Co:Z SFTP – New features and Batch Job Best Practices

December 13, 2011

Kirk Wolf
Steve Goetze

DovetailedTechnologies

http://dovetail.com
info@dovetail.com

Dovetailed Technologies

We provide z/OS customers world wide with innovative solutions that enhance and transform traditional mainframe workloads:

- Co:Z Co-Processing Toolkit for z/OS
- OpenSSH Accelerator for z/OS
- T:Z Quickstart for Tomcat and z/OS
- JZOS - acquired by IBM in 2005 and now part of the z/OS Java SDK

Agenda

- Co:Z Toolkit release 2.0.0 release new features
  - currently in “beta”, expect to provide production support by 1/31/2012.
- Best practices for Co:Z SFTP batch jobs
- Using file / data set name patterns to set transfer options

New SMF Recording Features

- New SMF 119 Record Subtypes
  - Co:Z SFTP messages, informational level or above, that were associated with the previous transfer.
    - Subtype 192 - Co:Z SFTP server log messages
    - Subtype 193 - Co:Z SFTP client log messages
- New internal utility program “ssh-socket-info”
  - Invoked by Co:Z SFTP client to obtain accurate local and remote host/port information from child SSH process
  - Uses IBM EZBNMIFR network management API
  - APF Authorized with fallback to default behavior if not authorized

Extended Address Volumes (EAV) Support

- New support in Co:Z 2.0.0
- Actual I/O to DSORG=PS-E datasets is supported in z/OS 1.12 by the IBM C library
- Co:Z adds EAV support for FMT8/9 DSCBs and data set space calculation to Co:Z SFTP, SFTP dataset listings, and the `catsearch` command
- New allocation options added to Co:Z SFTP -

  `dsntype=library|pds|`**`large|extreq|extpref|basic`**

Co:Z SFTP Batch jobs using the original sample RUNSFTP JCL

```
// EXEC  PGM=COZBATCH
//STDIN  DD *  -- input to z/OS Unix shell
 # Customize these...
 coz_bin="/opt/dovetail/coz/bin"
 ruser="uid"
 server="remote.host.name"
 servercp="ISO8859-1"
 remotefile="/path/to/file"

 export PASSWD_DSN='//COZUSER.PASSWD(SITE1)'
 export SSH_ASKPASS=$coz_bin/read_passwd_dsn.sh
 export DISPLAY=none

 ssh_opts="-oBatchMode=no"
 ssh_opts="$ssh_opts -oConnectTimeout=60"
 ssh_opts="$ssh_opts -oServerAliveInterval=60"
```

RUNSFTP JCL (continued)

```
 ...

$coz_bin/cozsftp $ssh_opts -b- $ruser@$server <<EOB
lzopts mode=text,servercp=$servercp
get $remotefile //DD:DOWNLOAD
EOB

//DOWNLOAD DD DSN=...,DISP=(NEW,DELETE),
//            DCB=(...),SPACE=(...)
//
```

RUNSFTP sample - shortcomings

- Common settings repeated in every JCL deck
- Exposes low-level shell script code (to folks that may not know Unix)
- ➔ Becomes a maintenance problem as the inventory of jobs grows

New sample SFTPPROC and scripts

```
//SFTPGET EXEC PROC=SFTPPROC
//SFTPIN DD *
pwdsn="COZUSER.PASSWD(SITE1)"
user=myuser
host=myhost
lzopts="mode=text"
lfile=//DD:MYDSN
rfile=/etc/profile

. $script_dir/sftp_get.sh

//MYDSN DD DSN=COZUSER.SFTPGET.DATA,DISP=(MOD,KEEP),
//         DCB=(LRECL=80,RECFM=FB),SPACE=(CYL,(3,1))
```

Sample SFTPPROC

```
//EXSFTP    PROC ARGS=,
//   LIBRARY='COZUSER.COZ.LOADLIB',
//   SFTPIND='COZUSER.COZ.SAMPJCL(SFTPIND)',
//   REGSIZE='64M',
//   LEPARM=''
//RUNSFTP  EXEC PGM=COZBATCH,REGION=&REGSIZE,
//   PARM='&LEPARM/&ARGS'
//STEPLIB  DD DSN=&LIBRARY,DISP=SHR
//STDIN    DD DSN=&SFTPIND,DISP=SHR  < Inst Defaults
//         DD DDNAME=SFTPIN          < User input
//SFTPIN   DD DUMMY
// PEND
```

SFTPPROC – scripts and variables

- Sample scripts:
  `sftp_get.sh, sftp_put.sh, sftp_connect.sh, sftp_cat.sh`
- Connection and authentication variables:
  `user, host, port, pwdsn, cert`
- SFTP / SSH options variables:
  `sftp_opts`
- Configuration variables:
  `cozbin_dir, script_dir`
- Transfer variables:
  `lfile, rfile, lzopts, rpat`

Another example: using JCL variables

```
...
//SFTPPUT EXEC PROC=SFTPPROC,
// ARGS='user=&USER host=&HOST rfile=&RFILE'
//SFTPIN DD *
cert="MY-RING RSA-CERT"
lzopts="mode=text"
lfile=//DD:MYDSN

. $script_dir/sftp_put.sh

//MYDSN DD DSN=COZUSER.SFTPPUT.DATA,DISP=SHR
```

- See Co:Z Batch User's Guide for information on passing in variables > 100 characters

New sample SFTPPROC - benefits

- Unix shell script logic factored into separate script files
- Standard set of variables available to control connection, authentication, options, filenames
- Separation of installation default options from individual JCL members
- Can be customized and extended

➔ Simplifies maintenance and support

File Patterns

- Previously, options are set prior to a file transfer. Defaults may be set by user or installation, but apply to all files.
- Co:Z SFTP now allows options to be automatically set based on matched file / data set name patterns
- Patterns use shell pattern matching syntax - “fnmatch()”
- Pattern / option associations set in the Co:Z configuration files

Config Files

- Site-wide SFTP default options
  - **/etc/ssh/cozsftp_config** (cozsftp client)
  - **/etc/ssh/cozsftp_server_config** (server)

- Three Sections:

  fixed: - site-wide options that cannot be overridden
  default: - site-wide defaults that can be overridden
  pattern: - options activated when transferring a file or dataset that matches a name pattern

- User specific config files are now allowed
  - Located in $HOME/.ssh/
  - **pattern:** sections have priority over site config files
  - No **fixed:** or **default:** sections allowed

File Pattern Examples (cozsftp_config)

- Ensure that JCL members are transferred in text mode

```
pattern: //*.JCL(*)
mode=text
```

```
cozsftp> put //MY.DEV.JCL(RUNJ) runj.txt
```

- Allocate space for specific datasets automatically

```
pattern: //*.TRANS*
space=cyl.3.2,recfm=fb,lrecl=80
```

```
cozsftp> get remote.file //PARTNER.TRANS03
```

File Pattern Examples, Con't

$HOME/.ssh/cozsftp_server_config

```
pattern: *.txt
mode=text,clientcp=1252,linerule=crlf
```

/etc/ssh/cozsftp_server_config

```
pattern: *.zip
mode=binary
pattern: *.pax
mode=binary
pattern: *.txt
mode=text,linerule=lf
```

```
sftp> get myarchive.pax local.tar
sftp> get mynotes.txt local.text
```

Client Session Options Priority Order

- Client session settings are determined in the following priority order (from highest to lowest):

1. The **fixed:** section of `/etc/ssh/cozsftp_config`
2. The first matching pattern (if any) from `$HOME/.ssh/cozsftp_config`
3. The first matching pattern (if any) from `/etc/ssh/cozsftp_config`
4. The interactive command: **`lzopts`**
5. The environment variable `SFTP_ZOS_OPTIONS`
6. The **default:** section of `/etc/ssh/cozsftp_config`

Server Session Options Priority Order

- Server session settings are determined in the following priority order (from highest to lowest):

1. The **fixed:** section of `/etc/ssh/cozsftp_server_config`
2. The first matching pattern (if any) from `$HOME/.ssh/cozsftp_server_config`
3. The first matching pattern (if any) from `/etc/ssh/cozsftp_server_config`
4. The interactive command: **`ls /+`**
5. The environment variable `SFTP_ZOS_OPTIONS`
6. The **default:** section of `/etc/ssh/cozsftp_server_config`

More information

- IBM Ported Tools for z/OS: OpenSSH User's Guide
- Co:Z SFTP User's Guide
- http://dovetail.com/forum (public bulletin board)
- Open a support ticket with your questions, or call us
- Our webinar archives: http://dovetail.com/webinars
- Future webinar:
  Using z/OS SAF Keyrings with SSH and SFTP