMP and the save file as QANExxxxxx. This is an internal representation of the save API process. The device type will show as a save file.

→ DSPOBJD OBJ(DED) OBJTYPE(*LIB) DETAIL(*FULL)

```
Display Object Description - Full
Library 1 of 1
Object . . . . . : DED      Attribute . . . . . : TEST
Library . . . . . : QSYS    Owner . . . . . : WSS
Library ASP device . : *SYSBAS Primary group . . . : *NONE
Type . . . . . : *LIB

Save/Restore information:
Save date/time . . . . . : 08/10/05 11:16:41
Restore date/time . . . . . : 08/10/05 09:31:26
Save command . . . . . : SAVLIB
Device type . . . . . : Save file
Save file . . . . . : QTEMP/QANE020372
```