Guardium Activity Monitor & Db2 for i
Serviceability Guide
Version 3.3

Scott Forstie
forstie@us.ibm.com

Table of Contents

Preface:

This document contains basic IBM i service techniques and Guardium – Db2 for i as a data source service detail.

Technical Contacts:

When this document doesn't address all your questions, problems or customer requests for enhancement, contact the following:

Author:
Scott Forstie
Title: Db2 for i Business Architect & Db2 for i SQL Team Leader
Email: forstie@us.ibm.com

Version History:

Version 1.0 – Original document
Version 2.1 – Added FTP sections and Uninstall instructions.
Version 2.2 – Added Version history, S-TAP for IBM i instructions, entity → Report field mapping for Exception and Activity reports, details for having a dedicated subsystem for the Audit Server and expanded on best practices.
Version 2.3 –

- Revised recommended report definitions & entity mapping
- Added PREVENT_SKIPPED_ENTRIES to the configuration control
- Added NUMBER_SKIPPED_QAUDJRN_ENTRIES to the Audit Server status

Version 2.4 –

- Clarify authorization requirements for SYSPROC procedures
- Add COMMENT ON trace control for ERROR ONLY

Version 3.0 –

- Add IBM i Command Cheat Sheet Appendix
- Enhanced Audit Server trace option
- GO SAVE option 21 (21. Entire system) automation steps for restarting the Audit Server

Version 3.1 –

- Clarify Subsystem setup steps

Version 3.2 –

- Add section "Automating the restart of the Audit Server when the Audit Server subsystem is manually ended and restarted"
- Add failure and remediation steps to the "Recycling the Audit Server" section
- Add advertisement to use ACS instead of STRSQL
- Added authorization detail for iSTAP administration

Version 3.3 –

- Updated iS-TAP version sections

Resources

Refer to and use the following education resources.

- **Guardium Data Monitoring - Db2 for i fact page**
  - https://ibm.biz/GuardiumDAMonIBMi

- **Guardium Data Monitoring - Db2 for i White Paper**
  - http://www.ibm.com/developerworks/ibmi/library/i-infosphere_guardium_db2

- **Guardium Activity Monitor & Db2 for i Serviceability Guide (this document)**
  - https://ibm.biz/GuardiumOniServiceabilityGuide

- **Best of breed Db2 for i SQL tool… IBM i Access Client Solutions (ACS)**
  - If you're using STRSQL, you need to shift to using ACS.
    You will be far more productive. There is no charge to using this tool for Run SQL Scripts activity. This tool is based on Java, so it works anywhere Java is supported.

    For education, look here:
    http://www.omniuser.org/downloads/omniTech17ForstieRoweWhat'snewinIBMiAccessClient%20Solutions(ACS).pdf

    To download this tool, go here:
    (note- you need an IBM ID, but there is no charge)
    (note- follow the readme for installation details)
    https://www.ibm.com/services/forms/preLogin.do?source=swg-ia

Service Checklist

If you follow these basic steps (1-2-3's, a-b-c's, you get the point) you can self-diagnose and solve many of the basic setup and usage problems.

- Verify the IBM i service level. Look here (http://bit.ly/GuardiumOni) for the Db2 PTF Group level and for any additional unique PTFs.
- Examine the Audit Server configuration.
  From the configuration information:
  - Confirm whether the collector IP address is correct. Try pinging the collector.
  - Display details about the user profile. Confirm that the user profile exists and has adequate authority.
  - Understand whether filters are being used.
- Call the Audit Server status procedure and examine the results.
- When you need help, collect the detail as explained in the next section.

Collecting detail/documents before engaging Level 3

There are a few of us who understand the Guardium and Db2 for i support very well. We're always interested in helping our extended teammates and their customers to achieve success with Guardium. In those cases where this document and the White paper haven't gone far enough, this section lists the information required by Level 3.

For the sake of overall efficiency, please gather the following documents when engaging Level 3.

1) Gather the Db2 PTF Group level
   WRKPTFGRP and look for SF99601 or SF99701
   or
   > STRSQL
   SELECT CHAR(PTF_GROUP_NAME,7) as GRPPTF, PTF_GROUP_LEVEL FROM QSYS2.GROUP_PTF_INFO WHERE PTF_GROUP_NAME IN ('SF99701', 'SF99601') AND PTF_GROUP_STATUS = 'INSTALLED'
   ORDER BY PTF_GROUP_LEVEL DESC FETCH FIRST 1 ROWS ONLY

2) Determine the user profile used by the audit server and dump it
   > STRSQL
   SELECT start_user FROM QSYS2/SYSAUDIT
   > PF3 (exit) with option 1
   > DSPUSRPRF USRPRF(<start_user-name>) TYPE(*BASIC) OUTPUT(*PRINT)
3) Determine the Audit server job and dump the joblog and job.
   > STRSQL
   CALL SYSPROC/SYSAUDIT_STATUS()
   SELECT rtrim(substr(server_job,21,6)) concat '/' concat rtrim(substr(server_job,11,10)) concat '/' concat substr(server_job,1,10) from QTEMP/SYSAUDSTS
   > PF3 (exit) with option 1
   > DSPJOBLOG JOB(<audit-jobname>) OUTPUT(*PRINT)
   > DSPJOB JOB(<audit-jobname>) OUTPUT(*PRINT)
4) Determine the S-TAP for IBM i job and dump the joblog and job.
   > WRKOBJLCK OBJ(<start_user-name>) OBJTYPE(*USRPRF)
   > Look for the job named QP0ZSPWT and enter 5 (Work with job)
   > Job: QP0ZSPWT User: SCOTTF Number: 415391
   > DSPJOBLOG JOB(415391/SCOTTF/QP0ZSPWT) OUTPUT(*PRINT)
   > DSPJOB JOB(415391/SCOTTF/QP0ZSPWT) OUTPUT(*PRINT)
5) Call the status procedure and capture the output.

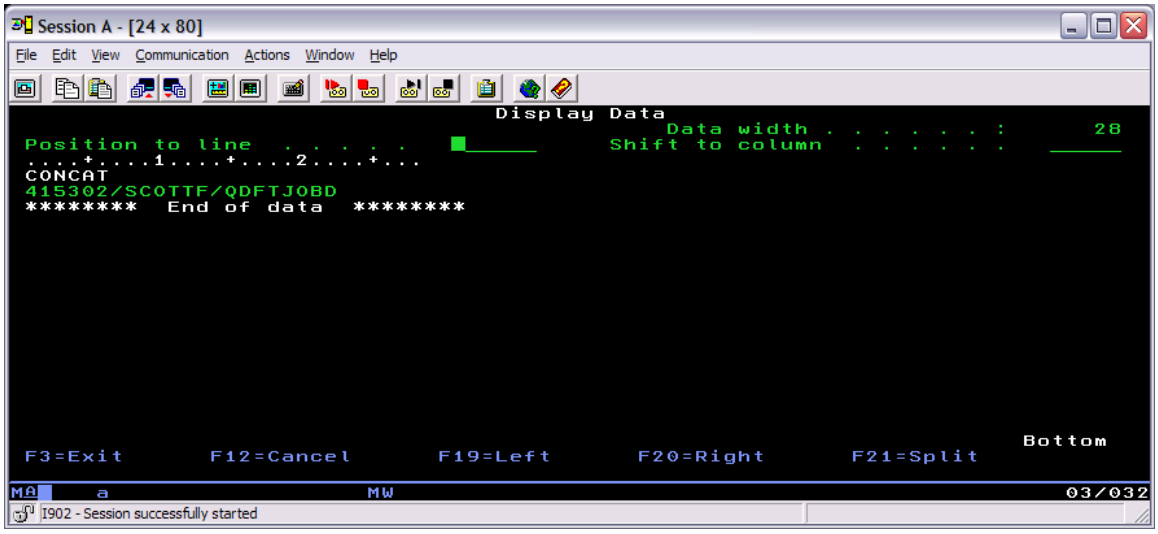
6) Capture the QSYS2/SYSAUDIT file settings.
7) Capture the security configuration using the Display Security Auditing (DSPSECAUD) command:
   > DSPSECAUD

Items 1-4 will appear in the spool file. Spool files can be saved as files on a PC and submitted by attaching them to an email.
Items 5&6 are database files that can either be displayed or saved (sent as attachments)
Item 7 is shown on a screen.

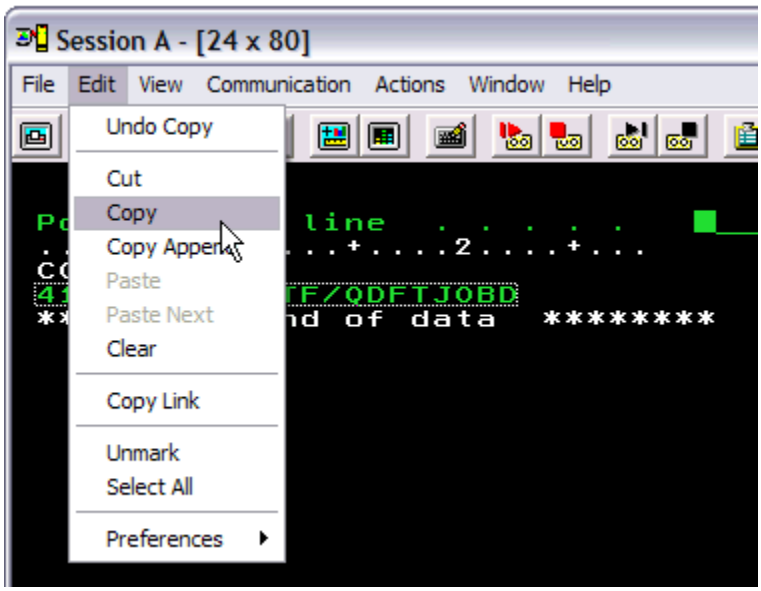
Examining the Audit Server job

Let's find the Audit Server job and try a few commands against it.

> **strsql**
> **call sysproc/sysaudit_status()**
> **select rtrim(substr(server_job,21,6)) concat '/' concat**
rtrim(substr(server_job,11,10)) concat '/' concat
substr(server_job,1,10) from QTEMP/SYSAUDSTS

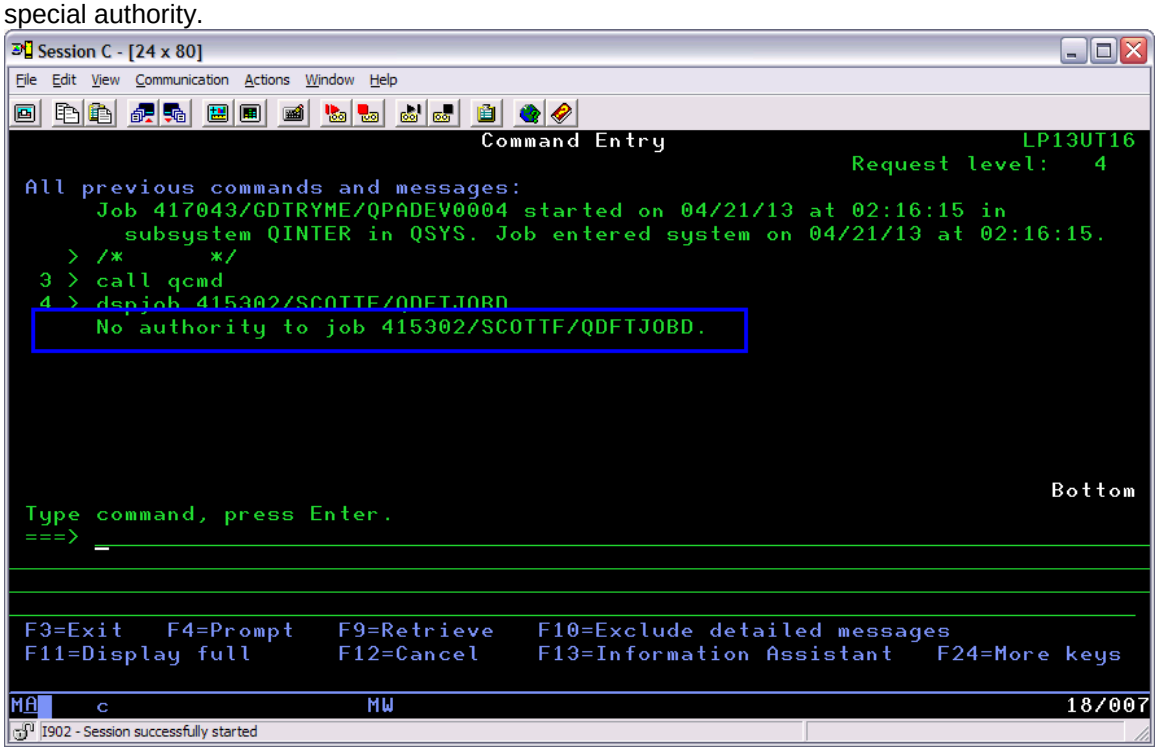


Highlight and copy the qualified jobname.

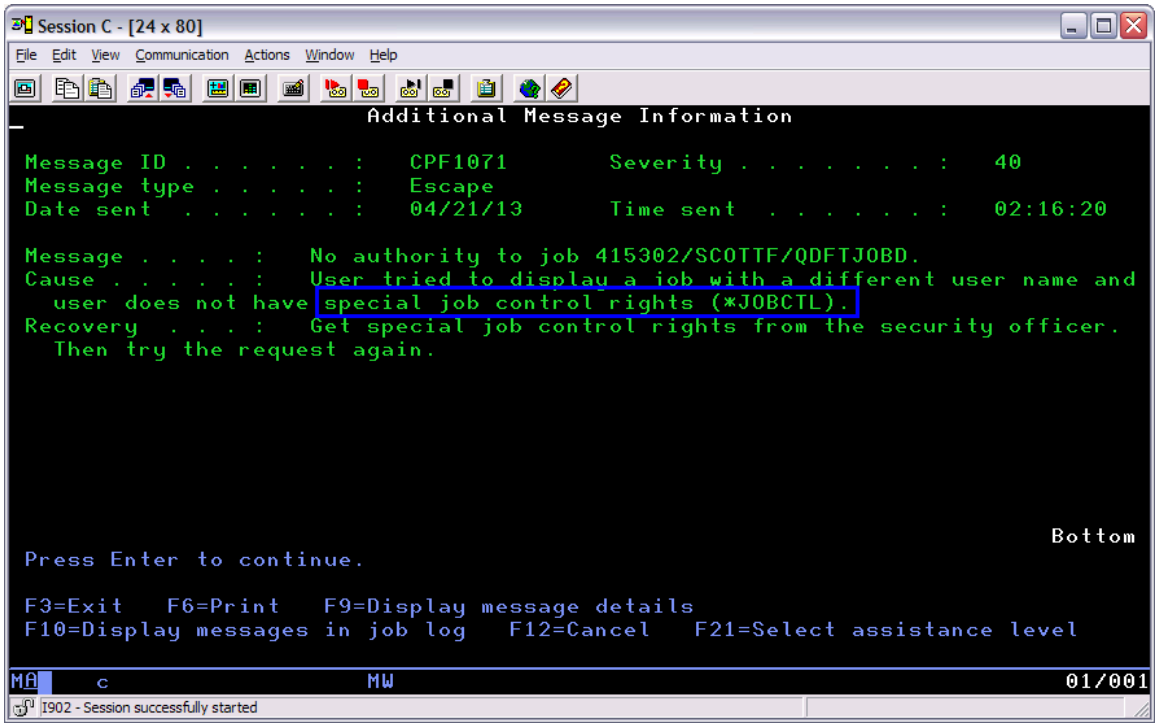


> **dspjob 415302/SCOTTF/QDFTJOBD**

If you see an authority failure, it means the user you have hasn't been granted *JOBCTL user special authority.



If you place the cursor on the authority failure message and press F1, you'll see the extended message details, as shown in the screenshot below.

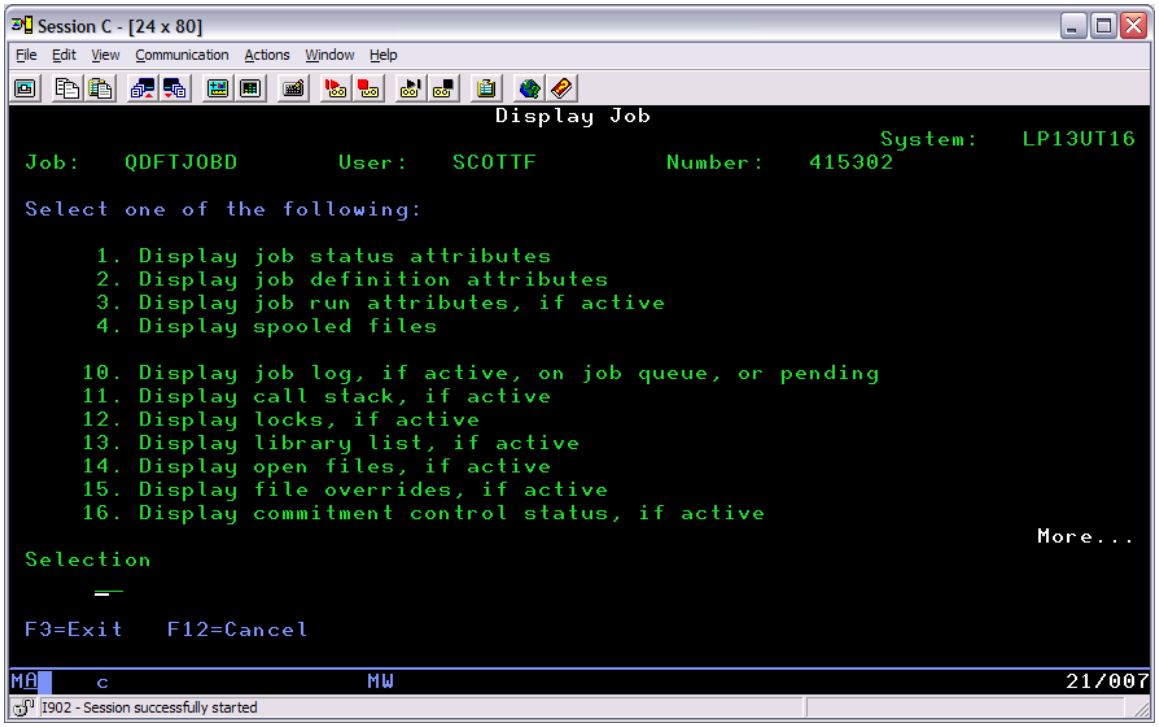


This is the command that could be used to grant the required authority:
CHGUSRPRF USRPRF(<username>) SPCAUT(*JOBCTL)

Only the security officer (someone with *SECADM authority) can hand out authority. The important thing to learn here is:

1) F10 allows me to see the failure messages
2) F1 on any message allows me to see extended detail on the message
3) Authority roadblocks probably can't be solved quickly

Once the DSPJOB command is working for you, you'll see the screen shown below. Options 10, 11 and 20 are the most useful for Guardium debug; however, the information you observe will not be fully explained in this document.



Tip: F9 is another extremely useful key to use from the command line of when you're within STRSQL. F9 retrieves the previous command. Every time you press F9, the previous command is retrieved. Keep pressing F9 to find earlier commands. Use F9 to avoid having to rekey commands.

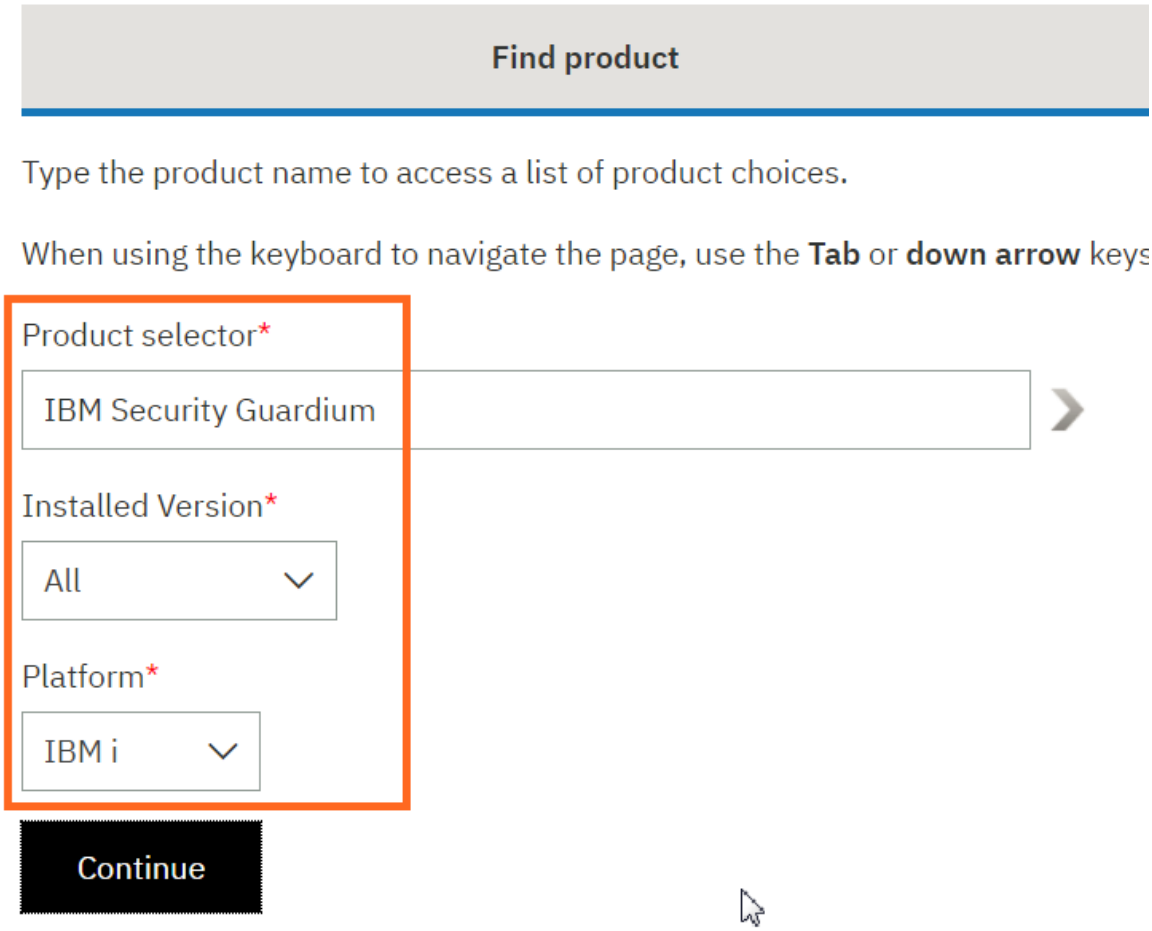
iS-TAP Service level requirements

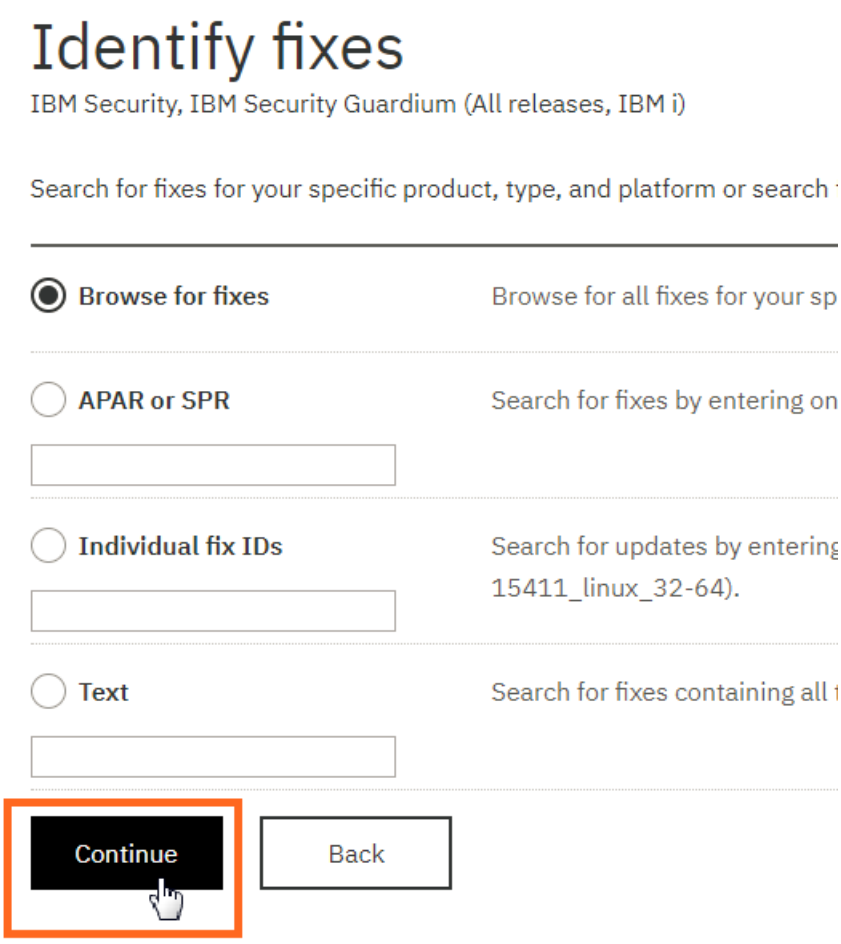
The Guardium on IBM i fact page (https://ibm.biz/GuardiumDAMonIBMi) is the single, best place to look to understand the IBM i service level requirements.

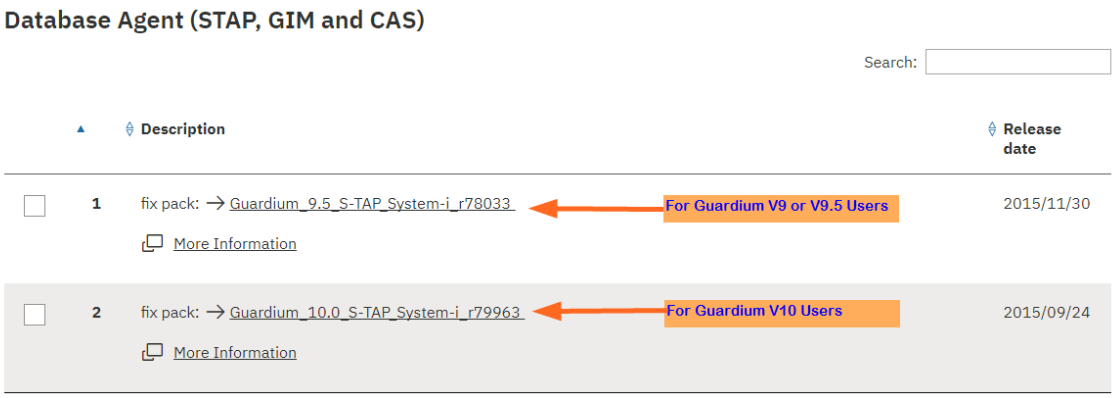
The latest S-TAP for System i (PASE program) service can be found here, using the "Find product" search facility. Download and install the software

http://www-933.ibm.com/support/fixcentral/

The three images below show how to navigate to the S-TAP download.
Choose the download that matches your Guardium version.


Find product
Type the product name to access a list of product choices.
When using the keyboard to navigate the page, use the Tab or down arrow keys
Product selector*
IBM Security Guardium
Installed Version*
All
Platform*
IBM i
Continue



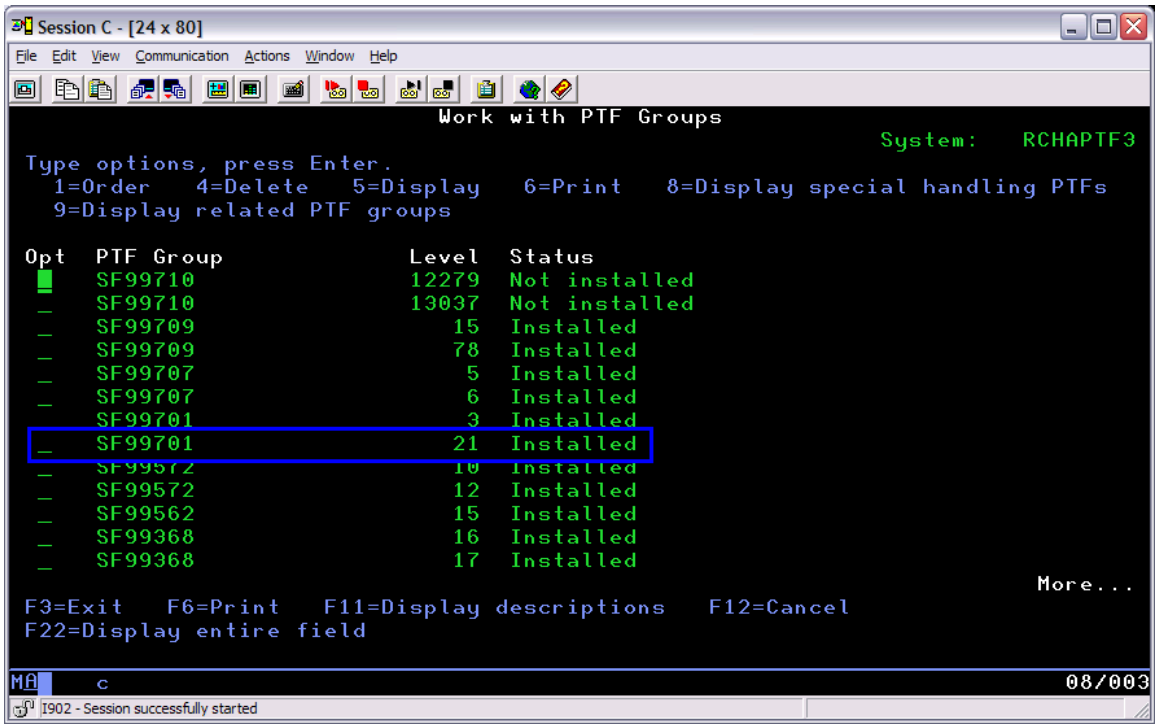


Db2 for i Service Level:

The Guardium on IBM i fact page (https://ibm.biz/GuardiumDAMonIBMi) lists the recommended Db2 PTF Group Service Level.

The Db2 PTF Group installed level can be examined using the WRKPTFGRP command.

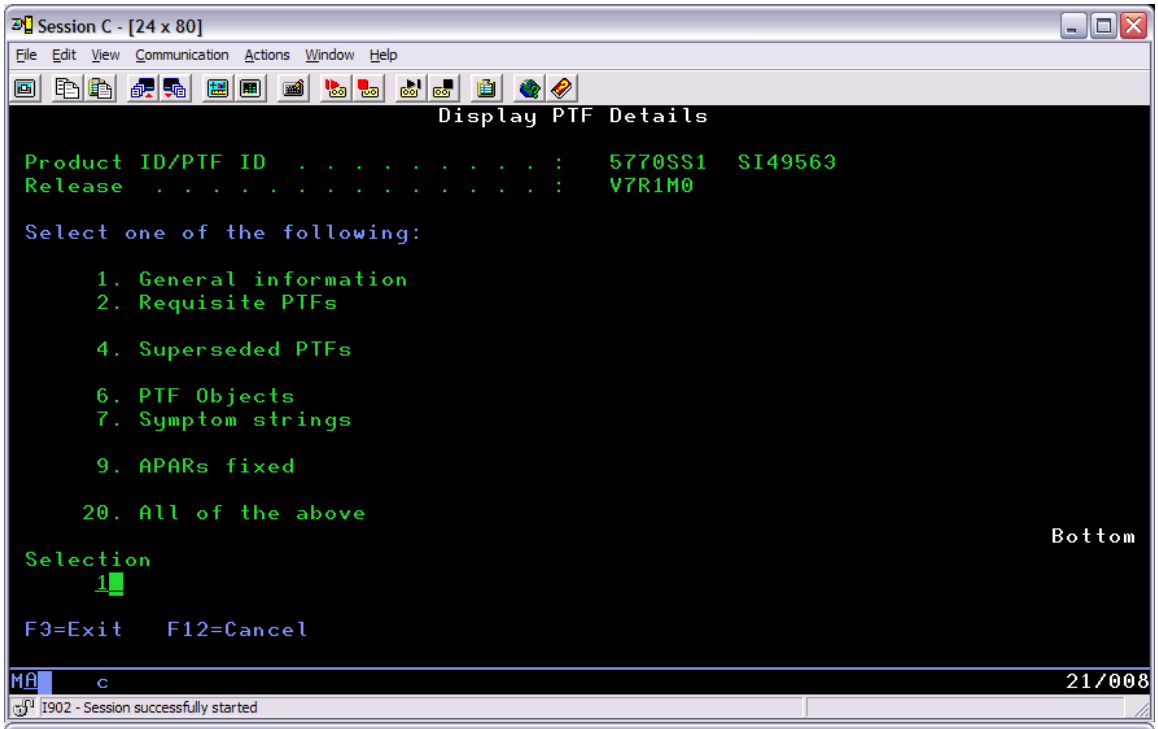
On IBM i 6.1, the Db2 PTF Group identifier is SF99601.
On IBM i 7.1, the Db2 PTF Group identifier is SF99701.
On IBM i 7.2, the Db2 PTF Group identifier is SF99702.
On IBM i 7.3, the Db2 PTF Group identifier is SF99703.
An example appears below. From this command, you will also under which IBM i operating system release is being used.

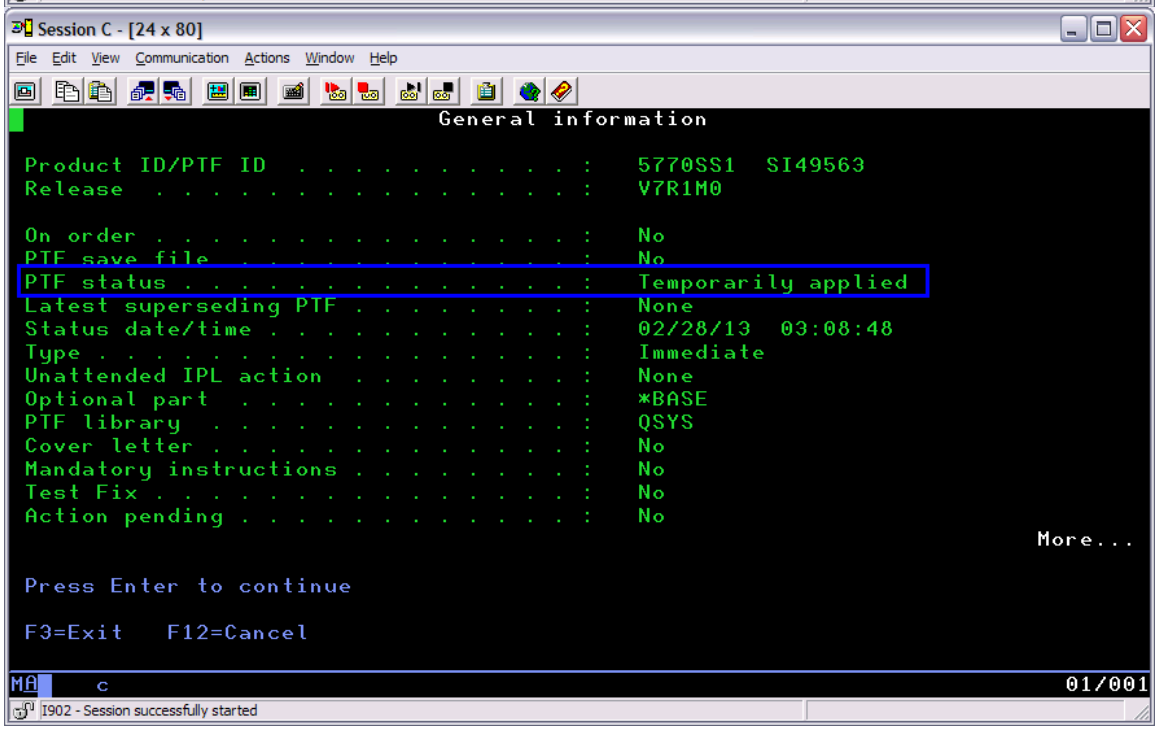


Group PTF checking handles the basic service level. In some situations, individual PTFs may be needed. To check whether the customer has an individual PTF installed, use the DSPPTF command.

For example:
DSPPTF LICPGM(5770SS1) SELECT(SI49563)

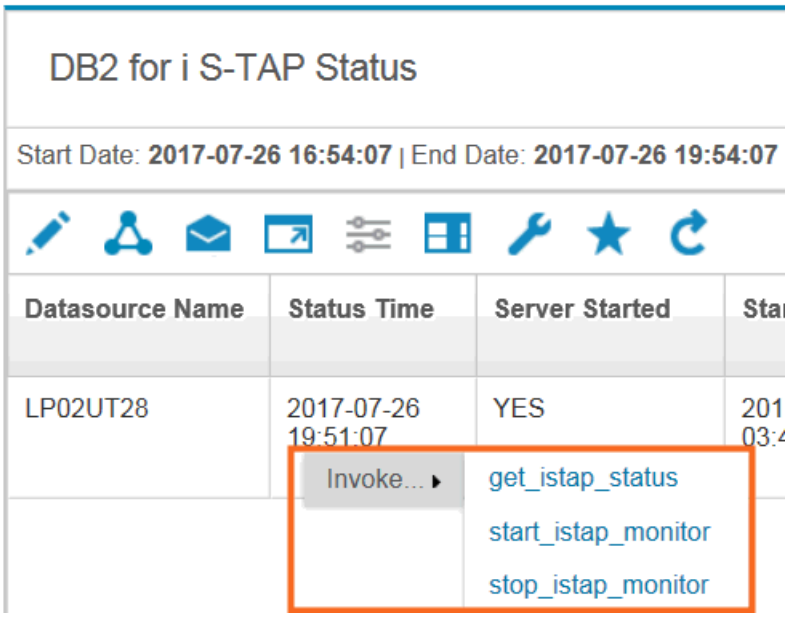
If this PTF were not loaded, the DSPPTF command would fail. If you see the first screen shown below appear, choose option 1 to observe the state of the PTF. If it lists a status of TEMPORARILY APPLIED, PERMANENTLY APPLIED or SUPERCEDED, the PTF in question is installed.


Session C - [24 x 80]
File Edit View Communication Actions Window Help
Display PTF Details
Product ID/PTF ID . . . . . : 5770SS1 SI49563
Release . . . . . : V7R1M0
Select one of the following:
1. General information
2. Requisite PTFs
4. Superseded PTFs
6. PTF Objects
7. Symptom strings
9. APARs fixed
20. All of the above
Bottom
Selection
1
F3=Exit F12=Cancel
MA c 21/008
I902 - Session successfully started

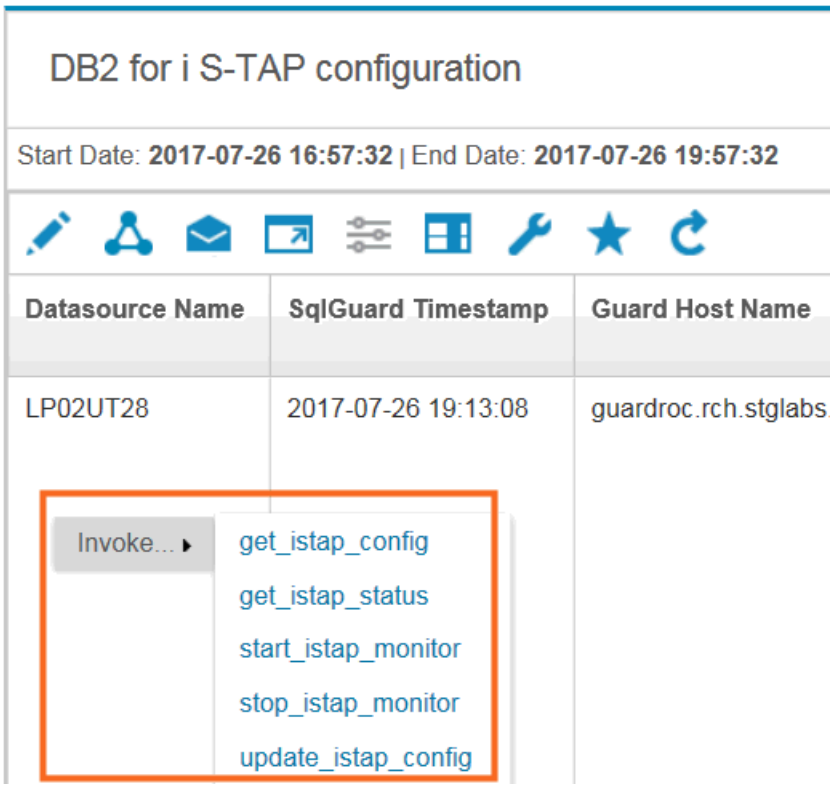

Session C - [24 x 80]
File Edit View Communication Actions Window Help
General information
Product ID/PTF ID . . . . . : 5770SS1 SI49563
Release . . . . . : V7R1M0
On order . . . . . : No
PTF save file . . . . . : No
PTF status . . . . . : Temporarily applied
Latest superseding PTF . . . . . : None
Status date/time . . . . . : 02/28/13 03:08:48
Type . . . . . : Immediate
Unattended IPL action . . . . . : None
Optional part . . . . . : *BASE
PTF library . . . . . : QSYS
Cover letter . . . . . : No
Mandatory instructions . . . . . : No
Test Fix . . . . . : No
Action pending . . . . . : No
More...
Press Enter to continue
F3=Exit F12=Cancel
MA c 01/001
I902 - Session successfully started

IBM i Authorization requirements for using Guardium to manage the iSTAP

DB2 for i S-TAP Status	
Action	**IBM i Authorization Requirements**
get_istap_status	None
start_istap_monitor	*EXECUTE system authority on QSYS/QDBSSUDF *SRVPGM **and** *JOBCTL special authority or QIBM_DB_SQLADM function usage
stop_istap_monitor	*EXECUTE system authority on QSYS/QDBSSUDF *SRVPGM **and** *JOBCTL special authority or QIBM_DB_SQLADM function usage



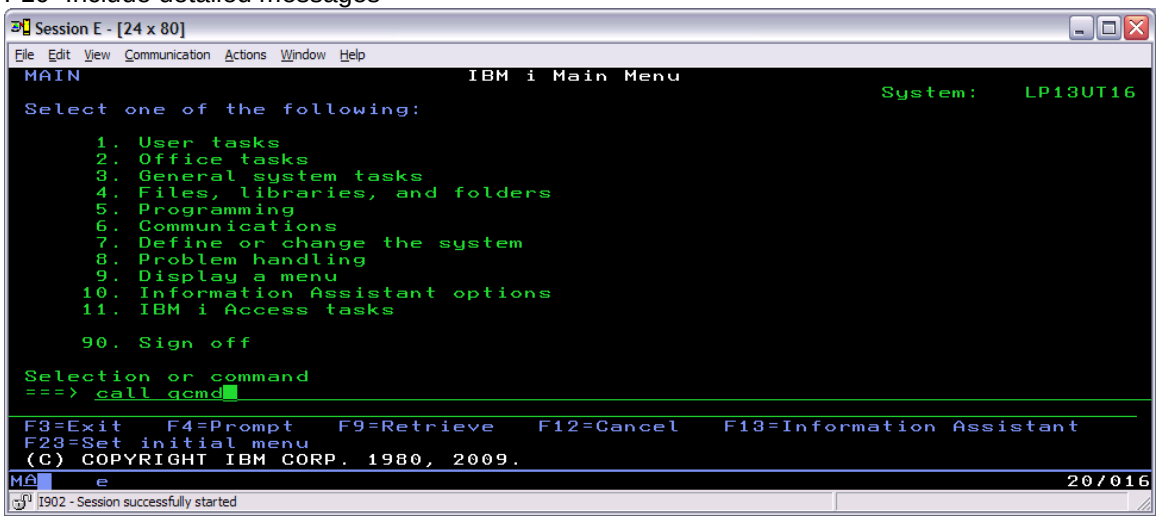
DB2 for i S-TAP Configuration	
Action	**IBM i Authorization Requirements**
get_istap_config	None
get_istap_status	None
start_istap_monitor	*EXECUTE system authority on QSYS/QDBSSUDF *SRVPGM **and** *JOBCTL special authority or QIBM_DB_SQLADM function usage
stop_istap_monitor	*EXECUTE system authority on QSYS/QDBSSUDF *SRVPGM **and** *JOBCTL special authority or QIBM_DB_SQLADM function usage
update_istap_config	*EXECUTE system authority on QSYS/QDBSSUDF *SRVPGM and EXECUTE privilege on procedure SYSPROC.SYSAUDIT_START_BATCH **and** *JOBCTL special authority or QIBM_DB_SQLADM function usage **and** ALL privilege on QSYS2.SYSAUDIT *FILE **and** ALTER privilege on QSYS2.SYSAUDMONT *FILE



Audit Server Status

If you don't see a command line, enter **CALL QCMD** (as shown below).

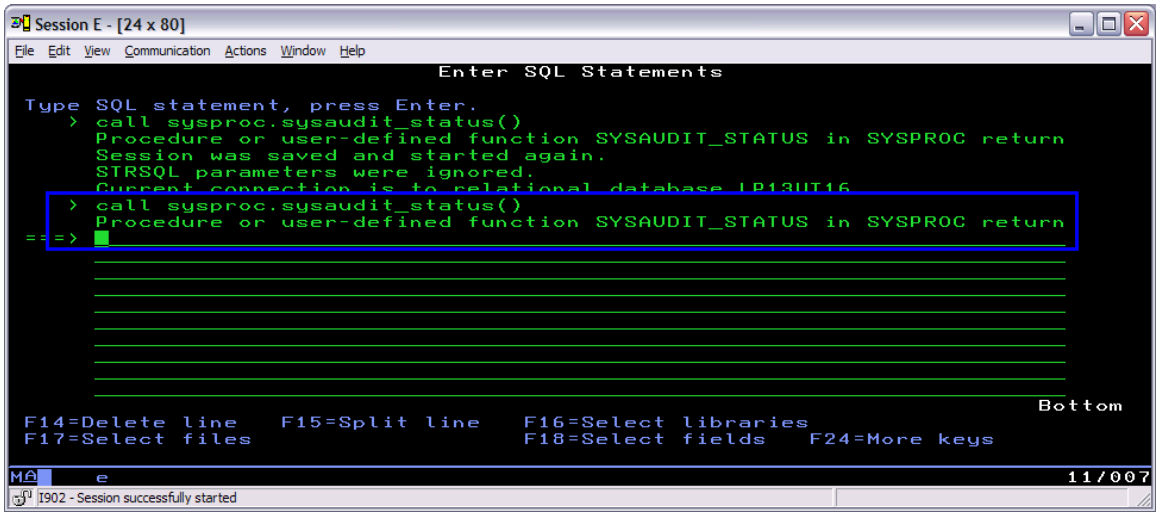
I always press PF10 to see more detail.
F10=Include detailed messages



Enter the Start SQL command: **STRSQL**

Enter the call to the status checking procedure: **call sysproc/sysaudit_status()**

If you see the message “Procedure or user-defined function SYSAUDIT_STATUS in SYSPROC return” (as shown below), the status procedure successfully collected status detail and the results can be examined.

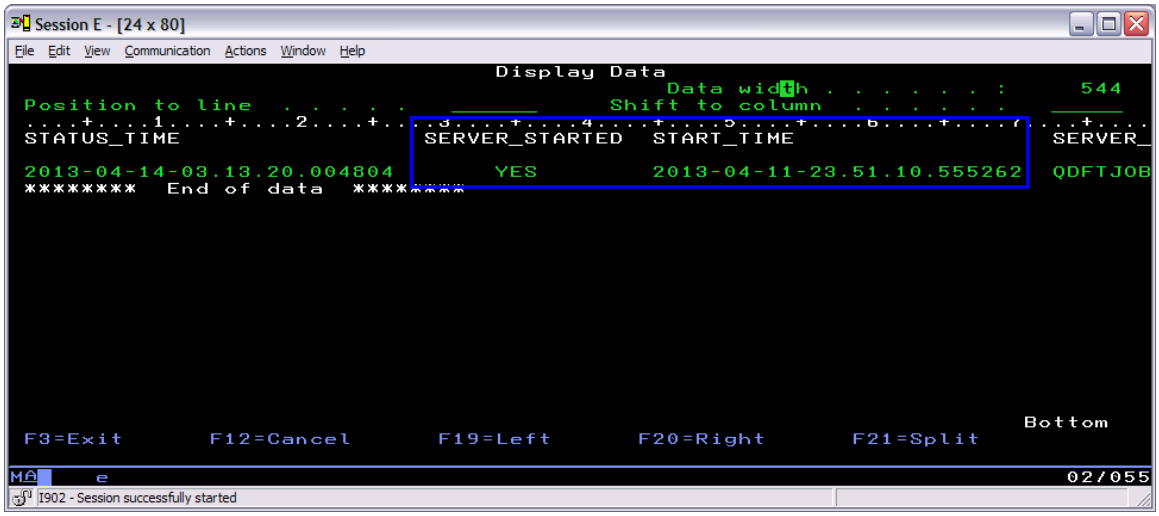


When working on customer problems, it’s common to call this procedure many times. Every time the procedure is called, a single row is inserted into the QTEMP/SYSAUDSTS *FILE. If you signoff and signon, you start over with a fresh *FILE with no rows. If you remain signed on, new rows will be added. When observing the query output, be mindful to look at the **STATUS_TIME** column to understand the time sequence of the status rows.

The following query will display the status routine output:

select * from qtemp/sysaudsts

The first screen you'll see (shown below) will reveal a lot. This screen shows whether the audit server is started and if it was started, when it was started.



The status file has many columns, so it might be beneficial to only query those column names that are of interest.

The QTEMP/SYSAUDSTS status detail contains different categories of status.

General status detail:

STATUS_TIME TIMESTAMP – Time that the sysproc/sysaudit_status() procedure was called to output this row.

SERVER_STARTED CHAR(4) - YES or NO

START_TIME TIMESTAMP – When the server was started (or restarted after a system IPL)

SERVER_JOB CHAR(26) – The jobname of the audit server job.
Note that the format of this column does not match the formatting needed when you work with IBM i commands.

For example: QDFTJOBD RUIYU 410890
Refers to the qualified jobname: 410890/RUIYU/QDFTJOBD

SQL Monitor detail:
This detail does not include the SQL statements that were filtered at the source through the use of one or more database monitor (STRDBMON) filters.

NUMBER_JOBS_AUDITED_USING_SQL BIGINT – The count of the number of different jobs that have sent at least one instance of audit detail to the PASE program.

NUMBER_PROCESSED_SQL_STATEMENTS BIGINT - The count of the number of SQL statements that have been received by the Instead of Trigger program. (QSQGDIOT)

NUMBER_ENQUEUED_SQL_STATEMENTS BIGINT - The count of the number of SQL statements that have been sent to the PASE program by the Instead of Trigger program. (QSQGDIOT)

NUMBER_SKIPPED_SQL_STATEMENTS BIGINT – Indicates the number of SQL statements that could have been sent to the PASE program, but have not been sent. Under normal conditions this value will be zero. When PREVENT_SKIPPED_ENTRIES is set to 'N', each job will attempt to an SQL statement on the queue up to three times then will give up (typically because queue is full).

NUMBER_PROCESSED_VARIABLE_SETS BIGINT – The total number of 3010 variable sets received by the Instead of Trigger program. (QSQGDIOT) The variable sets are the data needed to populate the "Bind Variables Values" column on the Guardium client.

NUMBER_SKIPPED_VARIABLE_SETS BIGINT - The total number of variable sets received, but could not be handed off to the Audit Server. Under normal conditions this value will be zero. When PREVENT_SKIPPED_ENTRIES is set to 'N', we keep a buffer of up to 300 variable sets before we begin to skip variable sets.

QAUDJRN (security journal) detail:
This information does not include the audit entry filtering based upon the configured audit entry types.

NUMBER_PROCESSED_QAUDJRN_ENTRIES BIGINT – Number of individual audit entries received from the QAUDJRN audit journal.

NUMBER_ENQUEUED_QAUDJRN_ENTRIES BIGINT – Number of audit entries sent to the PASE program. Does not include any audit entries received (processed), but deemed not necessary to send to the Guardium collector

NUMBER_SKIPPED_QAUDJRN_ENTRIES BIGINT – Number of audit entries which could not be sent to the Audit Server (PASE program). Under normal conditions this value will be zero. Does not include any audit entries received (processed), but deemed not necessary to send to the Guardium collector. When PREVENT_SKIPPED_ENTRIES is set to 'N', audit entries can be discarded if the Audit Server is unable to receive the detail.

QUEUE detail:
The queue referred to here is the message queue being used to communicate the auditable entries between the server and the PASE program that sends the detail to the Guardium Collector.

QUEUE_DAMAGED CHAR(3) – YES or NO. If YES, level 3 should be contacted.

NUMBER_MESSAGES_ON_QUEUE INTEGER – The number of messages that the Guardium PASE program has NOT consumed. Under normal situations, the number of messages on the queue will be zero indicating that the PASE program is able to keep up with the audit data.

SIZE_OF_MESSAGES_ON_QUEUE INTEGER – Similar to NUMBER_MESSAGES_ON_QUEUE, but a different metric. Frequently zero.

MAXIMUM_SIZE_OF_QUEUE INTEGER – Not configurable and you should see 16,777,216.

TOTAL_ENQUEUING_THREADS INTEGER – Indicates how many different threads are placing audit detail into the message queue.

LAST_DEQUEUE_TIME TIMESTAMP – Indicator that the queue is working.

LAST_ENQUEUE_TIME TIMESTAMP - Indicator that the queue is working.

QUEUE_OWNER CHAR(10) – Not interesting

Monitor end detail:

LAST_END_MONITOR_JOB CHAR(26) – The jobname of the previous instance of the audit server.

LAST_END_MONITOR_USER CHAR(10) – The user name of the user that ended the audit server. When the customer IPLs the machine, the audit server will be stopped and automatically restarted. The user ID that started the audit server will appear here on an IPL.

This information is more useful in the cases where someone has manually ended the audit server. This detail allows the auditor to understand that the server was ended and by whom.

Examining the Audit Server Configuration

The settings in these fields can be established from the Guardium web client by setting up and using the Db2 for i S-TAP configuration report.

SELECT * FROM QSYS2/SYSAUDIT

SERVERNAME VARCHAR(128) – The IP address of the Guardium collector.
You should be able to ping the collector from the IBM i.

Example of good PING results:

> PING RMTSYS('9.5.39.189')

Verifying connection to host system 9.5.39.189.
PING reply 1 from 9.5.39.189 took 12 ms. 256 bytes. TTL 64.
PING reply 2 from 9.5.39.189 took 0 ms. 256 bytes. TTL 64.
PING reply 3 from 9.5.39.189 took 0 ms. 256 bytes. TTL 64.
PING reply 4 from 9.5.39.189 took 0 ms. 256 bytes. TTL 64.
PING reply 5 from 9.5.39.189 took 0 ms. 256 bytes. TTL 64.
Round-trip (in milliseconds) min/avg/max = 0/2/12.
Connection verification statistics: 5 of 5 successful (100 %).

FILTER_USER VARCHAR(110) – Explained in the white paper.

FILTER_JOB VARCHAR(28) – Explained in the white paper.
FILTER_TCPIP VARCHAR(254) – Explained in the white paper.
FILTER_TABLE VARCHAR(5170) – Explained in the white paper.
FILTER_PORT INTEGER – Explained in the white paper.
FILTER_CLIENT_ACCTING VARCHAR(128) – Explained in the white paper.
FILTER_CLIENT_APPLNAME VARCHAR(128) – Explained in the white paper.
FILTER_CLIENT_PROGRAMID VARCHAR(128) – Explained in the white paper.
FILTER_CLIENT_USERID VARCHAR(128) – Explained in the white paper.
FILTER_CLIENT_WRKSTNNAME VARCHAR(128) – Explained in the white paper.
FILTER_RDB VARCHAR(1290) – Explained in the white paper.
Note: Filter_RDB is a case sensitive filter. Execute this command on your target database and enter the name exactly as it is returned.
STRSQL
> VALUES(CURRENT SERVER)
FILTER_SYSTEM_SQL CHAR(1) – Explained in the white paper.
FILTER_AUDIT_ENTRY_TYPES VARCHAR(1000) – Explained in the white paper.
ITAP_PARAM VARCHAR(1000) – Internal information
START_JOB CHAR(26) – The jobname of the Audit Server job.
START_TIME TIMESTAMP – The timestamp when the Audit Server job was started.
MONITOR_ID CHAR(10) – The database monitor internal identifier used by Guardium.
START_USER CHAR(10) – Explained in the white paper.
DEBUG CHAR(1) - Internal information, defaults to 'N'
PREVENT_SKIPPED_ENTRIES CHAR(1) – Directs the SQL auditing to handle the case where the audit server job is overwhelmed with detail. When setting this control to 'Y', the audit server is given preference over the performance of the work stream. Defaults to 'N'.

Ending the Audit Server

Note: We haven't externally documented this procedure because the customer should use the Guardium Web console to start and stop the audit server.

Authorization requirement:
*JOBCTL user special authority
Or
QIBM_DB_SQLADM function usage

STRSQL
CALL sysproc/sysaudit_end()

Or

RUNSQL SQL('CALL SYSPROC/SYSAUDIT_END ( )') COMMIT(*NONE) NAMING(*SYS)

When the server is ended, you should see this:

```
> RUNSQL SQL('CALL SYSPROC/SYSAUDIT_END ( )') COMMIT(*NONE) NAMING(*SYS)
  ENDJOB started for job 410890/RUIYU/QDFTJOBD.
```

Starting the Audit Server

Note: We haven't externally documented this procedure because the customer should use the Guardium client to start and stop the audit server. (You would do this by invoking the Db2 for i start_istap_monitor API from the Db2 for i status report or from the CLI.)

Its better to use **sysproc/sysaudit_start_batch()** because sysaudit_start() will not return control to the caller because the audit server will be running in that job.

Authorization requirement:
*JOBCTL user special authority
Or
QIBM_DB_SQLADM function usage

STRSQL
call sysproc/sysaudit_start('')

Or

RUNSQL SQL('CALL SYSPROC/SYSAUDIT_START('''')') COMMIT(*NONE) NAMING(*SYS)

Recyling the Audit Server

Note: We haven't externally documented this procedure because the customer should use the Guardium client to start and stop the audit server by using the start_istap_monitor and stop_istap_monitor APIs)

When a change is made to the audit server configuration, you normally have to stop and start (recycle) the server to allow the configuration changes to be used. This procedure ends the audit server and restarts it.

Authorization requirement:
*JOBCTL user special authority
Or
QIBM_DB_SQLADM function usage

STRSQL
call sysproc/sysaudit_start_batch('')

Or

RUNSQL SQL('CALL SYSPROC/SYSAUDIT_START_BATCH('''')') COMMIT(*NONE) NAMING(*SYS)

When the server is recycled, you should see this:

```
RUNSQL SQL('CALL SYSPROC/SYSAUDIT_START_BATCH('''')') COMMIT(*NONE) NAMIN
G(*SYS)
Output file A created in library QTEMP.
Member A added to output file A in library QTEMP.
Job 412123/RUIYU/QDFTJOBD submitted to job queue QBATCH in library QGPL.
```

If the audit server fails to start, example the failing joblog using
WRKSPLF <user-name found in the START_USER column in the QSYS2.SYSAUDIT file>
You should see messages which indicate WHY the audit server is not running.

Reasons why the audit server would fail to start:

1. If the QSYS2.SYSAUDIT file contains an invalid value in any of the FILTER_xxxxx columns.
   Remediation: Correct the FILTER value using the UPDATE SQL statement.
2. If the QSYS2.SYSAUDIT file is not journaled to QSYS2.QSQJRN.
   Remediation: Start journaling on the QSYS2.QSQJRN.
   Example: STRJRNPF QSYS2/SYSAUDIT QSYS2/QSQJRN
3. If the joblog contains the CPF1147 message (e.g. Job priority 2 exceeds limit 3 for user SCOTTF), use the subsequent remediation steps.

Determine which user profile is being used to run the iS-TAP

```
select START_USER
  from qsys2.sysaudit
  where START_USER is not null;
```

Query the maximum job priority allowed for the audit server user.
Note, replace GDUSER with the value returned from the previous query.

```
select HIGHEST_SCHEDULING_PRIORITY, u.* from
  qsys2.user_info u
  where AUTHORIZATION_NAME = 'GDUSER';
```

Use the CHGUSRPRF command to allow the use of a higher job priority.
CHGUSRPRF USRPRF(GDUSER) PTYLMT(2)

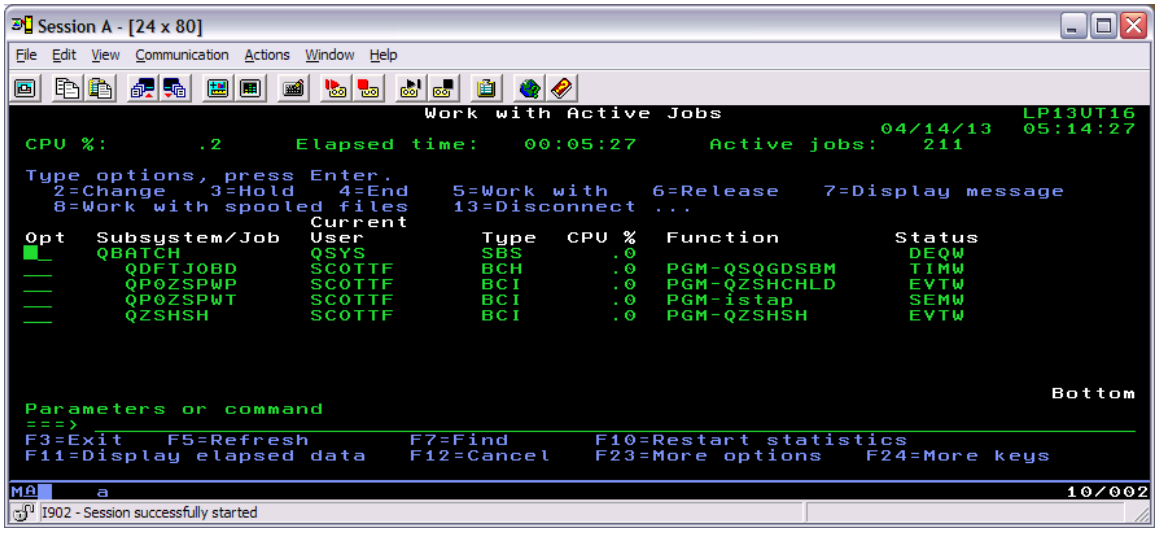
After any remediation steps, retry the starting of the iS-TAP audit server.

Examining the Audit Server

At some point, you may want to examine the actual jobs and see if it looks “normal”.

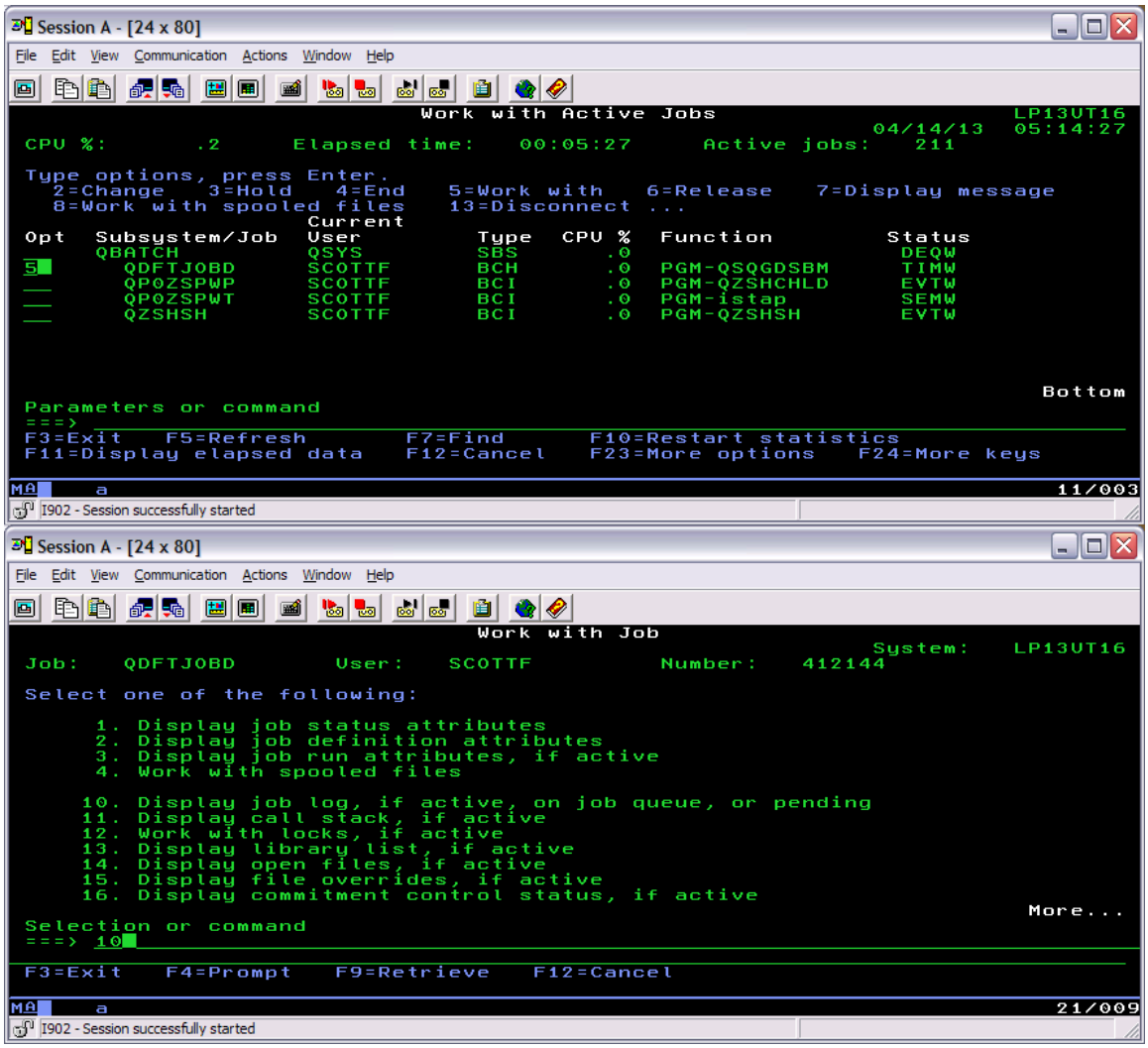
WRKACTJOB SBS(QBATCH) – The command that will display all the jobs running within the QBATCH subsystem. In the image below, we see the 4 jobs needed for normal Audit server processing.

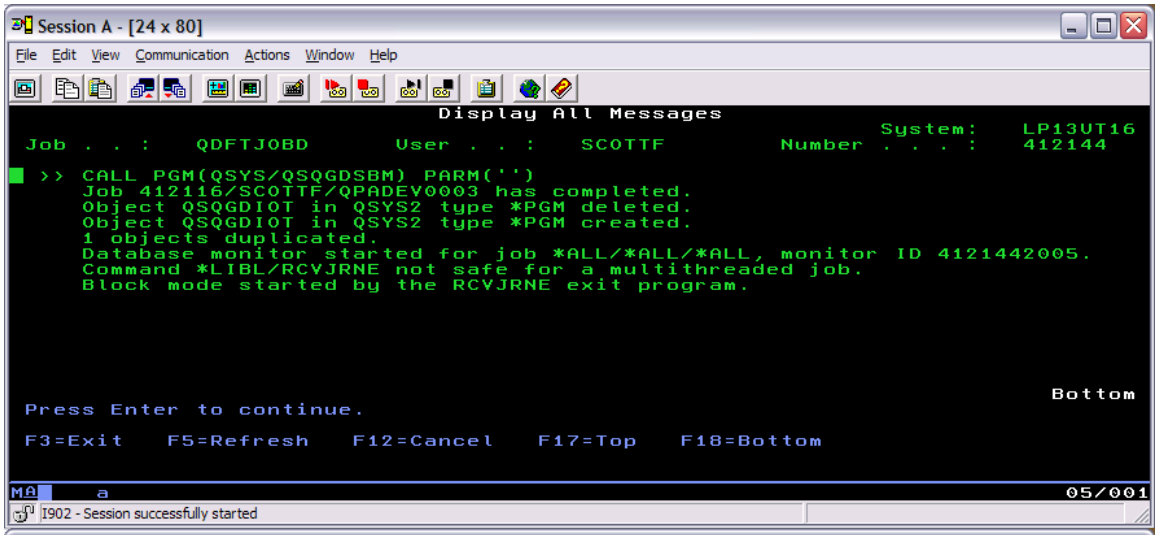
The status of the jobs in this image is also what you want to see. I’ve seen several cases where the PGM-istap job has Status = RUN and never leaves that state. That’s an indication of a problem state.

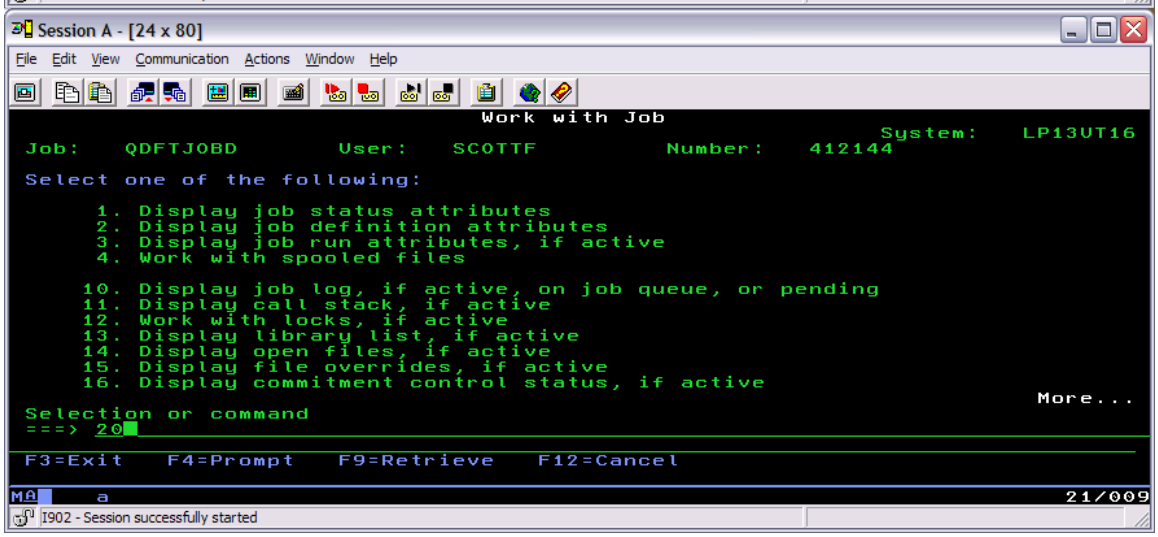


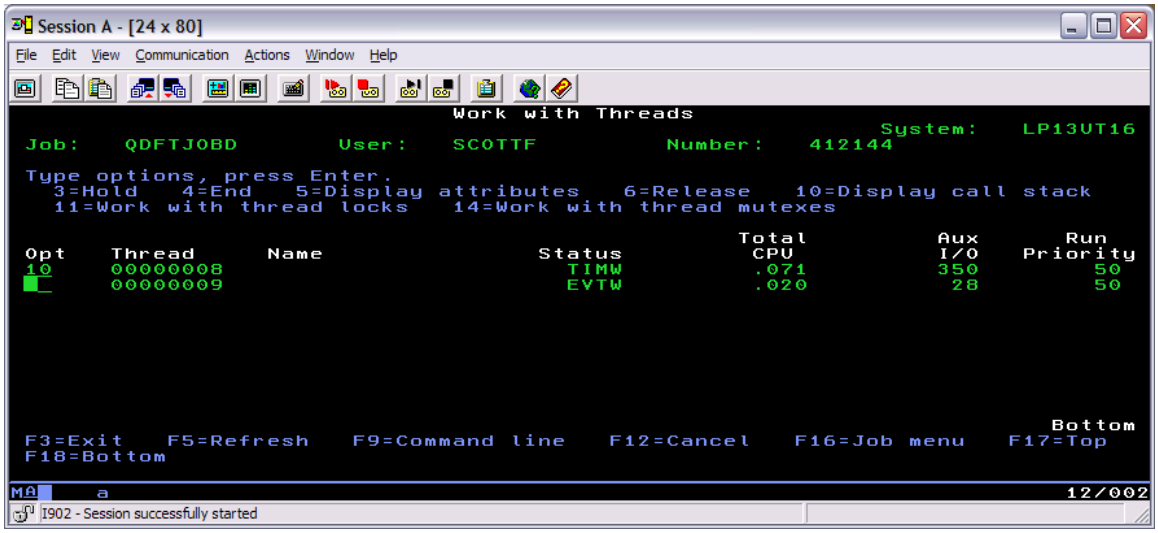
Let’s peruse 2 of the 4 jobs shown here; to observe additional symptoms of normal processing. The other two jobs are important, but they are not handling any of the audit server entry processing.

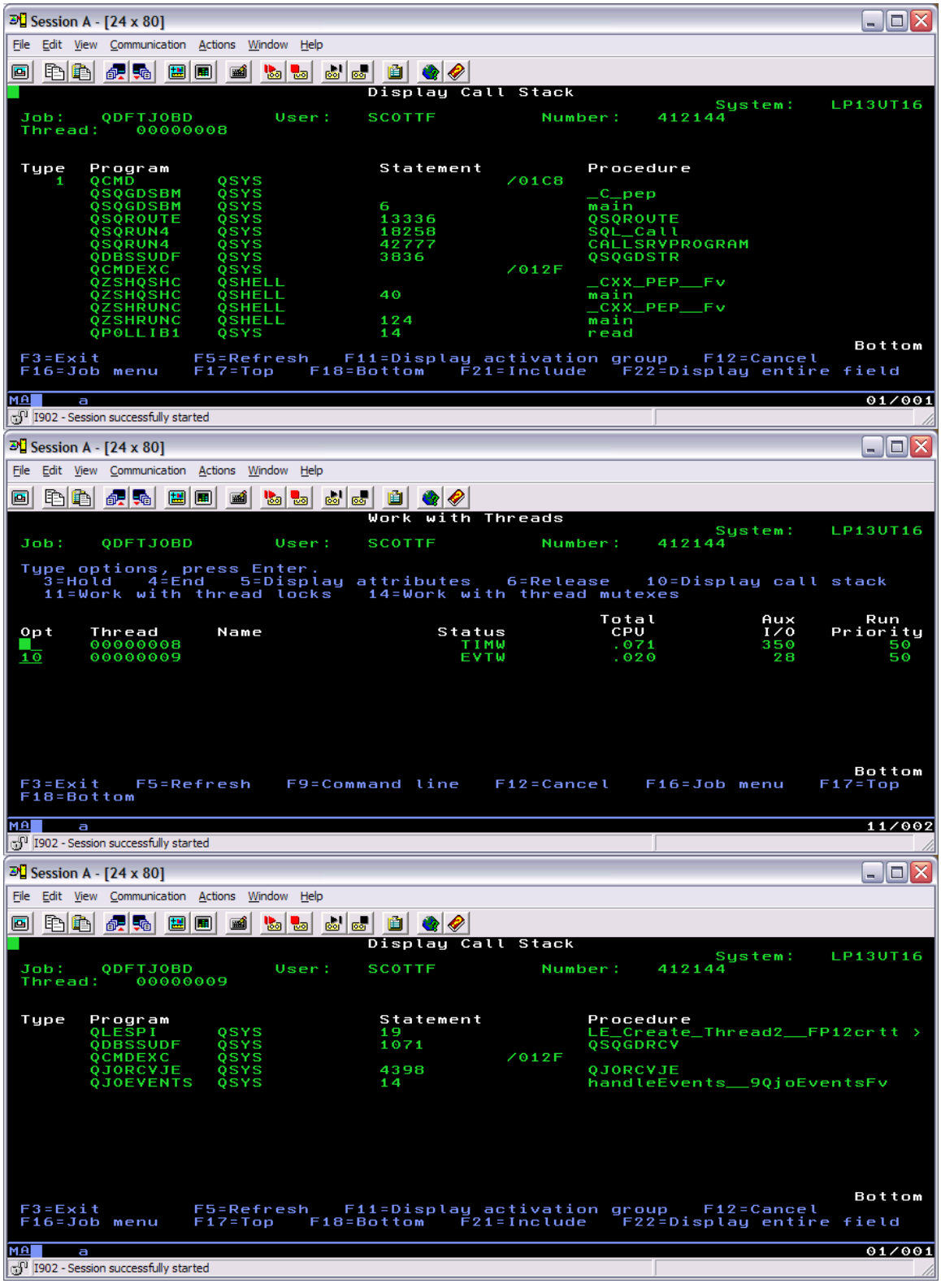
QDFTJOBD – QSQGDSBM job:



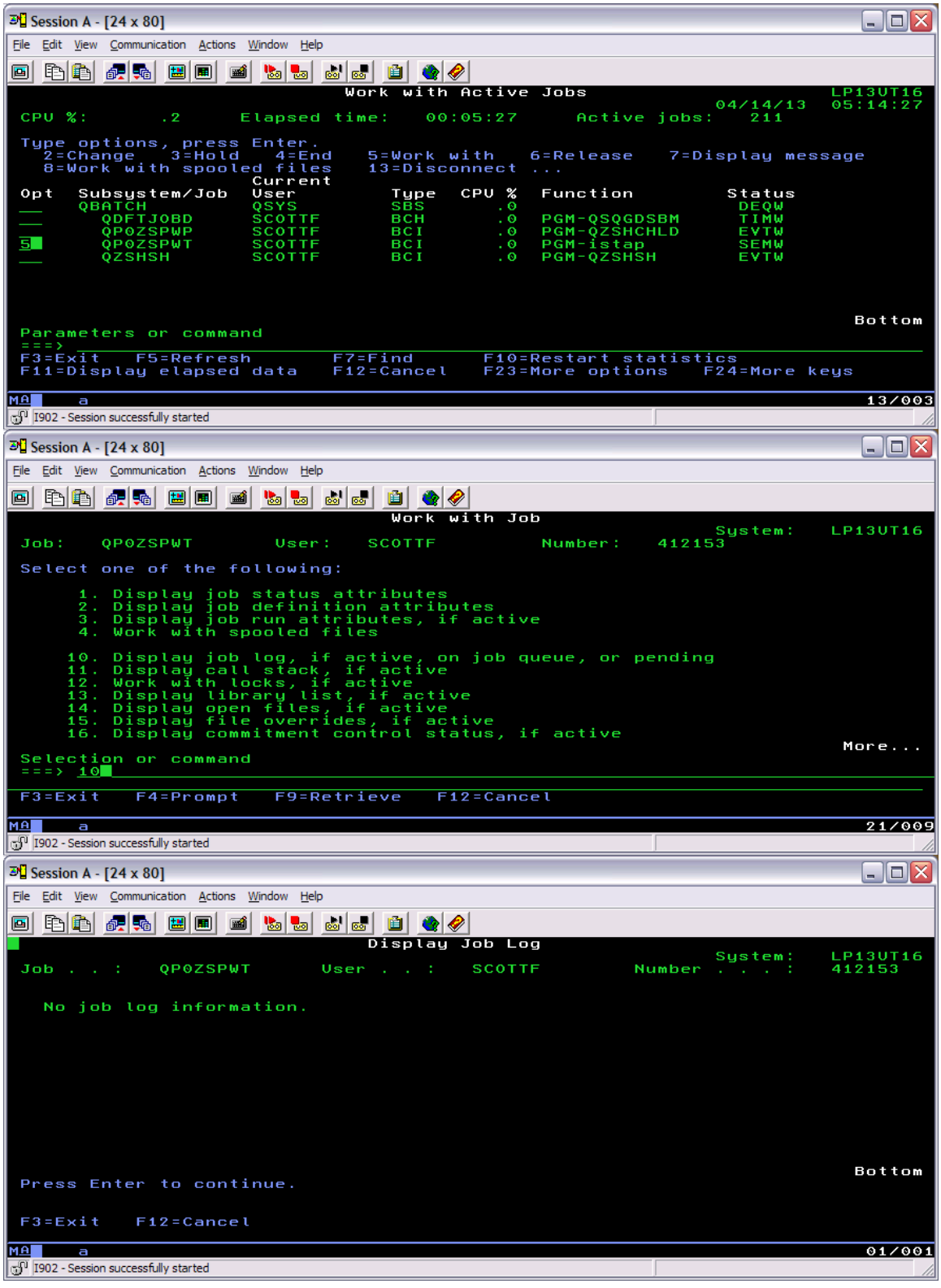

Session A - [24 x 80]
File Edit View Communication Actions Window Help
Display All Messages
System: LP13VT16
Job . . : QDFTJOBD User . . : SCOTTF Number . . . : 412144
>> CALL PGM(QSYS/QSQGDSBM) PARM('')
Job 412116/SCOTTF/QPADEV0003 has completed.
Object QSQGDIOT in QSYS2 type *PGM deleted.
Object QSQGDIOT in QSYS2 type *PGM created.
1 objects duplicated.
Database monitor started for job *ALL/*ALL/*ALL, monitor ID 4121442005.
Command *LIBL/RCVJRNE not safe for a multithreaded job.
Block mode started by the RCVJRNE exit program.
Bottom
Press Enter to continue.
F3=Exit F5=Refresh F12=Cancel F17=Top F18=Bottom
MA a 05/001
I902 - Session successfully started

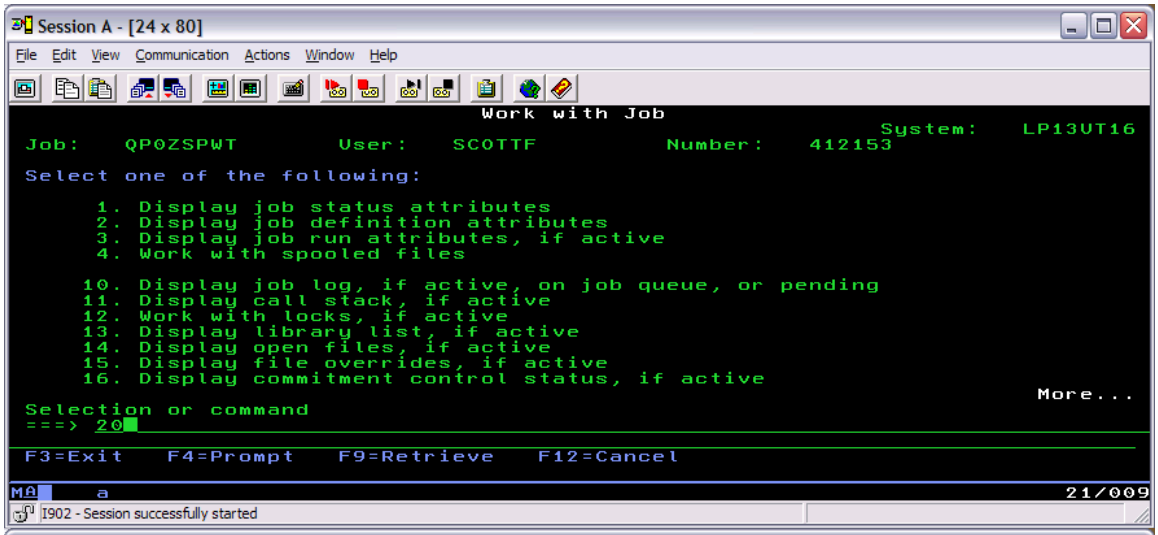

Session A - [24 x 80]
File Edit View Communication Actions Window Help
Work with Job
System: LP13VT16
Job: QDFTJOBD User: SCOTTF Number: 412144
Select one of the following:
1. Display job status attributes
2. Display job definition attributes
3. Display job run attributes, if active
4. Work with spooled files
10. Display job log, if active, on job queue, or pending
11. Display call stack, if active
12. Work with locks, if active
13. Display library list, if active
14. Display open files, if active
15. Display file overrides, if active
16. Display commitment control status, if active
More...
Selection or command
===> 20
F3=Exit F4=Prompt F9=Retrieve F12=Cancel
MA a 21/009
I902 - Session successfully started

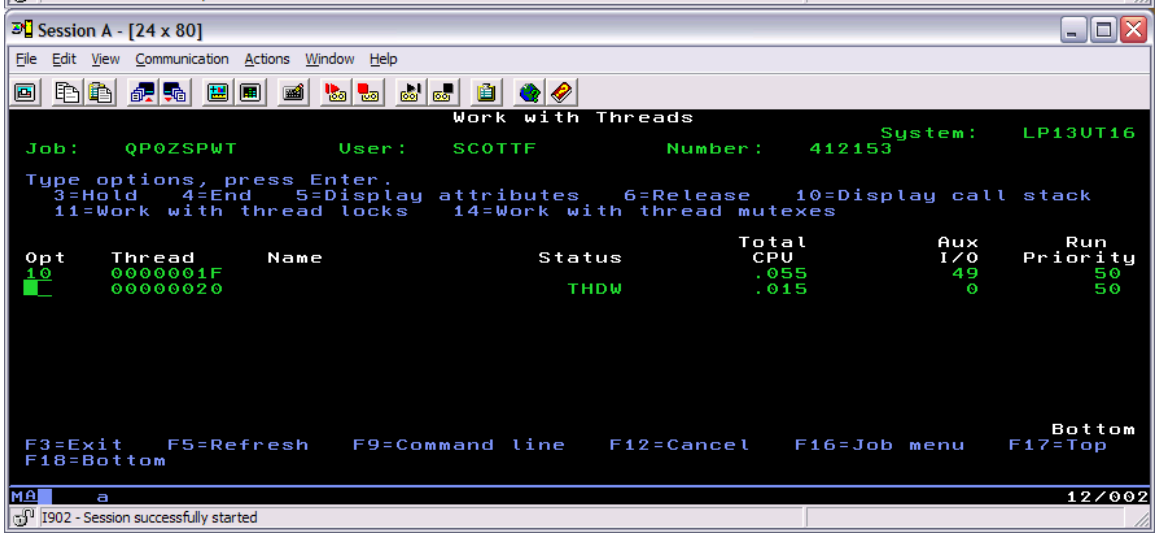

Session A - [24 x 80]
File Edit View Communication Actions Window Help
Work with Threads
System: LP13VT16
Job: QDFTJOBD User: SCOTTF Number: 412144
Type options, press Enter.
3=Hold 4=End 5=Display attributes 6=Release 10=Display call stack
11=Work with thread locks 14=Work with thread mutexes
Opt Thread Name Status Total CPU Aux I/O Run Priority
10 00000008 TIMW .071 350 50
00000009 EVTW .020 28 50
Bottom
F3=Exit F5=Refresh F9=Command line F12=Cancel F16=Job menu F17=Top
F18=Bottom
MA a 12/002
I902 - Session successfully started

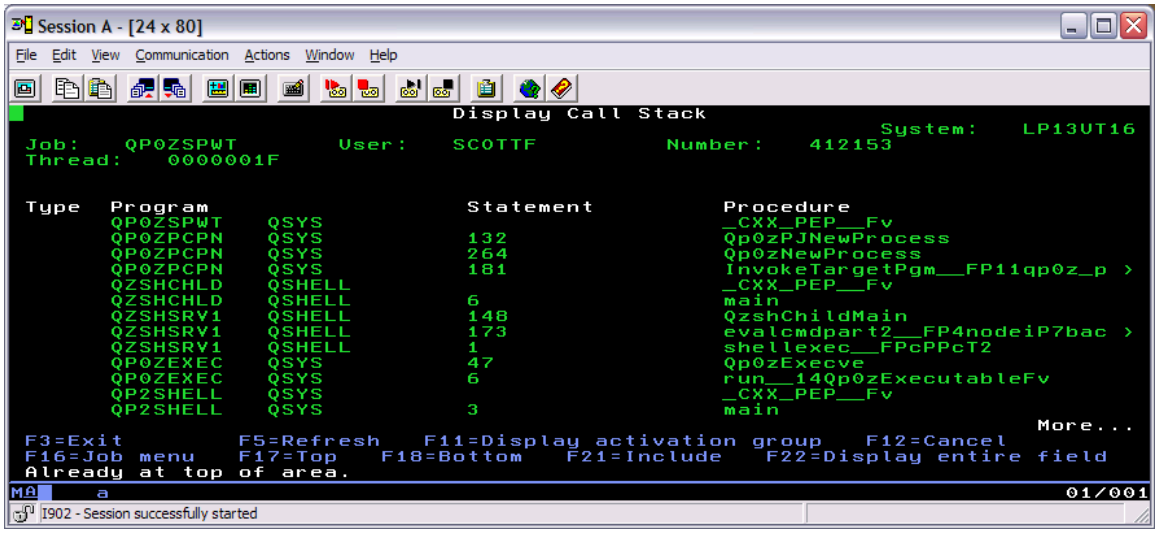


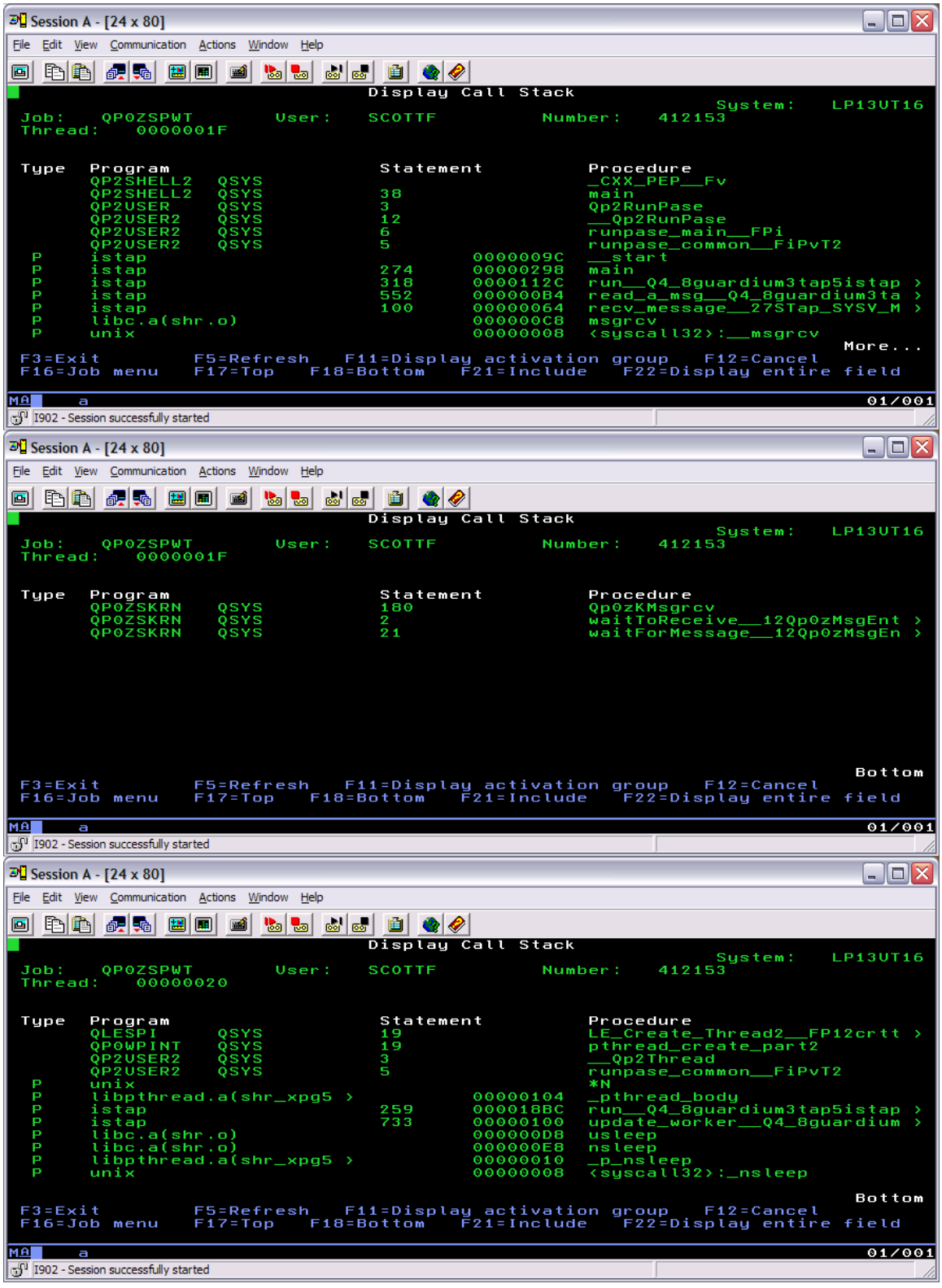
QP0ZSPWT – istap job:

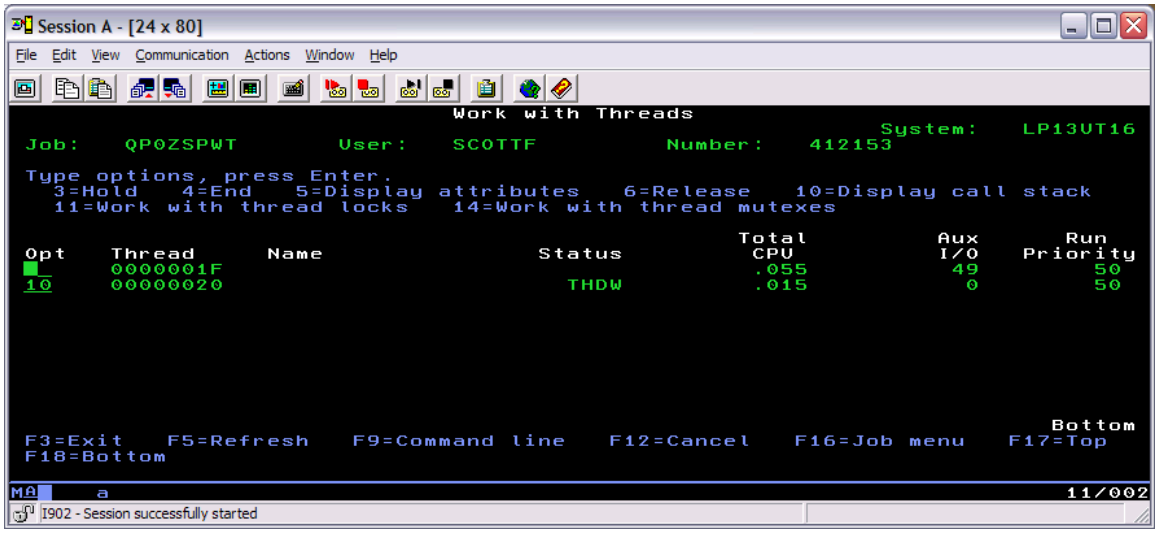

Session A - [24 x 80]
File Edit View Communication Actions Window Help
Work with Active Jobs
LP13UT16
04/14/13 05:14:27
CPU %: .2 Elapsed time: 00:05:27 Active jobs: 211
Type options, press Enter.
2=Change 3=Hold 4=End 5=Work with 6=Release 7=Display message
8=Work with spooled files 13=Disconnect ...
Opt Subsystem/Job Current User Type CPU % Function Status
QBATCH QSYS SBS .0 DEQW
QDFTJOBD SCOTTF BCH .0 PGM-QSQGDSBM TIMW
QP0ZSPWP SCOTTF BCI .0 PGM-QZSHCHLD EVTW
5 QP0ZSPWT SCOTTF BCI .0 PGM-istap SEMW
QZSHSH SCOTTF BCI .0 PGM-QZSHSH EVTW
Bottom
Parameters or command
===>
F3=Exit F5=Refresh F7=Find F10=Restart statistics
F11=Display elapsed data F12=Cancel F23=More options F24=More keys
MA a 13/003
I902 - Session successfully started
Session A - [24 x 80]
File Edit View Communication Actions Window Help
Work with Job
System: LP13UT16
Job: QP0ZSPWT User: SCOTTF Number: 412153
Select one of the following:
1. Display job status attributes
2. Display job definition attributes
3. Display job run attributes, if active
4. Work with spooled files
10. Display job log, if active, on job queue, or pending
11. Display call stack, if active
12. Work with locks, if active
13. Display library list, if active
14. Display open files, if active
15. Display file overrides, if active
16. Display commitment control status, if active
More...
Selection or command
===> 10
F3=Exit F4=Prompt F9=Retrieve F12=Cancel
MA a 21/009
I902 - Session successfully started
Session A - [24 x 80]
File Edit View Communication Actions Window Help
Display Job Log
System: LP13UT16
Job . . : QP0ZSPWT User . . : SCOTTF Number . . . . : 412153
No job log information.
Bottom
Press Enter to continue.
F3=Exit F12=Cancel
MA a 01/001
I902 - Session successfully started


Session A - [24 x 80]
File Edit View Communication Actions Window Help
Work with Job
System: LP13VT16
Job: QP0ZSPWT User: SCOTTF Number: 412153
Select one of the following:
1. Display job status attributes
2. Display job definition attributes
3. Display job run attributes, if active
4. Work with spooled files
10. Display job log, if active, on job queue, or pending
11. Display call stack, if active
12. Work with locks, if active
13. Display library list, if active
14. Display open files, if active
15. Display file overrides, if active
16. Display commitment control status, if active
More...
Selection or command
===> 20
F3=Exit F4=Prompt F9=Retrieve F12=Cancel
MA a 21/009
I902 - Session successfully started


Session A - [24 x 80]
File Edit View Communication Actions Window Help
Work with Threads
System: LP13VT16
Job: QP0ZSPWT User: SCOTTF Number: 412153
Type options, press Enter.
3=Hold 4=End 5=Display attributes 6=Release 10=Display call stack
11=Work with thread locks 14=Work with thread mutexes
Opt Thread Name Status Total CPU Aux I/O Run Priority
10 0000001F .055 49 50
00000020 THDW .015 0 50
Bottom
F3=Exit F5=Refresh F9=Command line F12=Cancel F16=Job menu F17=Top
F18=Bottom
MA a 12/002
I902 - Session successfully started


Session A - [24 x 80]
File Edit View Communication Actions Window Help
Display Call Stack
System: LP13VT16
Job: QP0ZSPWT User: SCOTTF Number: 412153
Thread: 0000001F
Type Program Statement Procedure
QP0ZSPWT QSYS _CXX_PEP__Fv
QP0ZPCPN QSYS 132 Qp0zPJNewProcess
QP0ZPCPN QSYS 264 Qp0zNewProcess
QP0ZPCPN QSYS 181 InvokeTargetPgm__FP11qp0z_p
QZSHCHLD QSHELL _CXX_PEP__Fv
QZSHCHLD QSHELL 6 main
QZSHSRV1 QSHELL 148 QzshChildMain
QZSHSRV1 QSHELL 173 evalcmdpart2__FP4nodeiP7bac
QZSHSRV1 QSHELL 1 shellexec__FPcPPcT2
QP0ZEXEC QSYS 47 Qp0zExecve
QP0ZEXEC QSYS 6 run__14Qp0zExecutableFv
QP2SHELL QSYS _CXX_PEP__Fv
QP2SHELL QSYS 3 main
More...
F3=Exit F5=Refresh F11=Display activation group F12=Cancel
F16=Job menu F17=Top F18=Bottom F21=Include F22=Display entire field
Already at top of area.
MA a 01/001
I902 - Session successfully started


Session A - [24 x 80]
File Edit View Communication Actions Window Help
Display Call Stack
System: LP13UT16
Job: QP0ZSPWT User: SCOTTF Number: 412153
Thread: 0000001F
Type Program Statement Procedure
QP2SHELL2 QSYS _CXX_PEP__Fv
QP2SHELL2 QSYS 38 main
QP2USER QSYS 3 Qp2RunPase
QP2USER2 QSYS 12 __Qp2RunPase
QP2USER2 QSYS 6 runpase_main__FPi
QP2USER2 QSYS 5 runpase_common__FiPvT2
P istap 0000009C __start
P istap 274 00000298 main
P istap 318 0000112C run__Q4_8guardium3tap5istap >
P istap 552 000000B4 read_a_msg__Q4_8guardium3ta >
P istap 100 00000064 recv_message__27STap_SYSV_M >
P libc.a(shr.o) 000000C8 msgrcv
P unix 00000008 <syscall32>:__msgrcv
More...
F3=Exit F5=Refresh F11=Display activation group F12=Cancel
F16=Job menu F17=Top F18=Bottom F21=Include F22=Display entire field
MA a 01/001
I902 - Session successfully started
Session A - [24 x 80]
File Edit View Communication Actions Window Help
Display Call Stack
System: LP13UT16
Job: QP0ZSPWT User: SCOTTF Number: 412153
Thread: 0000001F
Type Program Statement Procedure
QP0ZSKRN QSYS 180 Qp0zKMsgrcv
QP0ZSKRN QSYS 2 waitToReceive__12Qp0zMsgEnt >
QP0ZSKRN QSYS 21 waitForMessage__12Qp0zMsgEn >
Bottom
F3=Exit F5=Refresh F11=Display activation group F12=Cancel
F16=Job menu F17=Top F18=Bottom F21=Include F22=Display entire field
MA a 01/001
I902 - Session successfully started
Session A - [24 x 80]
File Edit View Communication Actions Window Help
Display Call Stack
System: LP13UT16
Job: QP0ZSPWT User: SCOTTF Number: 412153
Thread: 00000020
Type Program Statement Procedure
QLESPI QSYS 19 LE_Create_Thread2__FP12crtt >
QP0WPINT QSYS 19 pthread_create_part2
QP2USER2 QSYS 3 __Qp2Thread
QP2USER2 QSYS 5 runpase_common__FiPvT2
P unix *N
P libpthread.a(shr_xpg5 > 00000104 _pthread_body
P istap 259 000018BC run__Q4_8guardium3tap5istap >
P istap 733 00000100 update_worker__Q4_8guardium >
P libc.a(shr.o) 000000D8 usleep
P libc.a(shr.o) 000000E8 nsleep
P libpthread.a(shr_xpg5 > 00000010 _p_nsleep
P unix 00000008 <syscall32>:_nsleep
Bottom
F3=Exit F5=Refresh F11=Display activation group F12=Cancel
F16=Job menu F17=Top F18=Bottom F21=Include F22=Display entire field
MA a 01/001
I902 - Session successfully started



Audit Server tracing

If you are confronted with a case where the Audit server is started, the audit server status indicates that auditable events are being processed, but the endpoint Guardium client does not display any detail, you could examine several things.

These SQL statements are most easily executed from IBM i Navigator's Run SQL Scripts.

This trace provides some detail not explained in this document. I included the topic in case Level 3 requests an Audit Server trace.

NOTE: If the trace table is empty, that indicates successful delivery of audit messages to the S-TAP PASE program.

```
-- The trace is turned on by creating a trace table
-- The audit server has to be restarted to activate it
CREATE TABLE QRECOVERY.QSQGDTRC1 (
Trace_Entry_Timestamp TIMESTAMP DEFAULT CURRENT TIMESTAMP,
Journal_Sequence_Number BIGINT,
Enqueued CHAR(1),
Filter_Reason INT,
Message BLOB(4M));

--
-- Note: if you are only interested in capturing data for unexpected failures, execute
-- the following COMMENT ON statement.
--
-- However, if you want full trace detail captured, do not execute the COMMENT ON
statement
--
COMMENT ON TABLE qrecovery.qsqgdtrc1 IS 'ERROR ONLY'

commit;

-- Restart the audit server and run the statements you believe should be captured
```

```
select
Trace_Entry_Timestamp,
Journal_Sequence_Number,
Filter_Reason,
case Filter_reason
-- likely reasons
  when 1 then 'User filter'
  when 2 then 'RDB filter'
  when 3 then 'SV Entry Type Not "A"'
  when 4 then 'GR Entry Type Not "FZR *USAGEFAILURE"'
  when 5 then 'GR Entry Type Not "Connect Failure"'
  when 6 then 'Not a Database Type"'

-- uninteresting entries
  when 11 then 'SYSAUDSTS file'

-- unlikely reasons
  when 91 then 'Bad Entry Specific Data'
  when 92 then 'msgsnd Error'
  else 'Unknown' END

,A.*

FROM QRECOVERY.QSQGDTRC1 A
-- Uncomment this WHERE clause if needed
-- WHERE ENQUEUED = 'N'
ORDER BY Trace_Entry_Timestamp;
```

-- Don’t forget to drop the table and restart the audit server after you have finished
-- your analysis.

```
DROP TABLE QRECOVERY.QSQGDTRC1;
```

Guardium V9.0 and FTP monitoring

Since many customers have questions about ftp coverage from Guardium V9.0 Data Activity Monitor and Audit, this section was added to this document. As you'll see, the answer is not as simple as yes or no.

When a customer asks about ftp tracking capabilities, we should reply by asking the customer to clarify what they hope to accomplish. (i.e. clarify their requirement)

Table 1. Guardium, Db2 for i and ftp use cases

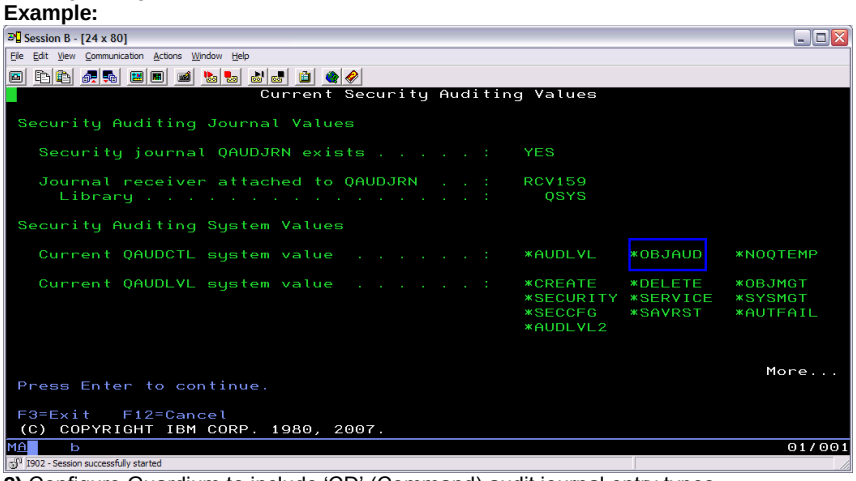
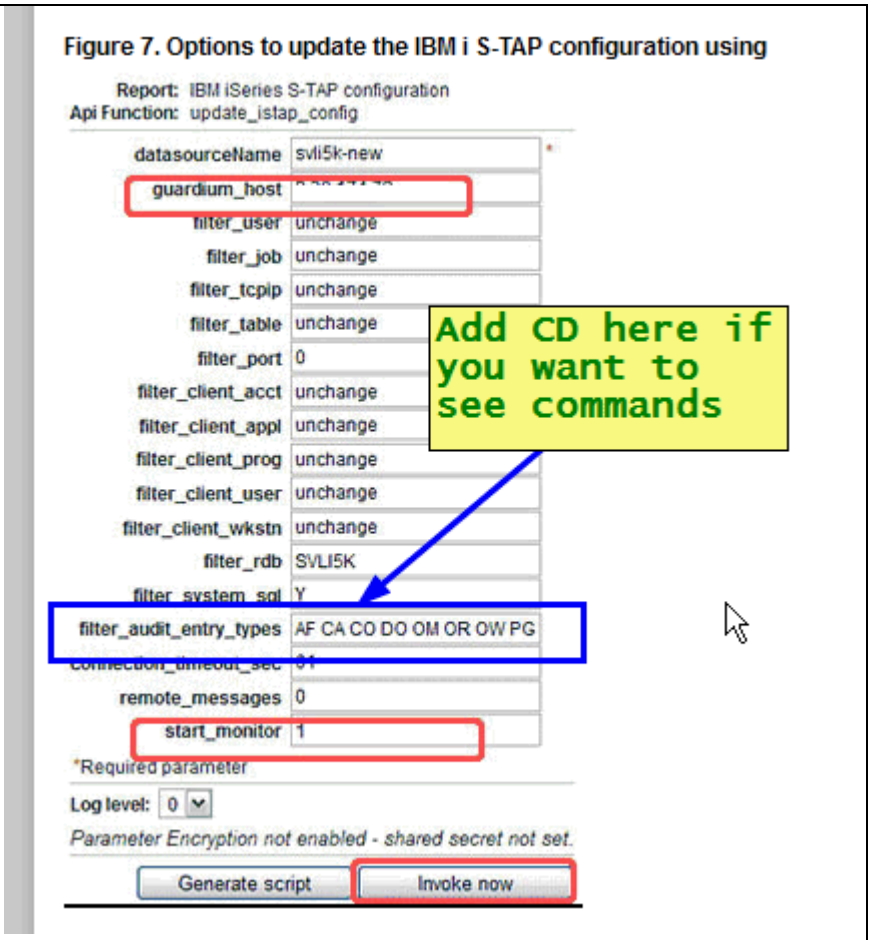
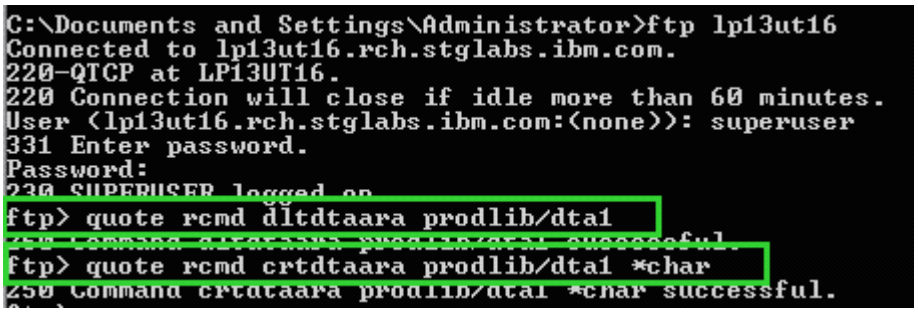
Use case	Guardium V9.0 Data Activity Monitor and Audit – Db2 for i as a data source
Comprehensive ftp activity monitoring	Not fully supported, consider a different solution.
Tracking ftp remote command execution by a trusted user	Supported. Since it is quite common to execute commands, we chose to have the 'CD' QAUDJRN audit entries NOT included by default under the filter_audit_entry_types configuration. To include command execution auditing you can adjust the configuration to include CD audit entry types. Steps: **1) CHGUSRAUD USRPRF(SUPERUSER) AUDLVL(*CMD)** **2) DSPSECAUD** needs to be set to create audit journal entries based upon individual object auditing settings **Example:**  **3)** Configure Guardium to include 'CD' (Command) audit journal entry types

Figure 7. Options to update the IBM i S-TAP configuration using



ftp example:



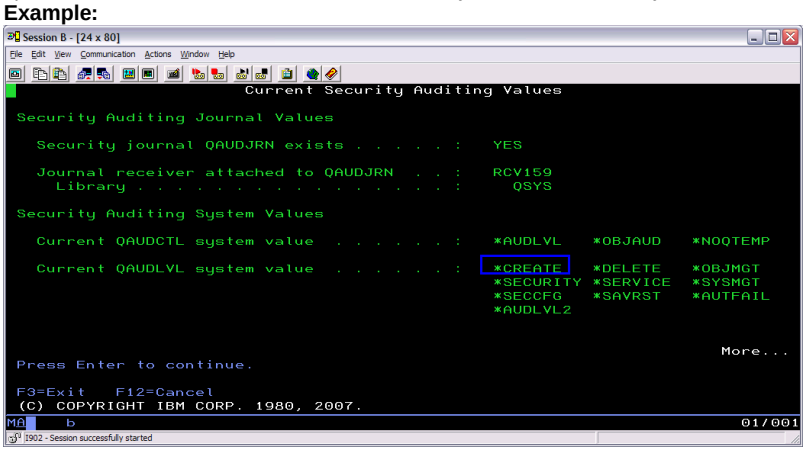
DB2 I FULL SQL WITH SQL PARAM

Start Date: **2013-03-29 11:01:33** End Date: **2013-03-29 15:01:33**
Aliases: OFF

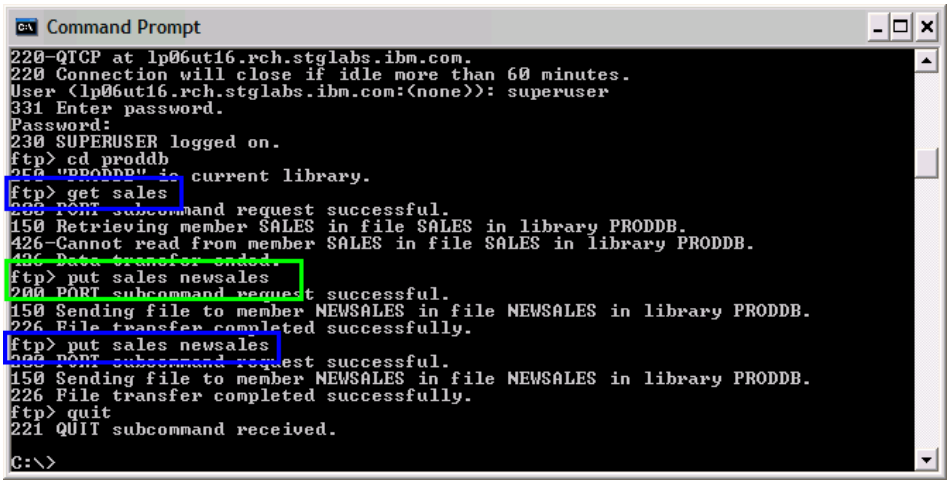
Timestamp	DB User Name	Application User	Full Sql		Response Time	Records Affected	Full SQL ID	Bind Variables Values
2013-03-29 12:38:43.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP13UT16	CA - Authority change DTA1 *DTAARA	PRODLIB	0	-1	19262555	
2013-03-29 12:38:43.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP13UT16	CD - Command string CRTDTAARA *CMD	QSYS	0	-1	19262554	CRTDTAARA DTAARA(PRODLIB/DTA1) TYPE(*CHAR)
2013-03-29 12:38:43.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP13UT16	CO - Create object DTA1 *DTAARA	PRODLIB	0	-1	19262556	
2013-03-29 12:37:35.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP13UT16	CD - Command string DLTDTAARA *CMD	QSYS	0	-1	19262552	DLTDTAARA DTAARA(PRODLIB/DTA1)
2013-03-29 12:37:35.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP13UT16	DO - Delete object DTA1 *DTAARA	PRODLIB	0	-1	19262553	
2013-03-29 12:36:31.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP13UT16	CD - Command string CHGCURLIB *CMD	QSYS	0	-1	19262539	QSYS/CHGCURLIB CURLIB(PRODLIB)

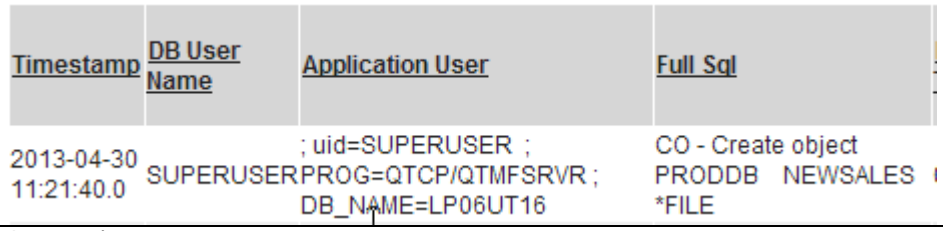
<table>
<tr>
<td>Tracking the creation of new files using ftp's put/mput by a trusted user</td>
<td>Supported.

If put is used to overlay an existing file, the action is not tracked because an object was not created.

Steps
1) Update the filter to include the trusted user name or trusted group profile name
Example: update qsys2/sysaudit set filter_user = 'SUPERUSER'
2) DSPSECAUD needs to be set to create audit journal entries for object creation.
Example:
 

ftp example:
The action in the green box (the middle box) is monitored.
The actions in the blue boxes are not monitored.

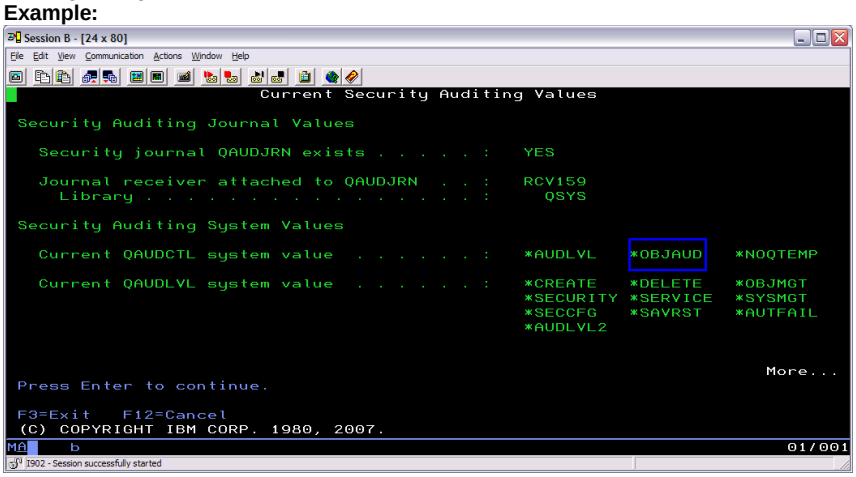
 

 
</td>
</tr>
<tr>
<td>Tracking the access of sensitive files using ftp's get/mget by a trusted user

Or</td>
<td>Supported.

If get is used to access a file, object auditing would need to be configured to track the object access.

You might be wondering why not just set all objects to have OBJAUD(*ALL)? The customer needs to carefully consider which objects are sensitive and require this level of tracking because of the amount of journaling that will occur.</td>
</tr>
</table>

Tracking the creation of new files or overlay of existing files	**Steps** 1) Update the filter to include the trusted user name or trusted group profile name **Example: update qsys2/sysaudit set filter_user = 'SUPERUSER'** 2) DSPSECAUD needs to be set to create audit journal entries based upon individual object auditing settings **Example:**  3) Set the object auditing values. **Example:** **CHGOBJAUD OBJ(PRODDB/SALES) OBJTYPE(*FILE) OBJAUD(*ALL)** **ftp example:** The actions in the green boxes are monitored.

Timestamp	DB User Name	Application User	Full Sql
2013-04-30 11:38:51.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP06UT16	ZC - Change object PRODDB SALES *FILE
2013-04-30 11:38:51.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP06UT16	ZC - Change object PRODDB SALES *FILE
2013-04-30 11:38:51.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP06UT16	ZC - Change object PRODDB SALES *FILE
2013-04-30 11:38:51.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP06UT16	ZC - Change object PRODDB SALES *FILE
2013-04-30 11:38:49.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP06UT16	ZC - Change object PRODDB SALES *FILE
2013-04-30 11:38:49.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP06UT16	ZC - Change object PRODDB SALES *FILE
2013-04-30 11:38:49.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP06UT16	ZC - Change object PRODDB SALES *FILE
2013-04-30 11:38:49.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP06UT16	ZC - Change object PRODDB SALES *FILE
2013-04-30 11:38:28.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP06UT16	ZR - Read object PRODDB SALES *FILE
2013-04-30 11:38:28.0	SUPERUSER	; uid=SUPERUSER ; PROG=QTCP/QTMFSRVR ; DB_NAME=LP06UT16	ZR - Read object PRODDB SALES *FILE
2013-04-30 11:37:34.0	SCOTTF	; uid=SCOTTF ; PROG=QSYS/QCMD ; DB_NAME=LP06UT16	AD - Auditing change PRODDB SALES *FILE

Capturing failed login attempts via ftp

Failed login attempts via ftp are like any other failed login attempt on IBM i.

A PW (Password) journal entry is generated within the audit journal and it contains the details of the access violation.

For details on the PW (Password) journal entries, look here:

http://pic.dhe.ibm.com/infocenter/iseries/v7r1m0/index.jsp?topic=%2Fapis%2Fqsyrusri.htm

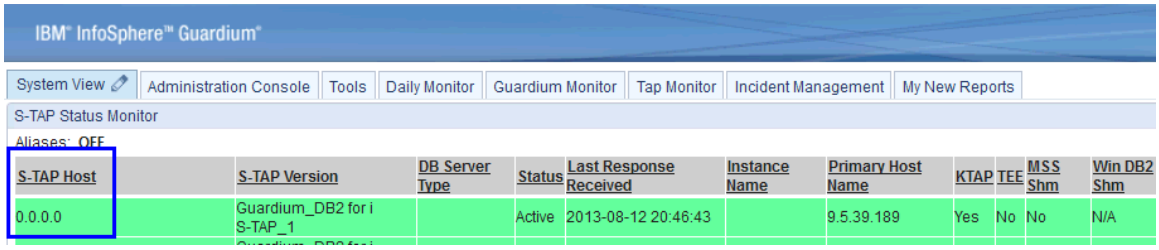
Keep in mind though, that the IBM i audit configuration has to be configured to indicate that authorization failures are being audited.

The QAUDCTL system value indicates whether QAUDLVL or QAUDLVL2 are being used.

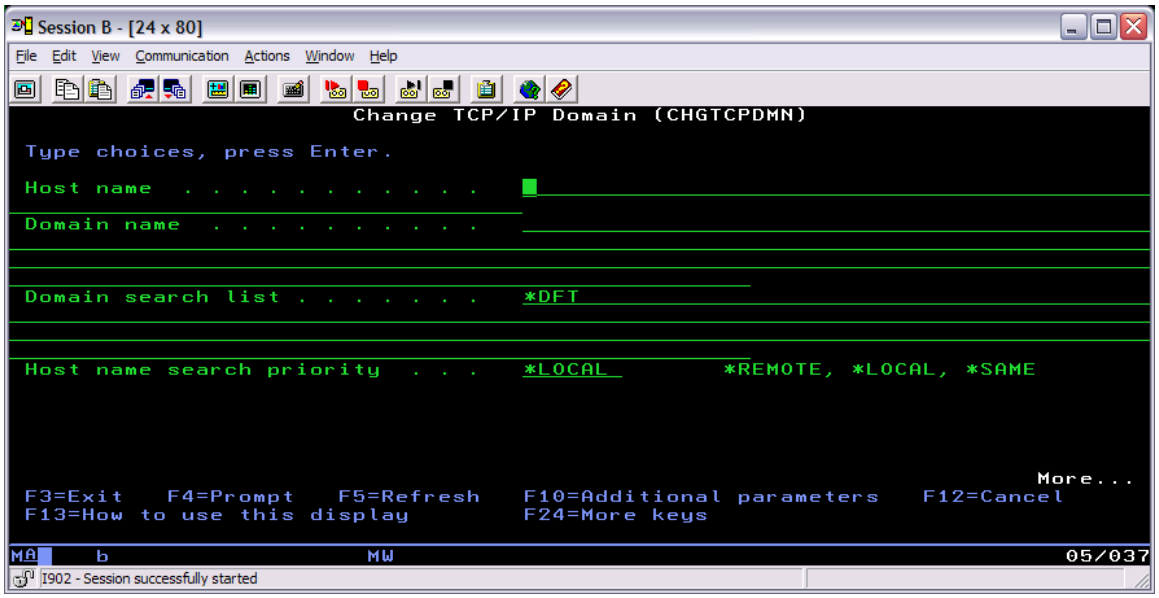
The QAUDLVL / QAUDLVL2 system value being used will need to include *AUTFAIL. Once *AUTFAIL is indicated, the PW entries should appear in QSYS/QAUDJRN. The Guardium S-TAP configuration includes PW by default, so the entries should flow to the Guardium appliance as long as PW wasn't removed.

Specifying a TCP/IP Domain name on the IBM i

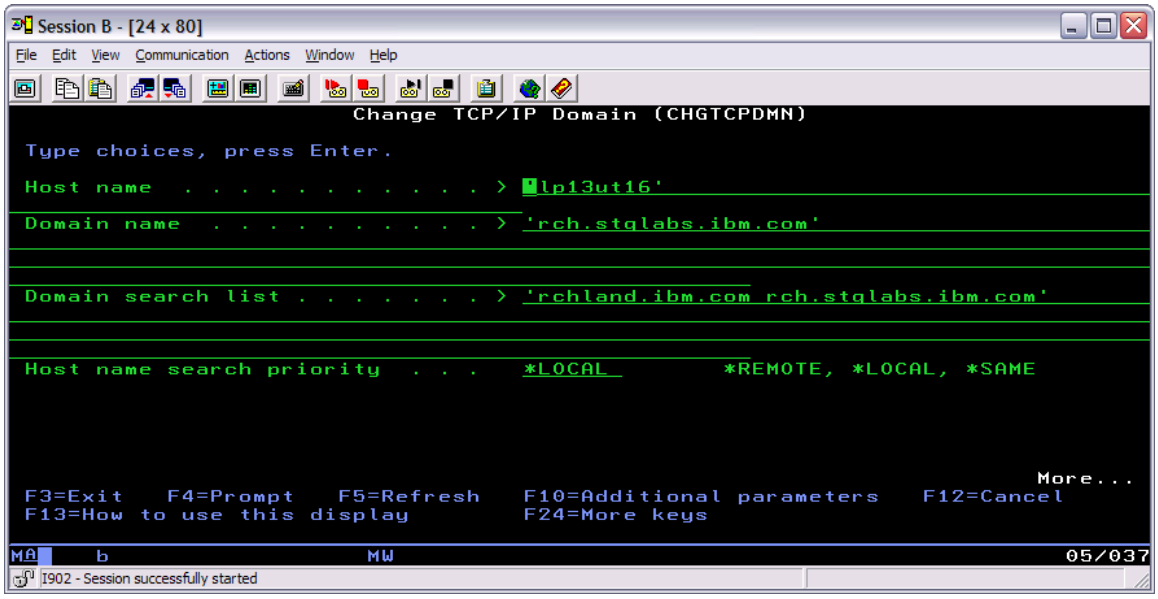
If you see 0.0.0.0 appear for the Db2 for i S-TAP Host, the target IBM i most likely does not have a defined name. This is not a hard requirement, but things will run better when TCP/IP can utilize a name for the host and not simply a dotted IP address.



When you see 0.0.0.0, enter the Change TCP/IP Domain (CHGTCPDMN) command and press PF4. If you see nothing for the Host name, you can improve the configuration by entering a name for the Host Name that matches the database name.

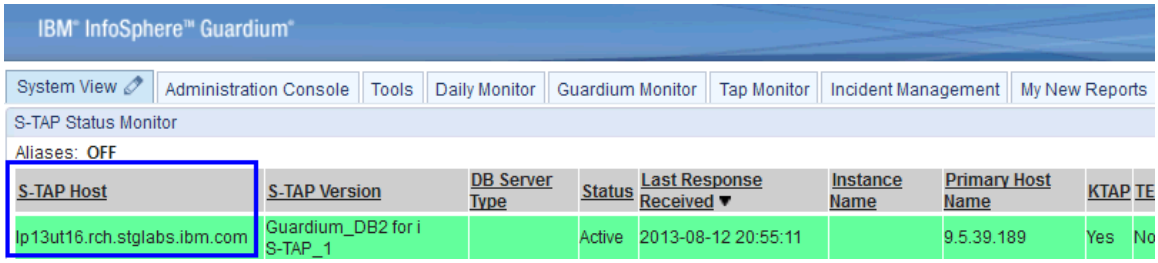


The name is not case sensitive.



After making any changes to the Domain, restart the Audit Server.

> RUNSQL SQL('call sysproc.sysaudit_start_batch("") ')
COMMIT(*NONE) NAMING(*SQL)



Removing Guardium S-TAP

To disable, end, and uninstall Guardium S-TAP for IBM i, issue the following commands:

RUNSQL SQL('call SYSPROC/SYSAUDIT_End') COMMIT(*NONE)
and
RMVDIR DIR('/usr/local/guardium') SUBTREE(*ALL)

Determining the PASE S-TAP version

To determine the version level of the IBM i S-TAP, execute the following:

CALL QP2TERM
cd /usr/local/guardium
$
strings -a istap | grep itap_version

```
                                  /QOpenSys/usr/bin/-sh
   $
 > cd /usr/local/guardium
   $
 > strings -a istap | grep itap_version
   ÿÿÿÿÿÿÿÿitap_version : 2_10.0.0_r80758_trunk_1
   $
```

Well-defined Port numbers for IBM i

When the Server Port indicates a non-zero port number, this is the resource page that can be referenced within the IBM i 7.1 Information Center to understand the interface being used.

http://bit.ly/ibmiPorts

IBM i Navigator ➔ 8471
Host Server ➔ 8471
DDM ➔ 446
DRDA ➔ 446

Configuring the Audit Server Subsystem

If you don't want Guardium to run in the QBATCH subsystem, here are the steps you can take to configure and use a user-specified subsystem for the Audit Server.

IBM i Commands:

- CRTSBSD SBSD(QGPL/GUARDSBS)
  POOLS((1 *BASE)) TEXT('Guardium SBS')

- CRTJOBQ QGPL/GDJOBQ TEXT('Guardium job queue')

- CRTUSRPRF GDUSER PASSWORD(*NONE) PWDEXP(*NO)
  STATUS(*ENABLED)
  SPCAUT(*ALLOBJ *JOBCTL) TEXT('Guardium user profile')

- CRTJOBD QGPL/GDAUDIT JOBQ(GDJOBQ) JOBPTY(2) USER(GDUSER)
  JOBMSGQFL(*WRAP) LOG(4 0 *SECLVL) TEXT('Guardium job description')

- CHGUSRPRF GDUSER JOBD(QGPL/GDAUDIT)

- ADDJOBQE SBSD(QGPL/GUARDSBS)
  JOBQ(QGPL/GDJOBQ) MAXACT(10) SEQNBR(40)

- CRTCLS CLS(QGPL/GDCLS) RUNPTY(1) TIMESLICE(10000)
  TEXT('Guardium class')

- ADDRTGE SBSD(QGPL/GUARDSBS) SEQNBR(800)
  CMPVAL(GUARDIUM) PGM(QSYS/QCMD) CLS(QGPL/GDCLS)

- STRSBS SBSD(QGPL/GUARDSBS)

To have this subsystem automatically started across IPLs, add the STRSBS command to the QSTRUP program

- RTVCLSRC PGM(QSYS/QSTRUP) SRCFILE(QGPL/QCLSRC)
- STRSEU SRCFILE(QGPL/QCLSRC) SRCMBR(QSTRUP)
  TYPE(CLP) OPTION(2)
- Immediately after the DONE: label add these two lines
  STRSBS SBSD(QGPL/GUARDSBS)
  MONMSG MSGID(CPF0000)
- CRTCLPGM PGM(QSYS/QSTRUP) SRCFILE(QGPL/QCLSRC)

From the Guardium Appliance, use **update_istap_config** to change **start_user** to GDUSER. Lastly, from the Guardium Appliance, use **start_istap_monitor** to Start/Restart the Audit Server.

Protecting the Audit Server Configuration File

Note: The following information is provided to be illustrative of some of the considerations and configuration choices. Every client is encouraged to employ a security expert or to contract with a security expert consultant prior to deploy or changing the security configuration.

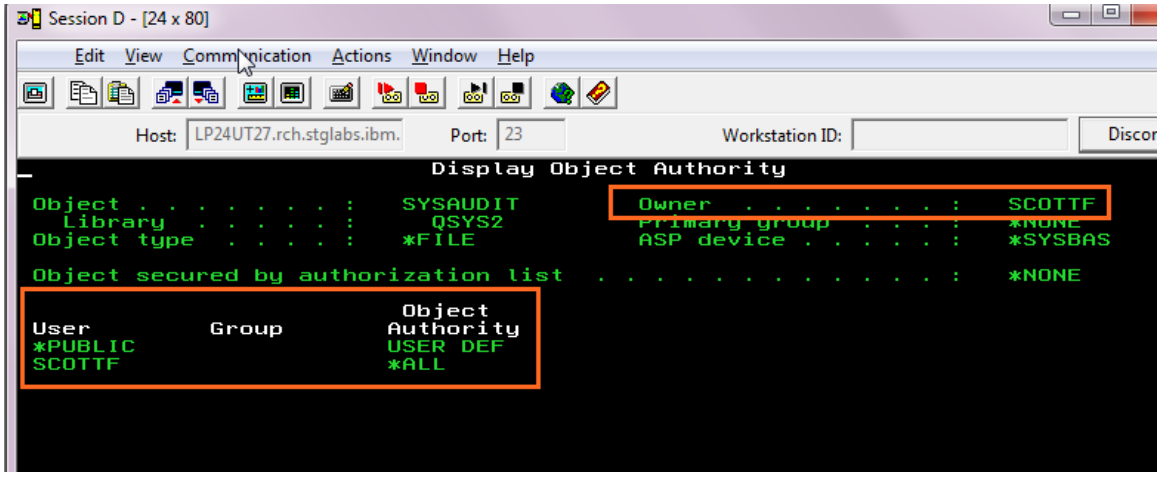
The QSYS2/SYSAUDIT *FILE indicates which database activity (users, jobs, tables, SQL queries, etc...) are monitored by the audit server. Therefore, the SYSAUDIT table is a critical security resource and needs to be both protected and audited.

To protect the QSYS2/SYSAUDIT you need to regularly review the following security settings:

- **Ownership** – By default, the SYSAUDIT table will be owned by whichever user installed the Audit Server PASE program. The owner is permitted to query, change, alter or remove the SYSAUDIT *FILE.
- **Private Authority** – By default, the owner of the file is granted *ALL authority to SYSAUDIT.
- **Public Authority** – By default, any user (i.e. the public) can query SYSAUDIT and discover the filtering strategy.

To review the security authorization of the QSYS2/SYSAUDIT *FILE:
DSPOBJAUT OBJ(QSYS2/SYSAUDIT) OBJTYPE(*FILE) AUTTYPE(*OBJECT)

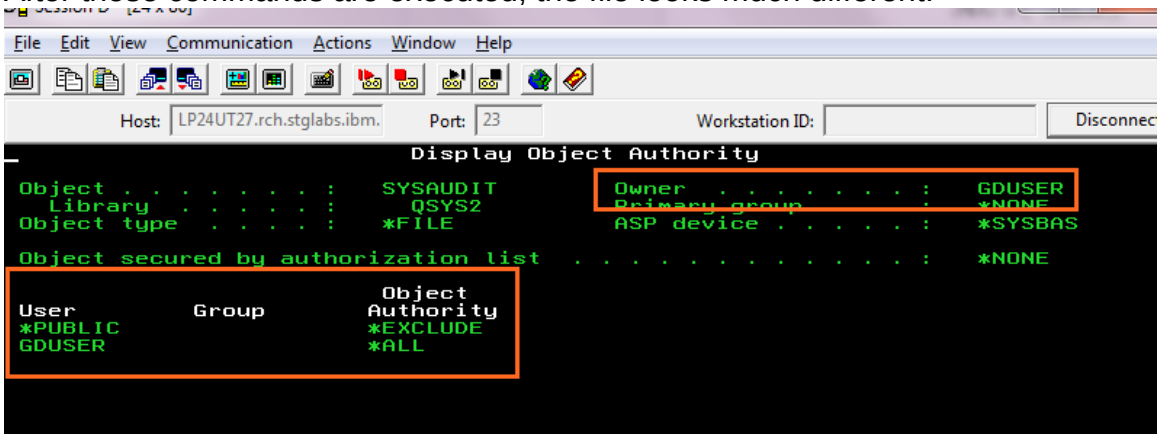
As we see in the image, the user SCOTTF is both the owner and also has private authorities. Also, by default, any user is allowed to query the contents of QSYS2/SYSAUDIT.



To completely lock down and limit access to the SYSAUDIT, we can use the following commands:

- CHGOBJOWN OBJ(QSYS2/SYSAUDIT) OBJTYPE(*FILE) NEWOWN(GDUSER)
- GRTOBJAUT OBJ(QSYS2/SYSAUDIT) OBJTYPE(*FILE) USER(GDUSER) AUT(*ALL)
- RVKOBJAUT OBJ(QSYS2/SYSAUDIT) OBJTYPE(*FILE) USER(*PUBLIC) AUT(*ALL)

After these commands are executed, the file looks much different:



Even though the SYSAUDIT table now is protected from many potential security related exposures, we still need to take additional steps to audit changes to the file. Why? Any user with *ALLOBJ user special authority can change the contents of this file. By enabling SYSAUDIT to generate audit records, we will be able to see any changes appear on the Guardium activity report.

To configure object auditing:

- CHGOBJAUD OBJ(QSYS2/SYSAUDIT) OBJTYPE(*FILE) OBJAUD(*ALL)

Of course, auditing relies upon several other things being set up correctly. For this reason, it is again recommended that you engage a security expert, construct a Guardium activity report for QSYS2/SYSAUDIT, and conduct tests to confirm that this file is properly protected, audited and monitored.

Automating the restart of the Audit Server when leaving restricted state

Background:
Restricted state requires that all subsystems end. When a client enters restricted state (ENDSBS *ALL), the Guardium i-STAP audit server is ended.
When leaving restricted state via an IPL, the audit server will be automatically restarted.

When leaving restricted state via STRSBS, the client needs to either manually restart the audit server or automate the restart of the Audit Server using the following steps:

GO SAVE is a command which displays many system save options. Option 21 is a popular option because it saves the entire system and automatically leaves restricted state when completed. To start workloads in addition to subsystems (i.e. the Audit Server jobs), the QMNSRBND program can be customized as shown in the steps below.

Post-save Auto-restart Customization steps:

1. **CRTSRCPF FILE(QGPL/QCLSRCDBCS) CCSID(937)**
2. **RTVCLSRC PGM(QSYS/QMNSRBND) SRCFILE(QGPL/QCLSRCDBCS)**
3. **STRSEU SRCFILE(QGPL/QCLSRCdbcs) SRCMBR(QMNSRBND) TYPE(CLP) OPTION(2)**
4. Search for END2:
5. Add the next two lines, prior to the END2:, and save:
   RUNSQL SQL('call sysproc/sysaudit_start_batch("")') COMMIT(*NONE) MONMSG MSGID(CPF0000)
6. **CRTCLPGM PGM(QSYS/QMNSRBND) SRCFILE(QGPL/QCLSRCDBCS)**

Automating the restart of the Audit Server when the Audit Server subsystem is manually ended and restarted

Background:
To have coverage over the scenario where GUARDSBS is ended and restarted outside of an IPL or leaving restricted state, use the steps below to automate the restart of the Audit Server.

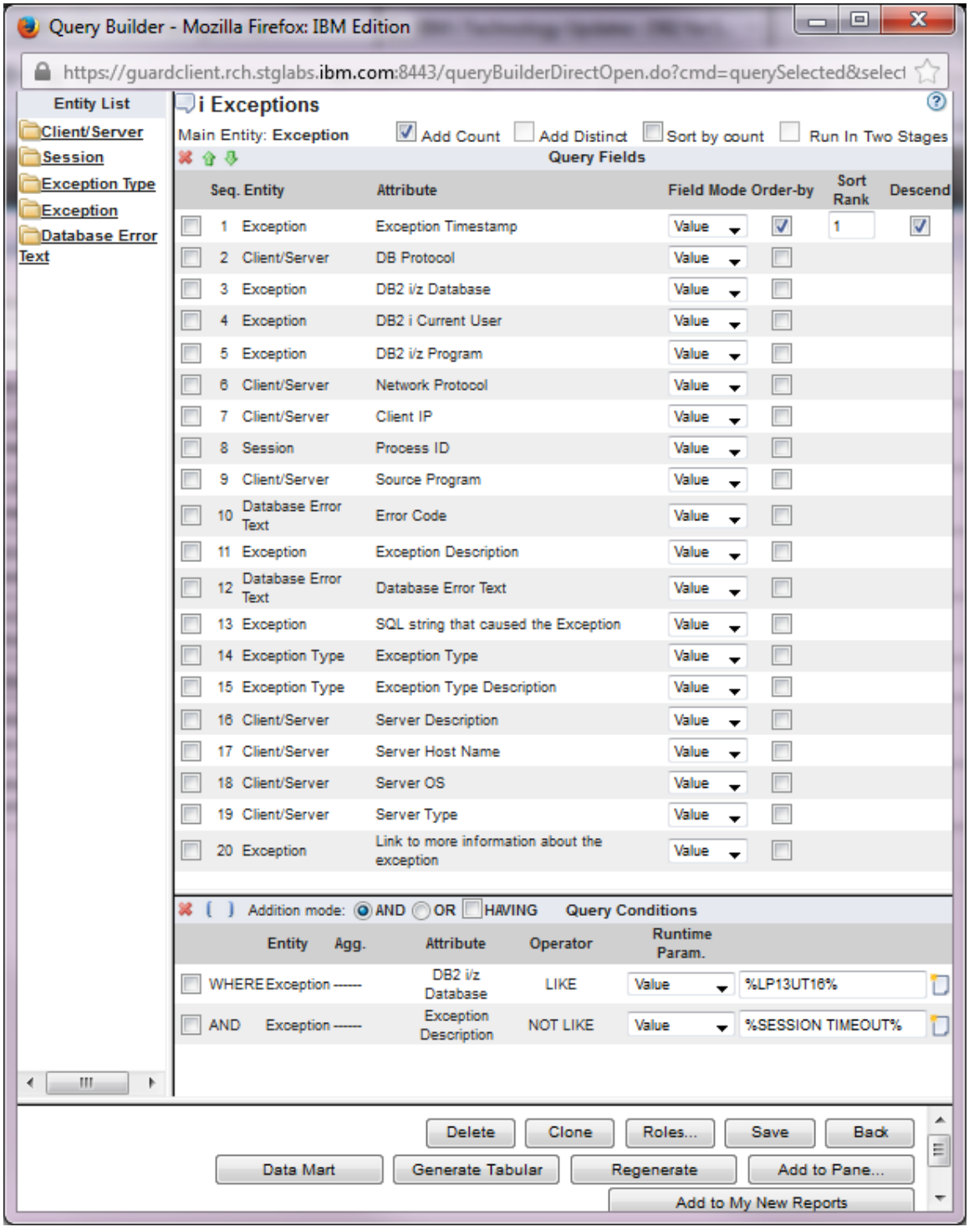
Post-save Auto-restart Customization steps:

1. **CHGJOBD JOBD(QGPL/GDAUDIT) USER(GDUSER) RQSDTA('RUNSQL SQL(''CALL SYSPROC/SYSAUDIT_START_BATCH(''''''')'') NAMING(*SYS) COMMIT(*NONE)')**
2. **ADDAJE SBSD(QGPL/GUARDSBS) JOB(GUARDIUM) JOBD(QGPL/GDAUDIT)**
3. **CHGRTGE SBSD(QGPL/GUARDSBS) SEQNBR(800) CMPVAL(*ANY)**

Exception Report – Recommended Report Definition

This report will show you the failures in descending time order. Beware of the "SESSION TIMEOUT" rows as they only indicate that an active session has not produced any new audit data (~ 60 minutes) and the Exception Timestamp value comes from the collector. If your collector's time does not match the IBM i's time, the resulting report can look confusing.

Exception Report definition:



Exception Report example output:

Exception Timestamp	DB Protocol	DB2 i/z Database	DB2 i Current User	DB2 i/z Program	Network Protocol	Client IP	Process ID	Source Program	Error Code	Exception Description	Database Error Text	SQL string that caused the Exception
2014-03-13 09:47:54.0	DB2 I	LP13UT16	NEWBIE1235	QTCP/QTMFSRVR	QAUDJRN	9.10.110.45	154568	QTCP/QTFTP00018	N/A	PW - Invalid password or user ID NEWBIE1235 REASON: P - PASSWORD NOT VALID	N/A	
2014-03-13 09:42:52.0	DB2 I	LP13UT16	NEWBIE1235	QTCP/QTMFSRVR	QAUDJRN	9.10.110.45	154625	QTCP/QTFTP00115	-551	42501:-551	The authorization ID does not have the privilege to perform the specified operation on the identified object.	AF - Authority failure SYSIBM SYSDUMMY1 *FILE
2014-03-13 09:42:47.0	DB2 I	LP13UT16	NEWBIE1235	QTCP/QTMFSRVR	QAUDJRN	9.10.110.45	154625	QTCP/QTFTP00115	-551	42501:-551	The authorization ID does not have the privilege to perform the specified operation on the identified object.	AF - Authority failure STORE123 EMPLOYEE *FILE

Exception Report - Mapping data to Entity Fields

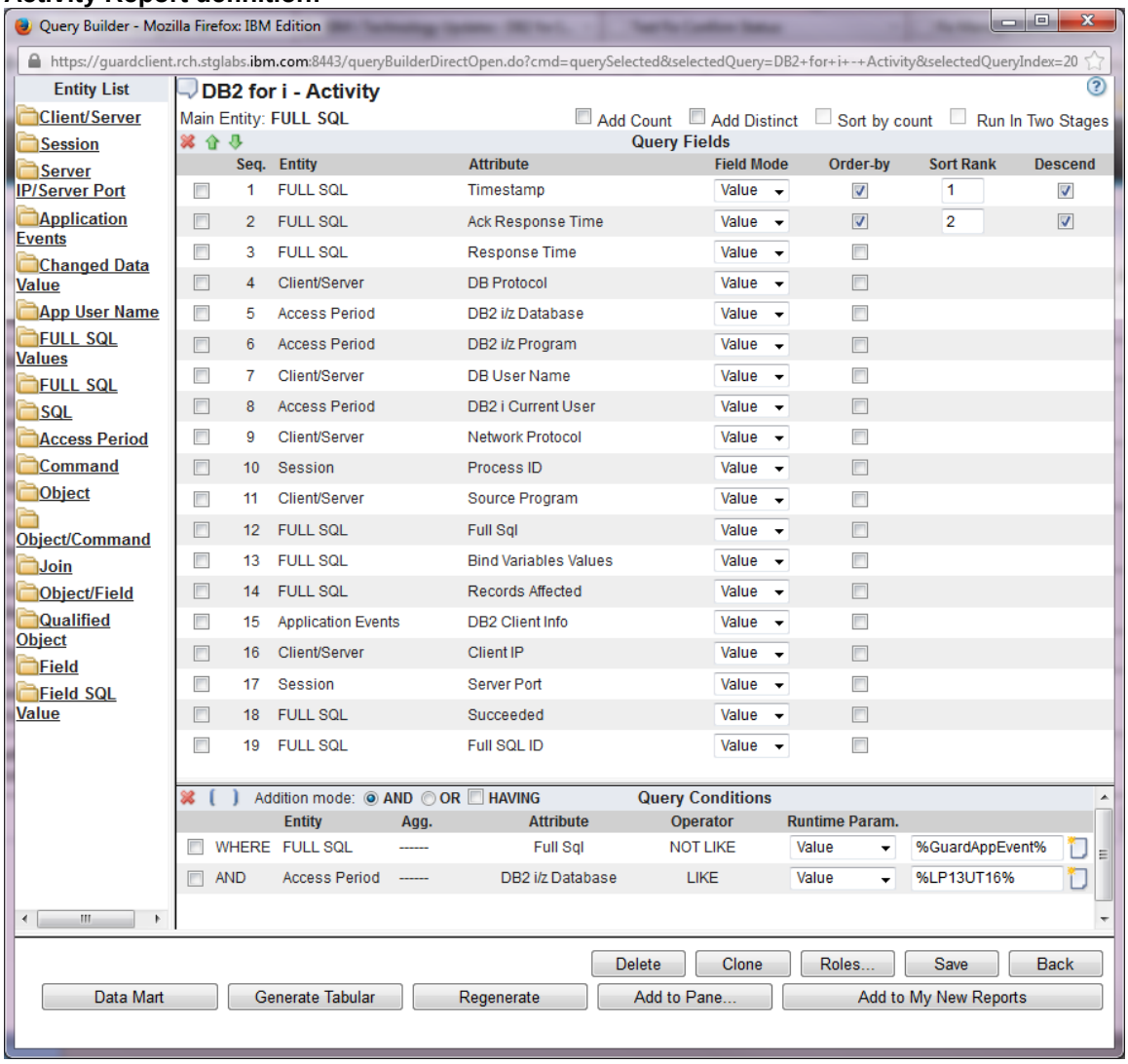
Failure Detail	Included in the SQL Database Monitor data? (DBMON Column name)	Included in the Audit Journal entry data?	Availability within Guardium (Entity Name → Field)
Database Type	Yes Always set to "Db2 I"	Yes Always set to "Db2 I"	Client/Server → DB Protocol
Job number	Yes (QQJNUM)	Yes	Session → Process ID
Job user/Job name	Yes (QQUSER/QQJOB)	Yes	Client/Server → Source Program
Start time	Yes (QQSTIM)	Yes	Exception → Exception Timestamp
Start time (microsecond portion)	Yes (QQSTIM)	No	Not Available
SQLSTATE:SQLCODE	Yes (QQC81/QQI8)	08001 for invalid password (PW) and for general purpose audit records (GR) 42501 for authority failure (AF) 00000 everything else When the failure cannot be mapped to an SQLSTATE & SQLCODE, text describing the failure will appear. Example: PW - Invalid password or user ID	Exception → Exception Description
SQLCODE	Yes (QQI8)	-30080 for invalid password (PW) and for general purpose audit records (GR) -551 for authority failure. (AF) 0 everything else When the failure cannot be mapped to an SQL equivalent, N/A will appear.	Database Error Text → Error Code
Exception Type	Yes – always set to "SQL_ERROR"	Yes – set to "SQL_ERROR" or "LOGIN_FAILED"	Exception Type → Exception Type
Database Error Text	Yes – verbose explanation of failure type	Only available when the audit entry can be mapped to a similar SQL failure	Database Error Text → Database Error Text
Exception Description	Yes – brief description of failure type	Yes – brief description of failure type	Exception Type → Exception Description
SQL statement	Yes – limited to 60K (QQ1000L)	Yes – Journal Entry data	Exception → SQL string that caused the Exception
SQL variables	Yes - limited to 1000 bytes (QQ1000 from QQRID=3010)	No	Not Available
Interface	Yes – Subsystem Name (QVC5001)	Yes - Always QAUDJRN	Client/Server → Network Protocol
Client application name	Yes (QVC3001)	No	Exception → Link to more information about the exception
Client user ID	Yes (QVC3002)	No	Exception → Link to more information about the exception
Client workstation	Yes (QVC3003)	No	Exception → Link to more information about the exception
Client accounting	Yes (QVC3005)	No	Exception → Link to more information about the exception
Client program	Yes (QVC3006)	No	Exception → Link to more information about the exception
Current user	Yes (QVC102)	Yes	Exception → Db2 i Current User
Thread ID	Yes (QQI9)	Yes	Not Available
Program schema/ Program name	Yes, if the statement is executed from a program or service	Yes, if the statement is executed from a program or service program	Exception → Db2 i/z Program

	program (QQC104/QQC103)		
Client IP Address	Yes (QQC183)	Yes	Client/Server → Client IP
Client Port Number	Yes (QQSMINT2)	Yes	Session → Client Port
RDB name	Yes (QQRDBN)	Yes	Exception → Db2 i/z Database
Number of rows	Yes, only for INSERT, DELETE, UPDATE, MERGE, OPEN*, VALUES INTO, CREATE TABLE AS, DECLARE GLOBAL TEMPORARY TABLE AS, and SET VARIABLE (QQI2)	No	Not Available

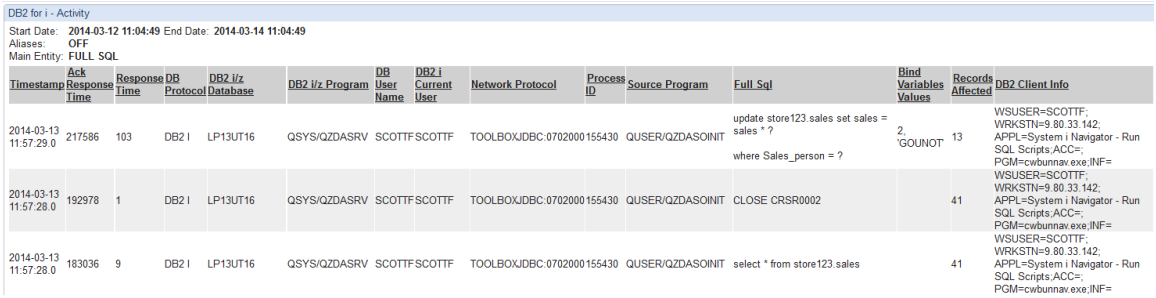
Activity Report – Recommended Report Definition

This report will show you all activity, both successes and failures in descending time order.

Activity Report definition:



Activity Report example output:



DB2 for i - Activity

Start Date: **2014-03-12 11:04:49** End Date: **2014-03-14 11:04:49**
Aliases: OFF
Main Entity: FULL SQL

Timestamp	Ack Response Time	Response Time	DB Protocol	DB2 i/z Database	DB2 i/z Program	DB User Name	DB2 i Current User	Network Protocol	Process ID	Source Program	Full Sql	Bind Variables Values	Records Affected	DB2 Client Info
2014-03-13 11:57:29.0	217586	103	DB2 I	LP13UT16	QSYS/QZDASRV	SCOTTF	SCOTTF	TOOLBOXJDBC:07020001	55430	QUSER/QZDASOINIT	update store123.sales set sales = sales * ? where Sales_person = ?	2, 'GOUNOT'	13	WSUSER=SCOTTF; WRKSTN=9.80.33.142; APPL=System i Navigator - Run SQL Scripts;ACC=; PGM=cwbunnav.exe;INF=
2014-03-13 11:57:28.0	192978	1	DB2 I	LP13UT16	QSYS/QZDASRV	SCOTTF	SCOTTF	TOOLBOXJDBC:07020001	55430	QUSER/QZDASOINIT	CLOSE CRSR0002		41	WSUSER=SCOTTF; WRKSTN=9.80.33.142; APPL=System i Navigator - Run SQL Scripts;ACC=; PGM=cwbunnav.exe;INF=
2014-03-13 11:57:28.0	183036	9	DB2 I	LP13UT16	QSYS/QZDASRV	SCOTTF	SCOTTF	TOOLBOXJDBC:07020001	55430	QUSER/QZDASOINIT	select * from store123.sales		41	WSUSER=SCOTTF; WRKSTN=9.80.33.142; APPL=System i Navigator - Run SQL Scripts;ACC=; PGM=cwbunnav.exe;INF=

Activity Report - Mapping data to Entity Fields

To order your report by most recent to least recent, use these sorting keys:
Sort Rank 1) FULL SQL → Timestamp (Descend)
Sort Rank 2) FULL SQL → Ack Response Time (Descend)

Failure Detail	Included in the SQL Database Monitor data?	Included in the Audit Journal entry data?	Availability within Guardium (Entity Name → Field)
Start time	Yes (QQSTIM)	Yes	FULL SQL → Timestamp
Start time (microsecond portion)	Yes (QQSTIM)	No	FULL SQL → Ack Response Time
Response Time (milliseconds)	Yes (QQETIM – QQSTIM)	No	FULL SQL → Response Time
Database Type	Always set to “Db2 I”	Always set to “Db2 I”	Client/Server → DB Protocol
RDB name	Yes (QQRDBN)	Yes	Access Period → Db2 i/z Database
Program schema/ Program name	Yes, if the statement is executed from a program or service program (QQC104/QQC103)	Yes, if the statement is executed from a program or service program	Access Period → Db2 i/z Program
Current user	Yes (QVC102)	Yes	Access Period → Db2 i Current User
Interface	Yes – Subsystem Name or Interface detail (QVC5001)	Yes - Always QAUDJRN	Client/Server → Network Protocol
Succeeded	Yes 0 = Failure 1 = Success	Yes 0 = Failure 1 = Success	FULL SQL → Succeeded
Job number	Yes (QQJNUM)	Yes	Session → Process ID
Job user/Job name	Yes (QQUSER/QQJOB)	Yes	Client/Server → Source Program
SQL statement text	Yes – limited to 60K (QQ1000L)	Yes - journal entry data	FULL SQL → Full SQL
SQL variables	Yes - limited to 1000 bytes (QQ1000 from QQRID=3010)	No	FULL SQL → Bind Variables Values
Number of rows	Yes, only for INSERT, DELETE, UPDATE, MERGE, OPEN*, VALUES INTO, CREATE TABLE AS, DECLARE GLOBAL TEMPORARY TABLE AS, and SET VARIABLE (QQI2)	Yes	FULL SQL → Records Affected
Client application name **Client user ID** **Client workstation** **Client accounting** **Client program**	Yes (QVC3001, QVC3002, QVC3003, QVC3005, QVC3006)	No	Application Events → Db2 Client Info
Thread ID	Yes (QQI9)	Yes	Not Available
Client IP Address	Yes (QQC183)	Yes	Client/Server → Client IP
Client Port Number	Yes (QQSMINT2)	Yes	Session → Server Port
Server Type (always ‘Db2’)	NA	NA	Client/Server → Server Type
Server OS	NA	NA	Client/Server →

(always 'IBM I') **DB Protocol** **(always 'Db2 I')**	NA	NA	Server OS Client/Server → DB Protocol

Interface – Detailed breakdown of QVC5001

	QVC5001 values	Guardium detail (Client/Server→Network Protocol)
ODBC	IBM i Access for Windows:ODBC:07010000 (7.1)	ODBC:07010000
OLE DB	IBM i Access for Windows:IBMDASQL OLE DB Provider:07010003	OLE DB:07010003
.NET	IBM i Access for Windows:.NET Provider:07010003	.NET:07010003
Toolbox JDBC	IBM Toolbox for Java:JDBC:07010003	ToolBoxJDBC:07010003
DB HSVR	empty strings for each of these registers	
Native JDBC	IBM Developer Kit for Java JDBC Driver:JDBC:06010014 (6.1) IBM Developer Kit for Java JDBC Driver:JDBC:07010004 (7.1) IBM Developer Kit for Java JDBC Driver:JDBC:07010004	JDBC:06010014
Native SQL CLI	CLI:CLI:0601000000 (6.1) CLI:CLI:0701000000 (7.1) CLI:CLI:0701000000 (7.2)	CLI:0601000000
Db2 Connect ODBC		
DRDA	Db2/I5OS:060100 (6.1) Db2/I5OS:070100 (7.1) Db2/I5OS:070200 (7.2)	Db2/I5OS:060100
DataDirect	ODBC4Db2 or DDT (Can be mixed up)	
IBM Advanced Payment System	Q4680	
IBM CICS	QCICS	
IBM JCC (Type 4 Only)	QDb2/JVM:vvvvvv	
IBM LUW (JCC Type 2, CLI/ODBC/.NET/etc)	QDb2/xxx:vvrrff	
IBM Netview/PC	QNETPC	
IBM z/OS	QDb2:vvrrmm	
Microsoft HIS	MSEIDRDA	
Oracle	ORA	
StarSQL	WIN3X	
XDB System	XDB	

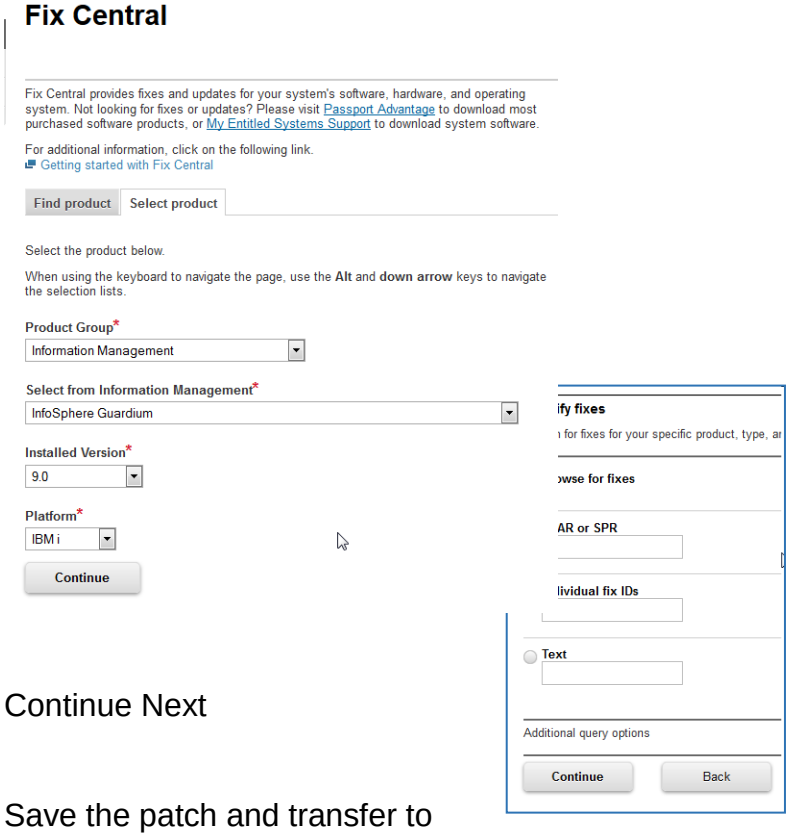
Appendix A - IBM i Command Cheat Sheet

Q: How to determine the S-TAP Service level:

CALL QP2TERM
cd /usr/local/guardium/
strings -a istap | grep itap_version

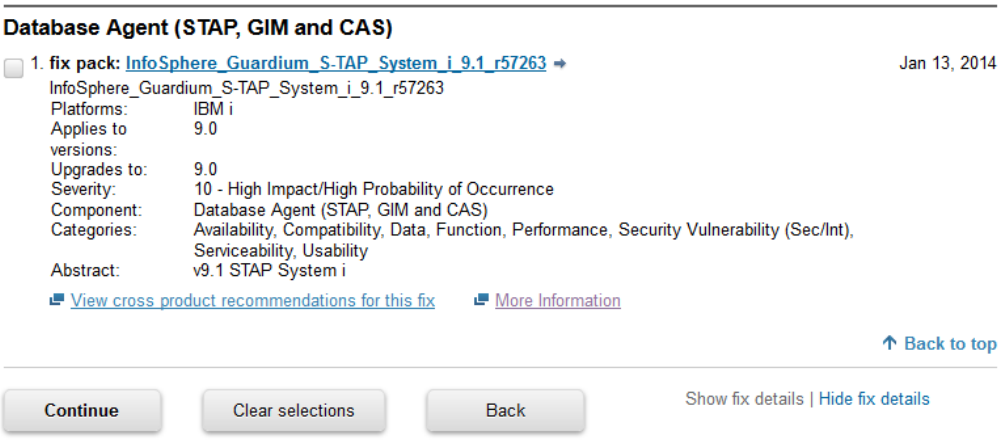
Q: How to determine the latest S-TAP patch level available:

http://www-933.ibm.com/support/fixcentral/



Continue Next

Save the patch and transfer to the IBM i.



Q: How to determine if the install was successful:

Install program example output:

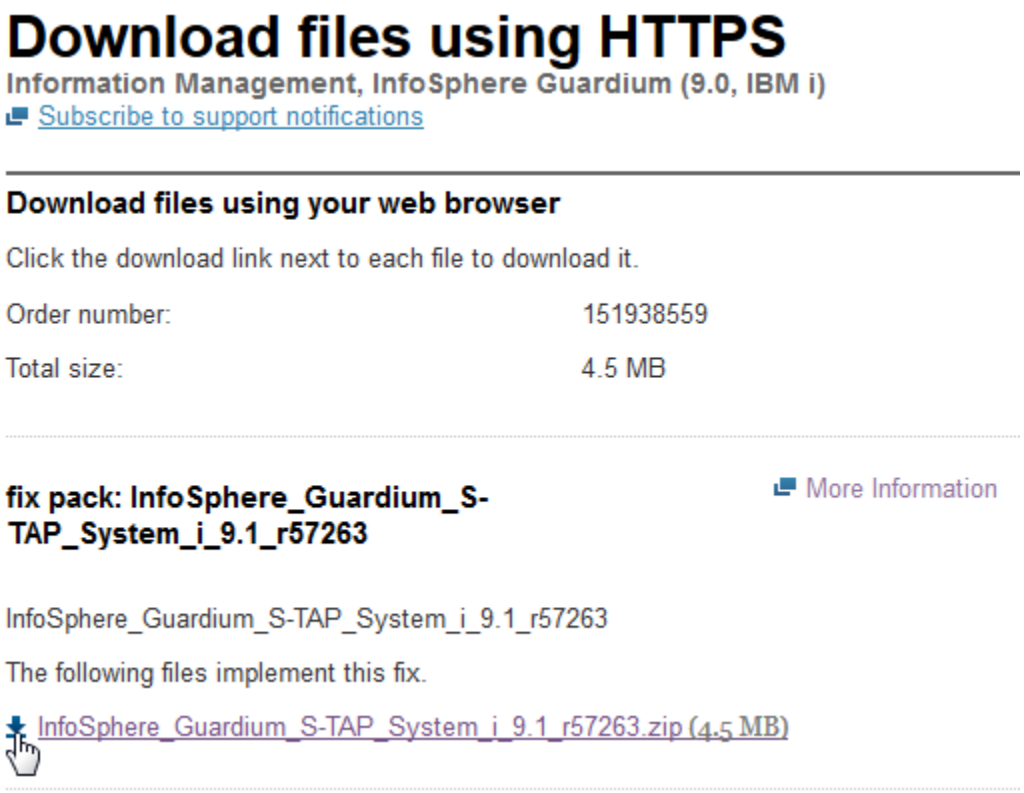
```
> guard-itap-9.0.0_r57263_v90_1-aix-5.3-aix-powerpc.sh   9.5.37.90
   21940+0 records in.
   43880+0 records out.
   Checking whether Audit Server has been stopped. (This call may take
a minute.
    Please wait.)
   Creating /usr/local/guardium directory.
   CPC221B: Object changed.
   Starting Audit Server.
   Installation successfully ended.
   $
```

Note – if the client does not see the "Installation successfully ended" message, have them capture all messages generated by the install program.
Also, have them look in **/usr/local/guardium/** for any error text files.

If they did not capture the install message, examine:
/usr/local/guardium/install_out.txt

install_out.txt should contain:

```
************Beginning of data**************
Db20000I  THE SQL COMMAND COMPLETED SUCCESSFULLY.
 ************End of Data*********************
```



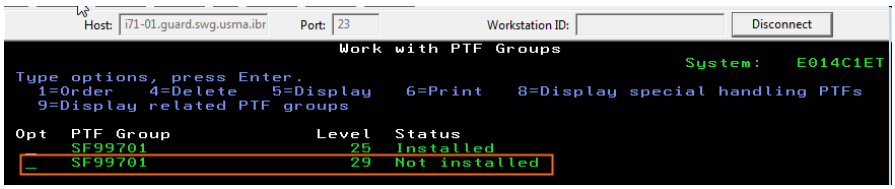
Recommended Db2 for i Service level:

Look here:
https://ibm.biz/GuardiumDAMonIBMi

To display the Db2 for i Service level:

If using IBM i 6.1:
WRKPTFGRP PTFGRP(SF99601)
If using IBM i 7.1:
WRKPTFGRP PTFGRP(SF99701)
If using IBM i 7.2:
WRKPTFGRP PTFGRP(SF99702)
If the IBM i OS release level is unknown:
WRKPTFGRP

Look for the largest Level # with status = 'Installed'.



Q: Who is the configured Audit Server start user?

A: STRSQL NAMING(*SYS)
SELECT START_USER, A.* from qsys2/sysaudit A

Q: Does the start user have the required authorities?

A: select SPECIAL_AUTHORITIES
from qsys2/user_info, qsys2/sysaudit where USER_NAME = START_USER

Alternative approach:
DSPUSRPRF USRPRF(<start-user-name>) TYPE(*BASIC)

Q: Is the Filter RDB name configured?

A: > STRSQL NAMING(*SYS)
SELECT SUBSTR(FILTER_RDB,1,10) AS RDB,A.* from qsys2/sysaudit A

Q: What value should be used for Filter RDB?

A: Execute this IBM i Command and look for the *LOCAL database name
> WRKRDBDIRE

Q: How do I update the configured Filter RDB name?

A: Update the QSYS2/SYSAUDIT table directly on Db2 for i
Note: This name is case sensitive.

```
RUNSQL SQL('update qsys2/sysaudit set filter_rdb = ''<local-rdb-name>''')
COMMIT(*NONE) NAMING(*SYS)

RUNSQL SQL('CALL SYSPROC/SYSAUDIT_START_BATCH('''')')
COMMIT(*NONE) NAMING(*SYS)
```

Q: How do I capture the Audit server status?

A:
RUNSQL SQL('CALL SYSPROC/SYSAUDIT_STATUS()') COMMIT(*NONE)
CRTSAVF QGPL/SYSAUDSTS
SAVOBJ OBJ(SYSAUDSTS) LIB(QTEMP) DEV(*SAVF) SAVF(QGPL/SYSAUDSTS)

For instructions on getting the SAVF to your PC see the following document:
N1017260: FTP Save Files between PC and IBM OS/400 or IBM i5/OS
Found at: http://www-01.ibm.com/support/docview.wss?uid=nas8N1017260

For how to send the data in, you can respond to this email and include the documents as attachments. Or, see the following Document:
N1019224:
MustGather: Instructions for Sending Data to IBM i Support found at:
http://www-01.ibm.com/support/docview.wss?uid=nas8N1019224

SQL statements that might be useful

```
-- Find the user profile being used to run the i-STAP Audit Server
select START_USER from qsys2.sysaudit
  where start_user is not null;

-- Review the Guardium iS-TAP TCP/IP connection to the Guardium Collector
select * from qsys2.netstat_info
 where remote_port = 16016;

-- Find group profiles that include *ALLOBJ special authority
with groups(grp) as (
```

```
 select distinct(group_profile_name) from qsys2.group_profile_entries
)
select * from qsys2.user_info, groups where authorization_name = grp
   and special_authorities like '%ALLOBJ%';

-- Review the IBM i HOST_NAME
select * from qsys2.tcpip_info;
select * from sysibmadm.env_sys_info;
-- Note: if you observe UNKNOWN, then you can use CFGTCP option 12 to see the
configuration

SELECT *
   FROM qsys2.sysaudit;

-- Induce a problem via a bad entry type
     UPDATE qsys2.sysaudit
        SET filter_audit_entry_types =
           'XX AD AF CA CO CP DO GR OM OR OW PG PW RA RO RZ SV ZC CD';

-- Fix the problem
  UPDATE qsys2.sysaudit
     SET filter_audit_entry_types =
        'AD AF CA CO CP DO GR OM OR OW PG PW RA RO RZ SV ZC CD';

-- Fix the problem (7.2 and up)
  UPDATE qsys2.sysaudit
     SET filter_audit_entry_types =
        'AD AF CA CO CP DO GR OM OR OW PG PW RA RO RZ SV ZC CD AX';

CALL sysproc.sysaudit_status();
SELECT *
    FROM qtemp.sysaudsts;

-- Find the STAP audit server jobname
WITH audit_config(su)
     AS (SELECT start_user
            FROM qsys2.sysaudit
            WHERE start_user IS NOT NULL)
SELECT job_name FROM audit_config,
               TABLE(qsys2.active_job_info('NO', '', '', su)) j
            WHERE job_name LIKE '%GDAUDIT';

--
-- description: Assess whether the audit server job is healthy
--
-- Note: you want to see zero rows returned
--
WITH audit_config(su)
     AS (SELECT start_user
            FROM qsys2.sysaudit
            WHERE start_user IS NOT NULL),
            audit_server(jn)
     AS (SELECT job_name
            FROM audit_config,
                 TABLE(qsys2.active_job_info('NO', '', '', su)) j
            WHERE job_name LIKE '%GDAUDIT')
     SELECT *
        FROM audit_server,
             TABLE(qsys2.joblog_info(jn)) s
        WHERE message_type = 'ESCAPE' AND MESSAGE_ID <> 'CPF436D';

select * from qsys2.group_profile_entries where group_profile_name = 'PRIVIDS';
```

Closing words

Contact Scott if you have ideas that would improve this document.