Migrating System Operations and Administration from Version “H” to Version “L”

A *QSS* Complimentary Webinar

Wednesday, October 2, 2013

10:00 AM – 12:00 PM

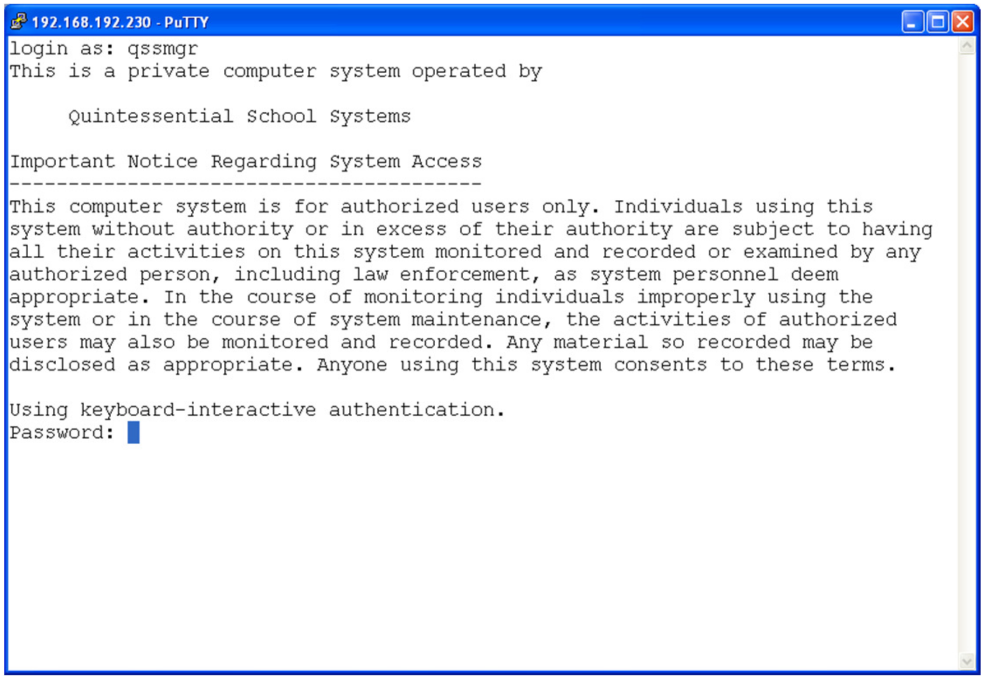
Logging On

- Type user and press enter after

  `logon as:`

- Type password and press enter after

  `Password:`



```
login as: qssmgr
This is a private computer system operated by

     Quintessential School Systems

Important Notice Regarding System Access
----------------------------------------
This computer system is for authorized users only. Individuals using this
system without authority or in excess of their authority are subject to having
all their activities on this system monitored and recorded or examined by any
authorized person, including law enforcement, as system personnel deem
appropriate. In the course of monitoring individuals improperly using the
system or in the course of system maintenance, the activities of authorized
users may also be monitored and recorded. Any material so recorded may be
disclosed as appropriate. Anyone using this system consents to these terms.

Using keyboard-interactive authentication.
Password:
```

System Users

- root – System administrator
  - Equivalent to MANAGER.SYS
  - ‘Best Practice’ is to use this user only when needed to prevent making a mistake while learning
  - Most commands do not prompt for confirmation
  - There is no undo!
  - Root can do anything!

QSS Users

- qssmgr – QSS System Administrator
  - Equivalent to MGR.QSSUSER
  - Used to install software and configure the system
- qssopr – QSS System operator
  - Equivalent to OPR.QSSUSER
  - Used for accounts payable and payroll processing

QSS Users (con’t)

- qssdba – QSS Database administrator
  - Equivalent to MGR.QSSUSER using QUERY
  - Is a SQL database user
  - Is only a Linux user if using PostgreSQL

- ADMIN - QCC User
  - Equivalent to SYSMGR.QSSUSER

Test Account

- qssmgrtest
  - Equivalent to MGR.QSSTEST
- Qssoprtest
  - Equivalent to OPR.QSSTEST
- Directory tree /var/opt/qss/test/
  - Equivalent to QSSTEST account

QSS Directories

- /etc/opt/qss – QSS configuration files
  - $QSS_CFG variable
- /opt/qss – QSS software
  - Equivalent to QSSPROD account
  - $QSS_ROOT variable
    - Replaces PACT
- /var/opt/qss – QSS data files
  - Equivalent to QSSUSER account
  - $QSS_DATA variable

QSS Configuration Files in /etc/opt/qss

- jobsched/ – configuration directory for QSS Job System
  - datemask – contains accepted date masks for submitting jobs with the -at option
  - jobid – job number odometer. Modify to reset your job numbers.
  - JobSec.cfg – lists users allowed to submit jobs. Maintain with JobSecCfg application
  - JobSvr.cfg – configures port number, queues, limits, file locations, etc.

QSS Configuration Files in /etc/opt/qss

- login/ – login directory (replaces logon UDC’s)
  - copyright – QSS copyright statement
  - message – login message
  - netcobolrt.sh – script to set up path to NetCOBOL runtime
  - PATH.qss – Added to $PATH of QSS user. Modify this to change path for all QSS users as directed by QSS.
  - qss_check – “hooked” into system login file
  - qssenv.sh – sets the default environment variables
  - **qsslocal.sh** – sets customer specific variables and commands. Modify this file to set your values.
  - qss_login – main login script
  - qssrun.sh – list of commands run when user logs in
  - user/ - directory for user specific settings

QSS Configuration Files in /etc/opt/qss

- qat.d/ – QSS automated tasks directory
  - logfiles - settings to restrict size of log files
  - releases - settings to remove older release / archive files
  - stdlists - settings for how to archive and delete stdlists
  - pfiles - reports/removes older print files
- qcc_ctl.cfg – configuration file for QCC servers (similar to QLOASIS.DATAQCC)
- qdbi/ – database configuration

QSS Software Directories in /opt/qss

- archive – archived versions of files
- bin – essential QSS binaries (utilities)
- DataDirect – DataDirect libraries for accessing database
- finance – Directory for QSS/OASIS software
- install – skeleton and initial version of installation files
- jobsched/bin – binaries for QSS Job System
- lib – QSS libraries
- qat – QSS Automated Tasks modules
- qcc/bin – binaries for QCC servers
- sbin – binaries for system tasks done by qssmgr

QSS Data Directories in /var/opt/qss

- cmd/ - command files, scripts, for any qss user
  - Individual files replace UDC for all users
- cmd/mgr/ - command files, scripts, for qssmgr
  - Individual files replace UDC for MGR.QSSUSER
- cmd/opr/ - command files, scripts, for qssopr
  - Individual files replace UDC for OPR.QSSUSER
- data/ - data files
  - Replaces DATA group
- data/qcc - data files for QCC servers
  - Replaces DATAQCC group
- dist/###/ - district specific files
  - Replaces DIST## groups

QSS Data Directories in /var/opt/qss

- fmt/ - printer format files
  - replaces PSIFMT.SYS group
- history/*subdir*/ - history files separated into sub directories such as 1099 - 1099 files, apy - accounts payable, pay - payroll, ret - retirement, W2 - w2 files, etc.
  - replaces HISTORY group
- jcl/ - site specific jcl files
  - replaces JCL group
- log/ - log files
- parms/ - parameter files
  - replaces PARMS group

QSS Data Directories in /var/opt/qss

- pdl/ - personnel downloader files and district specific sub directories
  - replaces PPDNLD@ groups
- pfiles/ - print file directory
  - print files are now stored in one directory with subdirectories for each user instead on various MPE home groups
- pox/ - purchase order print driver files
- qcc/ - execution directory for QCC servers
  - replaces the temporary files for QCC servers and their STDLISTS
- release/ - release directory
  - replaces release groups in QSSDEV account

QSS Data Directories in /var/opt/qss

- tmp/ - QSS temporary files
  - replaces MPE temporary files
- w21099/*year*/ - W2 and 1099 files in year specific sub directories
  - replaces W21099## groups
- web/ - web template files for QCC servers
  - replaces WEBQCC group
- xfr/ - xfr directory used by QSSXFR

Editing Files

- MPE
  - QEDIT
- Linux
  - vi
  - PC editor such as notepad, use WinSCP to transfer file from/to Linux
  - Other Linux editor

Using vi to Edit Files

- To edit a file with vi
  - vi filename
- vi's modes
  - Normal mode
    - One or two keystroke commands to navigate and make changes
    - Keystrokes are not echoed in most cases
  - Insert mode
    - Press 'i' to enter insert mode
    - Can use arrow keys, page up, page down, to move around, delete key to delete characters, what you type is inserted, etc. Much like Notepad.
    - Press Esc to return to normal mode
- To exit vi (must be in Normal mode)
  - :wq   to save file and exit
  - :q   to exit if you didn't make any change
  - :q!   to exit without saving
- To learn vi use the command: vimtutor

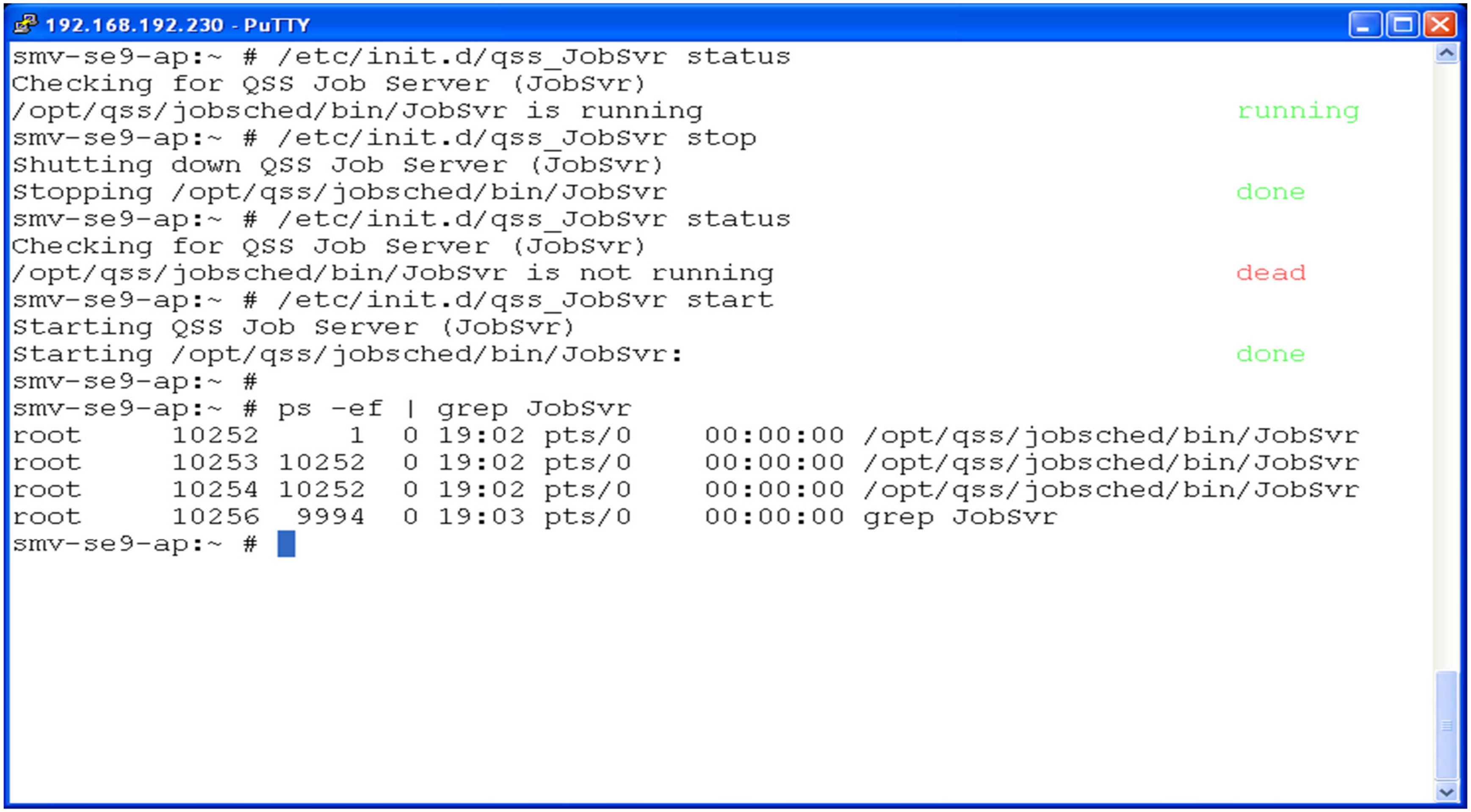
Batch Jobs

- Job processing built into MPE
  - $STDLISTS are a type of print file
- In Linux, done via additional software, the QSS Job System
  - STDLIST is a flat file
  - STDLIST printed via QCC Print Manager
  - Job System controlled via JobCli program or other commands

Starting/Stopping QSS Job System

- Status (from ‘qssmgr’ or ‘root’)
  - /etc/init.d/qss_JobSvr status
- Starting (from ‘root’ user)
  - /etc/init.d/qss_JobSvr start
  - Starts every “instance” defined in /etc/sysconfig/qss_JobSvr
  - Can be set to start automatically at boot
- Stopping (from ‘root’ user)
  - /etc/init.d/qss_JobSvr stop
  - Stops every “instance” defined in /etc/sysconfig/qss_JobSvr
  - Can be set to stop automatically at shutdown
- Stopping (from “ qssmgr “ user)
  - /opt/qss/jobsched/bin/JobCli
  - Use “connect“ and “stop“ commands

qss_JobSvr Examples



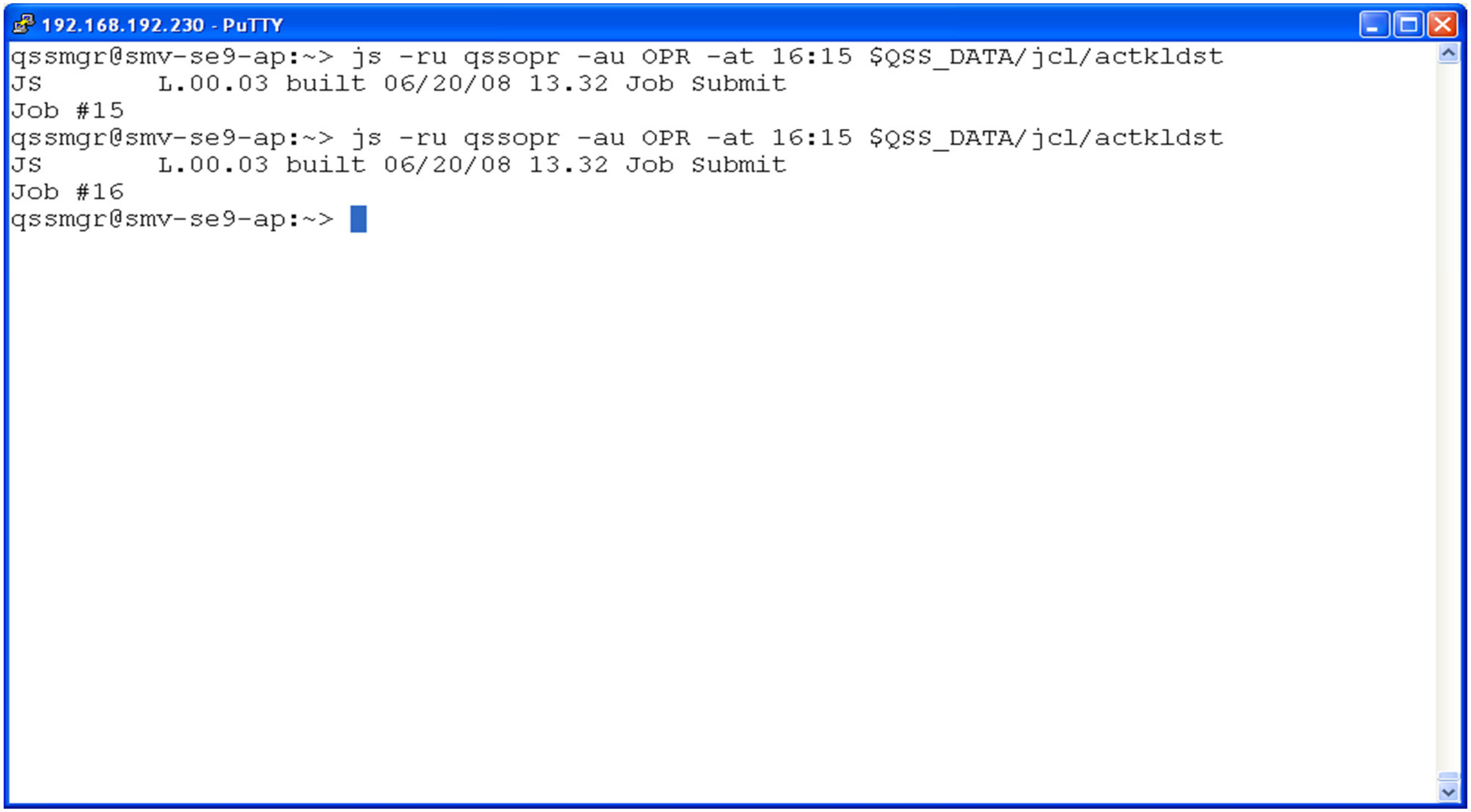
```
smv-se9-ap:~ # /etc/init.d/qss_JobSvr status
Checking for QSS Job Server (JobSvr)
/opt/qss/jobsched/bin/JobSvr is running                                     running
smv-se9-ap:~ # /etc/init.d/qss_JobSvr stop
Shutting down QSS Job Server (JobSvr)
Stopping /opt/qss/jobsched/bin/JobSvr                                       done
smv-se9-ap:~ # /etc/init.d/qss_JobSvr status
Checking for QSS Job Server (JobSvr)
/opt/qss/jobsched/bin/JobSvr is not running                                 dead
smv-se9-ap:~ # /etc/init.d/qss_JobSvr start
Starting QSS Job Server (JobSvr)
Starting /opt/qss/jobsched/bin/JobSvr:                                      done
smv-se9-ap:~ #
smv-se9-ap:~ # ps -ef | grep JobSvr
root     10252     1  0 19:02 pts/0    00:00:00 /opt/qss/jobsched/bin/JobSvr
root     10253 10252  0 19:02 pts/0    00:00:00 /opt/qss/jobsched/bin/JobSvr
root     10254 10252  0 19:02 pts/0    00:00:00 /opt/qss/jobsched/bin/JobSvr
root     10256  9994  0 19:03 pts/0    00:00:00 grep JobSvr
smv-se9-ap:~ #
```

(C) 2013 QSS

QSS Job System Add-on Commands

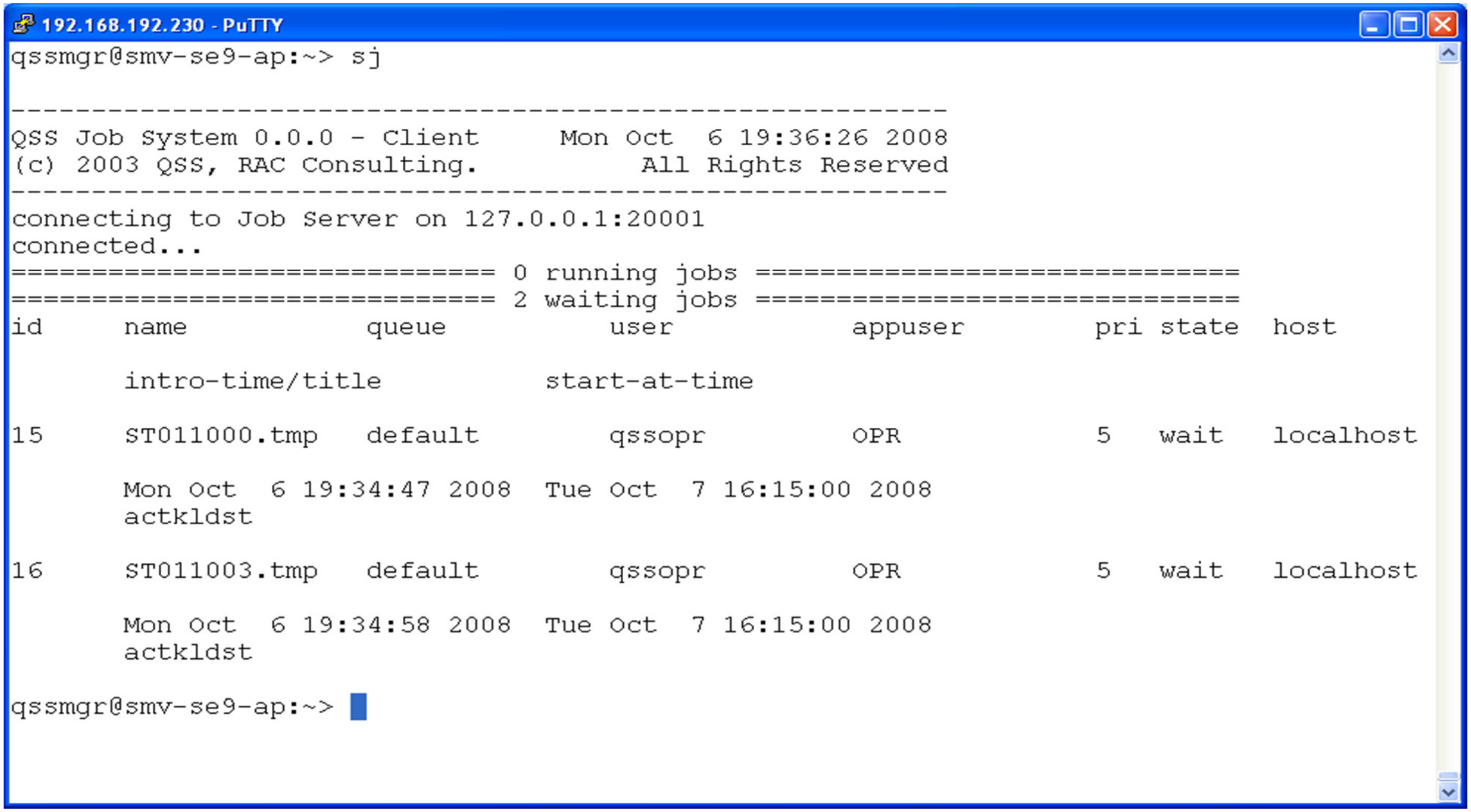
- sj – show job
  - Use sj -h to see syntax
- dj <jobnum> – delete job
- js – job submit
  - Use js -h to see syntax
- lsl [<user>]– list stdlists
  - use lsl -l for long version
- prjob <jobnum> - print stdlist
- dljob <jobnum> - delete stdlist (in development)

js Example to Schedule Jobs



```
qssmgr@smv-se9-ap:~> js -ru qssopr -au OPR -at 16:15 $QSS_DATA/jcl/actkldst
JS        L.00.03 built 06/20/08 13.32 Job Submit
Job #15
qssmgr@smv-se9-ap:~> js -ru qssopr -au OPR -at 16:15 $QSS_DATA/jcl/actkldst
JS        L.00.03 built 06/20/08 13.32 Job Submit
Job #16
qssmgr@smv-se9-ap:~>
```

sj Example



```
qssmgr@smv-se9-ap:~> sj

----------------------------------------------------------
QSS Job System 0.0.0 - Client     Mon Oct  6 19:36:26 2008
(c) 2003 QSS, RAC Consulting.          All Rights Reserved
----------------------------------------------------------
connecting to Job Server on 127.0.0.1:20001
connected...
============================== 0 running jobs ==============================
============================== 2 waiting jobs ==============================
id     name           queue          user           appuser        pri state  host

       intro-time/title          start-at-time

15     ST011000.tmp   default        qssopr         OPR            5   wait   localhost

       Mon Oct  6 19:34:47 2008  Tue Oct  7 16:15:00 2008
       actkldst

16     ST011003.tmp   default        qssopr         OPR            5   wait   localhost

       Mon Oct  6 19:34:58 2008  Tue Oct  7 16:15:00 2008
       actkldst

qssmgr@smv-se9-ap:~>
```

Job Queues

- A “default” job queue is installed on QSS/OASIS Version L servers
- Additional queues may be defined
  - For a specific job (set by line in jcl)
  - For a specific user, ex. qssopr (set by variable)
- Job queues have…
  - Max number of jobs, start priority
  - Can alter queue using JobCli
    - Equivalent to LIMIT and other commands

QCC Servers

- MPE
  - Are batch jobs and have a STDLIST
  - Control file QLOASIS.DATAQCC
- Linux
  - More like traditional Unix daemon
  - Output in files
    - $QSS_DATA/qcc/<server>/<server>-stdout.log
    - $QSS_DATA/qcc/<server>/<server>-stderr.log
  - Temporary files in directory
    - $QSS_DATA/qcc/<server>
  - Control file $QSS_CFG/qcc_ctl.cfg
    - similar to QLOASIS

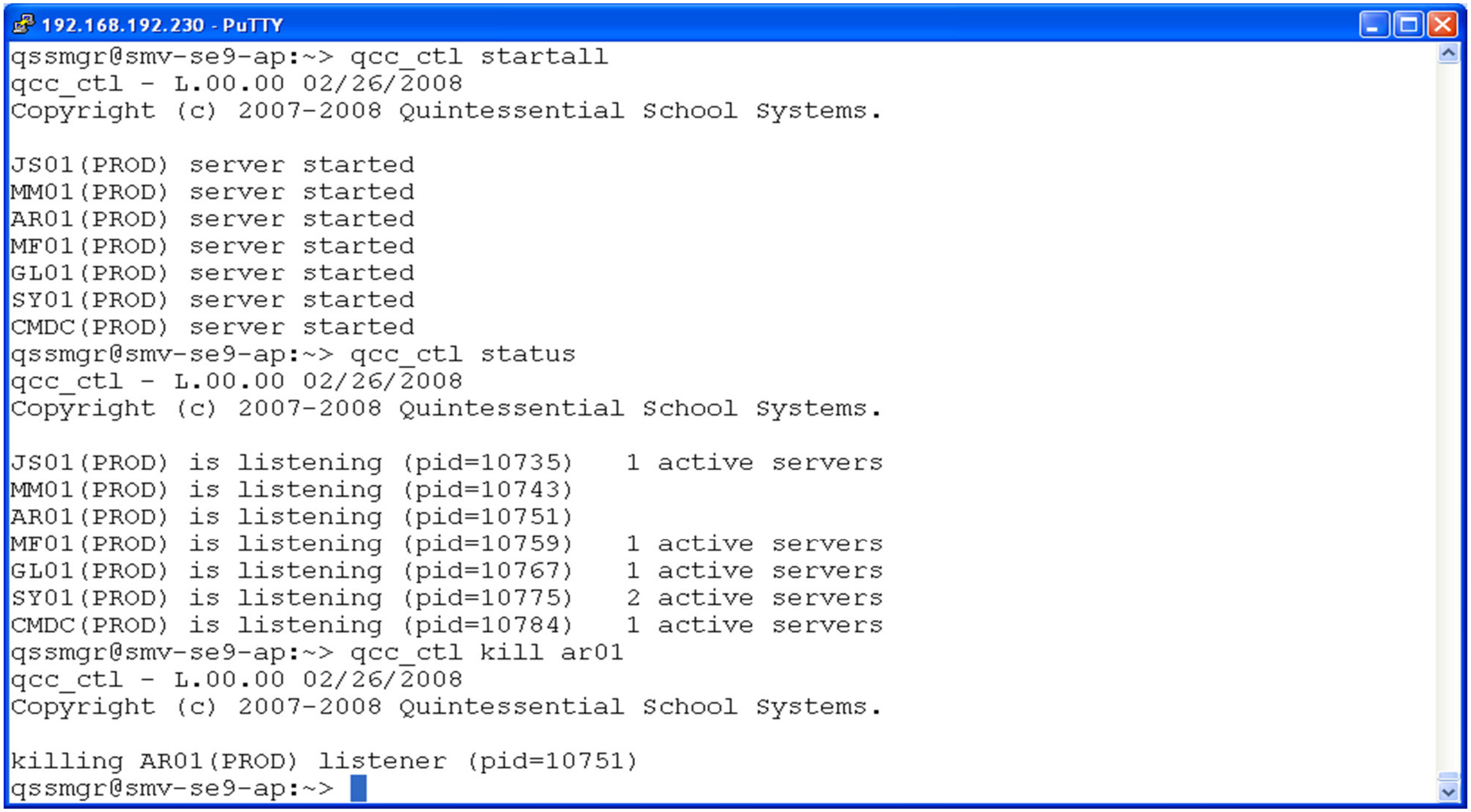
Starting/Stopping QCC Servers from “root”

- QCC servers status
  - /etc/init.d/qss_qcc_ctl status
  - Status of servers for all “instances”
- Starting QCC servers
  - /etc/init.d/qss_qcc_ctl start
  - Starts servers for all “instances.” Can be set to start automatically at boot.
- Stopping QCC servers
  - /etc/init.d/qss_qcc_ctl stop
  - Stops servers for all “instances”
  - Can be set to stop automatically at shutdown

Starting/Stopping QCC Servers from ‘qssmgr’

- qcc_ctl manages servers for the current ‘instance’
  - Status for all servers
    - qcc_ctl status
  - Status for a single server
    - qcc_ctl status gl01
  - Start all servers
    - qcc_ctl startall
  - Start a single server
    - qcc_ctl start gl01
  - Stop all servers
    - qcc_ctl killall
  - Stop a single server
    - qcc_ctl kill gl01

qcc_ctl Examples



```
qssmgr@smv-se9-ap:~> qcc_ctl startall
qcc_ctl - L.00.00 02/26/2008
Copyright (c) 2007-2008 Quintessential School Systems.

JS01(PROD) server started
MM01(PROD) server started
AR01(PROD) server started
MF01(PROD) server started
GL01(PROD) server started
SY01(PROD) server started
CMDC(PROD) server started
qssmgr@smv-se9-ap:~> qcc_ctl status
qcc_ctl - L.00.00 02/26/2008
Copyright (c) 2007-2008 Quintessential School Systems.

JS01(PROD) is listening (pid=10735)   1 active servers
MM01(PROD) is listening (pid=10743)
AR01(PROD) is listening (pid=10751)
MF01(PROD) is listening (pid=10759)   1 active servers
GL01(PROD) is listening (pid=10767)   1 active servers
SY01(PROD) is listening (pid=10775)   2 active servers
CMDC(PROD) is listening (pid=10784)   1 active servers
qssmgr@smv-se9-ap:~> qcc_ctl kill ar01
qcc_ctl - L.00.00 02/26/2008
Copyright (c) 2007-2008 Quintessential School Systems.

killing AR01(PROD) listener (pid=10751)
qssmgr@smv-se9-ap:~>
```

Temporary Files

- MPE
  - Temporary files exist in ;TEMP file domain
  - Built in to OS
- Linux
  - No such concept exists in Linux
  - Each process at logon creates a random directory in $QSS_DATA/tmp
  - Variable $QSS_TMP_DIR set to directory
  - Trap automatically deletes directory and files when process logs off
  - Creating files in other directories like $HOME or $QSS_DATA/data is equivalent to saved files

Running Programs in MPE

- RUN PROGNAME.GROUP.ACCOUNT
  - Example: RUN PAY000.!ppsch.!pact
- Some commands found based on !HPPATH
- JCW's needed by program set via SETJCW
  - Example: SETJCW OPTION,1
- A few programs get input via ;INFO= ;PARM=
  - No equivalent in Linux (close to command line arguments, if programmed, or replaced with variable)
- Redirection via < and > available but rare in both QSS/OASIS Version H and L

Linux Differences

- Linux is case sensitive!!!
  - Variables usually UPPERCASE
  - Most everything else lowercase, commands, program names
- Each process runs in its own shell with it's own set of variables inherited from parent process
  - Child process cannot pass variables back to parent shell.

Running Programs in Linux

- RUN PROGNAME.GROUP.ACCOUNT
  - Example: $QSS_ROOT/$QSS_BINSCH/pay000
    - $QSS_ROOT equivalent to !PACT
    - $QSS_BINSCH equivalent to !PPSCH
- Some commands found based on $PATH
- JCW’s needed by program set via setjcw
  - Example: . setjcw OPTION 1
    - Jcw’s not part of OS
    - Resulting variable is QENV_OPTION=00001
- Redirection via << EOD (known here as document very common in jcl files)

QSS Version

- Replacement for QSS/OASIS Version H getver command or MPE's version commands
- Use -h switch for list of options
  - qv -h
- Examples
  - qv $QSS_ROOT/lib/libqapp.so
    - Similar to 'version qsslib.library'
  - qv $QSS_ROOT/$QSS_BINSCH/gl0100
    - Similar to 'version gl0100.!ppsch.!pact'

Managing Nightly Processes

- Processes include ACTKLD, BALFIX, VENKLD
- JCL files are customizable, usually require end of year modification, and reside in $QSS_DATA/jcl
- Schedule file to specify when
  - $QSS_DATA/data/sched_actkld
- Command file to submit job
  - sub_actkld   submit reoccurring scheduled job based on schedule file
  - sub_actkld once   submit immediately
  - sub_actkld 18:00   submit reoccurring at 6:00 pm

Sample Setup of Accounts Payable Processing

- Logon as ‘qssopr’
- Scripts are in $QSS_DATA/cmd/opr
  - apyextract – extracts data to file in /home/qssopr
  - apywarr – prints warrants
  - apyreport – submits $QSS_DATA/jcl/apyrptst job to print reports
  - apyupdate - submits $QSS_DATA/jcl/apyrptst job to update database
  - apysave – saves production files from /home/qssopr to history directory $QSS_DATA/history/apy
- Process is customized for each client

Sample Setup of Payroll Processing

- Logon as ‘qssopr’
- Scripts are in $QSS_DATA/cmd/opr
  - – payextract – extracts data to file in /home/qssopr
  - – paywarr – prints warrants
  - – payreport – submits $QSS_DATA/jcl/payrptst job to print reports
  - – payupdate - submits $QSS_DATA/jcl/payupdst job to update database
  - – paysave – saves production files from /home/qssopr to history directory $QSS_DATA/history/pay
- Process is customized for each client

QSS/OASIS Releases

- Software Releases - keeps program, jcl, data files current
  - GS36-0, GS36-1, …
  - Current schedule includes new release every Thursday
  - Automated process that invokes install_qss script
  - Archive current files in $QSS_ROOT/archive
- Environment Releases - keep the QSS directory structure, login files, etc. current
  - Example EN34-0
  - Less common, but usually accompanies GSxx-0 release
  - Automated process that invokes build_qss script
- Sysctl data update - updates sysctl tables with latest menus, modules, tasks, etc.
  - sysctl_upate-1.36
  - Usually accompanies GSxx-0 release
  - Delivered via email, manual install process

autoupdate_qss

- Checks for new GS and EN releases
- Downloads and installs release automatically
- Releases are currently password protected
- Can be done either “after hours,” or while “in use”
  - “in use” method - safely replaces files being used allowing current process to finish using the previous file. May have “after hours” step to restart QCC servers.

autoupate_qss help Example

```
> autoupdate_qss -h
autoupdate_qss (QSS Auto Update) - L.01.02 03/21/2012
Copyright (c) 2009-2013 Quintessential School Systems.
 Syntax
  autoupdate_qss [-l] [-r rel_id] [-v]
  autoupdate_qss [-i | -n] [-r rel_id] [-v]

 Options
  -h         Help.
  -i         Interactive install.  Default option is an automated install.
             Cannot use this with -n.
  -l         List the current releases and indicate status of release.
             Does not install any release or get the latest list of releases.
  -n         Do not install release after downloading and extracting.
             Default option is an automated install.  Cannot use this with -i
  -r rel_id  Work with a specific release instead of just the all releases
  -v         Verbose mode. The script already shows what it is doing but this adds some additional detail.

 Examples
 To download, extract, and install all new releases
      autoupdate_qss

 To see what releases are currently known and check if they are installed or not
      autoupdate_qss -l

 To download and extract all new releases but not install. Install must be done manually at a later time.
      autoupdate_qss -n

 To download, extract, and install a specific releases and have an interactive install.
      autoupdate_qss -i -r GS22-0
```

“Hot Fix”

- In MPE
  - files put in QSSDEV account
  - copied to QSSPROD account
- In Linux
  - files put in /tmp
  - Installed via correct utility command
    - qbininst (programs), qjclinst (jcl files), qlibinst (libraries)
    - Commands detect if file is in use, can report what processes are using it, and uses appropriate methods to update even if in use
    - Previous version is archived: $QSS_ROOT/archive/updates
  - qundoinst allows individual file to be rolled back

TurboImage

- Logged on to QSSUSER account
- Use QUERY to access
  - Requires experienced user, command line based
- Naming conventions
  - Begins with A- for automatic master data sets
    - No such need in SQL
  - Begins with M- for manual master data sets
  - Begins with D- for detail data sets
- Multiple keys require duplicating data
  - Example, DI-NO, DI-SSN, DI-SSN-YR

SQL

- Most commonly accessed via a GUI client
  - Microsoft SQL Server Management Studio (SQL Server only)
  - DbVisualizer (any platform)
  - pgAdmin (PostgreSQL only)
- Fewer databases
  - hrspay, finance, sysctl, pque, strsmf
- Naming Convention
  - Lowercase names
  - Dash ‘-’ replaced by underscore ‘_’
  - Leading D- and M- dropped
  - Some names changed to prevent name conflict or indicate original database

Databases

- Version “L”
  - finance
  - hrspay
  - pque
  - strsmf
  - sysctl

- Version “H”
  - GLDSYS, ACTRCV, BDEVOL, CONVRT, ECOMRC, FASSET, RECON, STRSYS
  - PERPAY, BENMGT, PPHIST, PYHIST, RETSYS, STTSYS
  - PQUE
  - STRSMF
  - SYSCTL, MENU, MENCTL, QDACTL, QSSPRE