"Why doesn't the data from my uSKF File show up?"
May 12, 2012
FOR INTERNAL TSG USE ONLY

INTRODUCTION

The purpose of this article is to

- Provide an overview of the @ptitude Analyst Thin Client Transfer (@TCT) system
- Describe common issues with @TCT- File Based mode that prevent data from being processed
- How to recognize these issues and
- Steps that can be taken to fix the issues and ultimately get collected data to appear in the database.

The issues described in this article are:

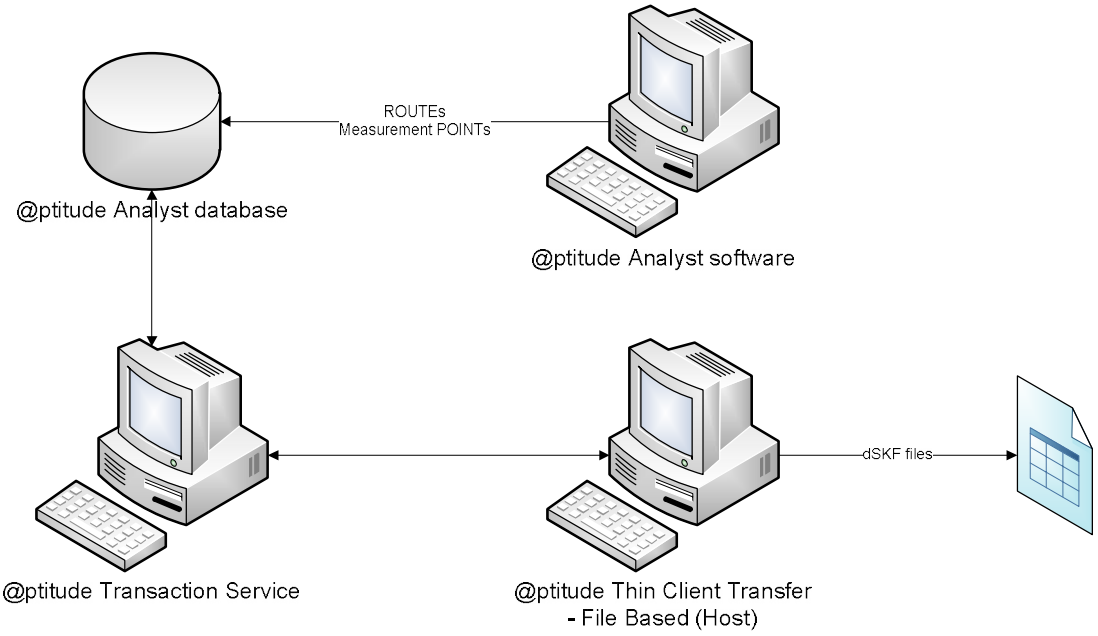
SYSTEM DESCRIPTION

The following describes how the @ptitude Thin Client Transfer (File based) system works:

Step 1 – creation of dSKF files

1. @ptitude Analyst (@A) is used to create measurement POINTs and then ROUTEs that contain those POINTs.
2. @TCT (Host) is used to create download (dSKF) files for use with the Microlog CMVA 65, GX 70 & 75 and the AX-80.

PLEASE NOTE: WHEN THE dSKF FILES ARE CREATED, THE CONFIGURATION INFORMATION IN THE FILES IS TIED DIRECTLY TO THE DATABASE THAT CREATED THE FILES. IF THE DATABASE CHANGES (CREATION OF A NEW DATABASE FROM MAB FILES FOR INSTANCE) THIS INVALIDATES ALL dSKF FILES CREATED FROM THE OLD DATABASE.



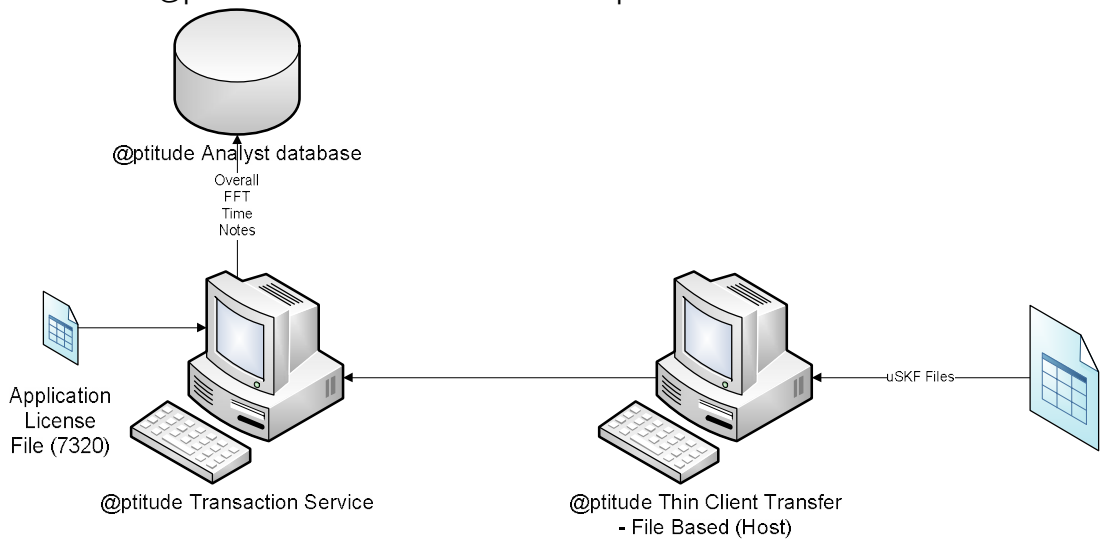
Step 2 – The dSKF files are delivered to a @TCT client computer. This can be done through email, copied to a USB flash drive, stored on a corporate shared network drive or ftp site, etc.

Step 3 – @TCT (Client) uses the dSKF file to configure a Microlog connected through the USB port.

Step 4 – Data is collected in the field, the Microlog is reconnected to the Client computer and then @TCT (Client) creates a uSKF upload file.

Step 5 – The uSKF files are delivered back to the @TCT host computer. This can be done through email, copied to a USB flash drive, stored on a corporate shared network drive or ftp site, etc.

Step 6 - @TCT (Host) is then used to read and process the data back into the database. Communication to the database is done through Transaction Service. An Application License File (ALF) with a CMSW 7320 (Thin Client Transfer – File) must exist on the @ptitude Transaction Services computer.



NOTE: IT IS IMPORTANT THAT THE SERVICE ENGINEER COPY THE ~GX FOLDER FROM THE MICROLOG TO HIS LAPTOP USING ACTIVE SYNC. THIS COPY SHOULD BE KEPT UNTIL CONFIRMATION HAS BEEN RECEIVED FROM THE DATA CENTER THAT THE MICROLOG DATA HAS BEEN UPLOADED SUCCESSFULLY.

TSG: WHEN TROUBLE SHOOTING AN ISSUE WITH PROCESSING A uSKF FILE, ALWAYS GET A COPY OF THE ~GX FOLDER FROM EITHER THE GX DIRECTLY OR FROM THE BACKUP COPY ON THE SERVICE ENGINEER'S LAPTOP.

TOOLS REQUIRED:

It will be useful to have available the following applications:

1. UploadViewer.exe – a tool developed by SKF CMC San Diego to view uSKF and MAUL upload files. It can be found at TSG-PLEASE PROVIDE COMMON DIRECTORY WHERE THIS CAN BE FOUND.
2. Mod_uSKF.exe – a tool developed by SKF CMC San Diego to stamp uSKF files with missing Status information (developed for JCI). It can be found at TSG-PLEASE PROVIDE COMMON DIRECTORY WHERE THIS CAN BE FOUND.
3. The latest Microlog GX and AX drivers that can be found at TSG-PLEASE PROVIDE COMMON DIRECTORY WHERE THIS CAN BE FOUND.
4. The latest version of @TCT with File Based and Direct keys.
5. The latest version of @ptitude Analyst.
6. Database access tools such as Oracle SQL Plus and Microsoft SQL Server Management Studio.
7. A low level hex file editor. The program used in this document is a freeware editor produced by Neo. The installation file can be found at TSG-PLEASE PROVIDE COMMON DIRECTORY WHERE THIS CAN BE FOUND.

ISSUES DISCOVERED TO DATE: DESCRIPTION; HOW TO IDENTIFY; PROBABLE ROOT CAUSE; AND FIX:

To date, the following issues have been encountered and work-around developed:

1. Invalid file status in TCT Import Upload File dialog

SYMPTOM:

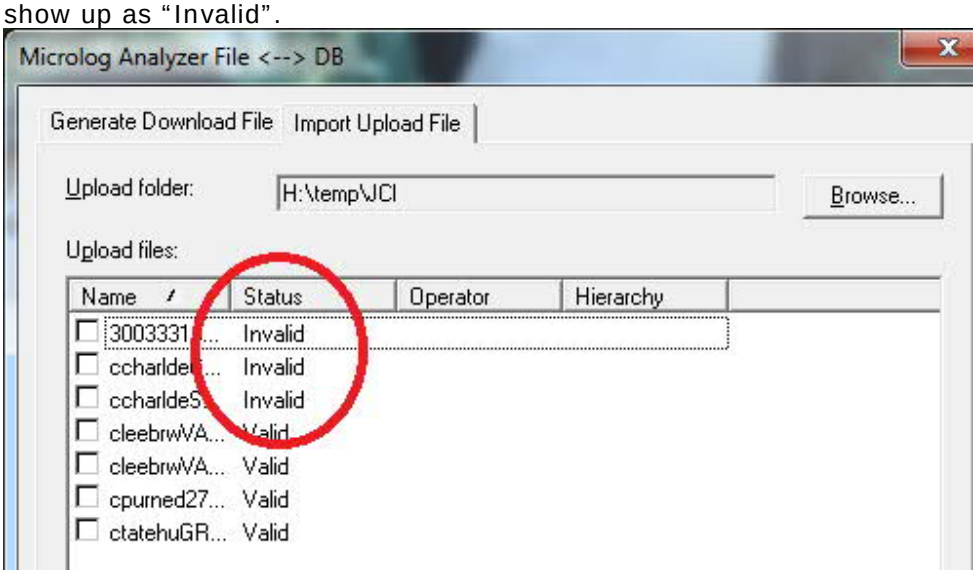
uSKF file shows up as "Invalid" in @TCT (HOST)

ISSUE:

uSKF files missing a "device label" such as MLOGPI or MARLIN

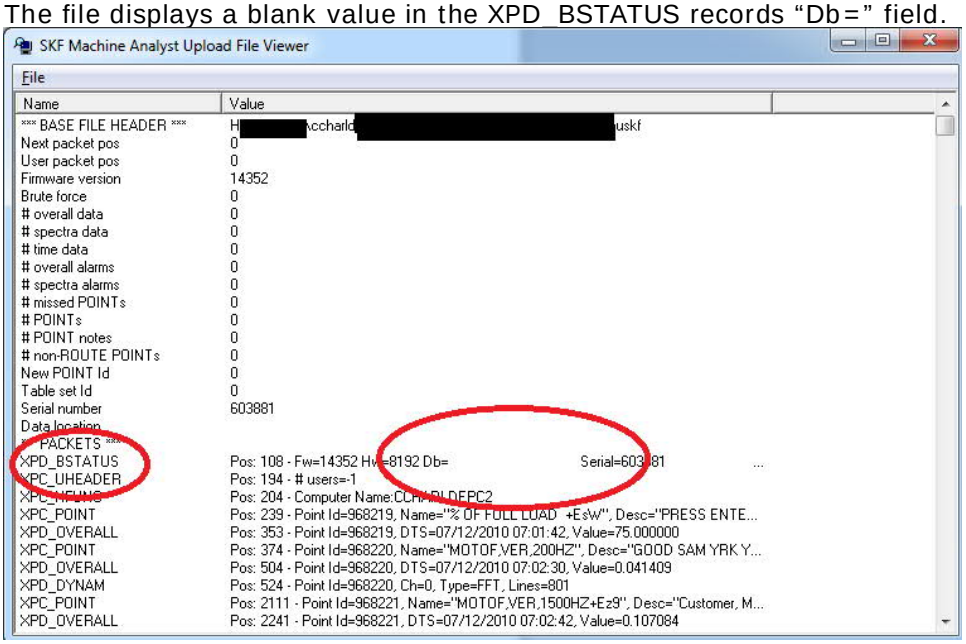
IDENTIFICATION (a):

On the Import Upload File tab of @TCT (Host), the Status of the uSKF files show up as "Invalid".



IDENTIFICATION (b):

The file displays a blank value in the XPD_BSTATUS records “Db=” field.

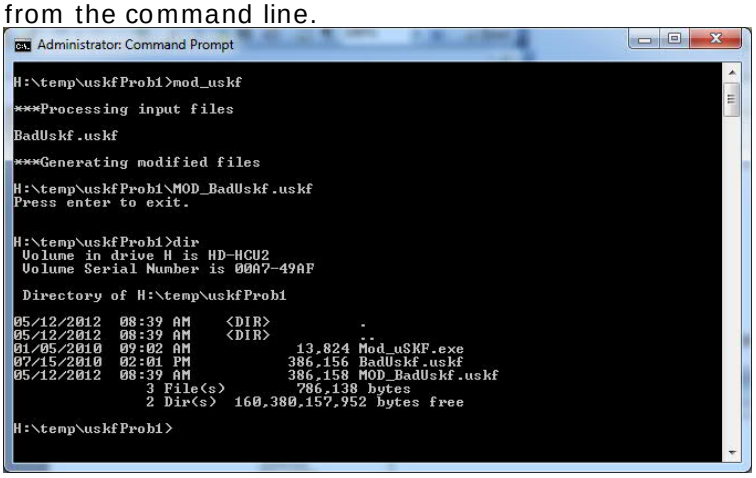


PROBABLE CAUSE:

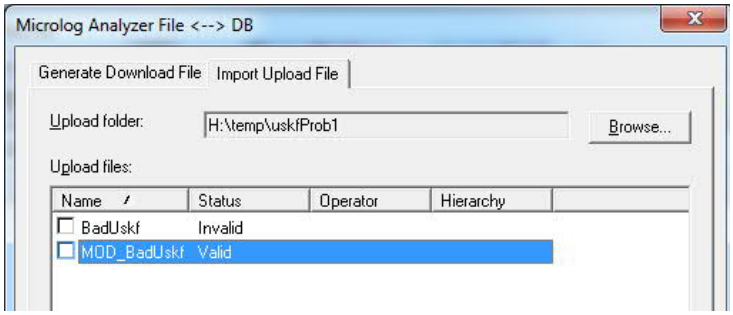
uSKF Created from older versions of @TCT predecessor “Remote ROUTE”.

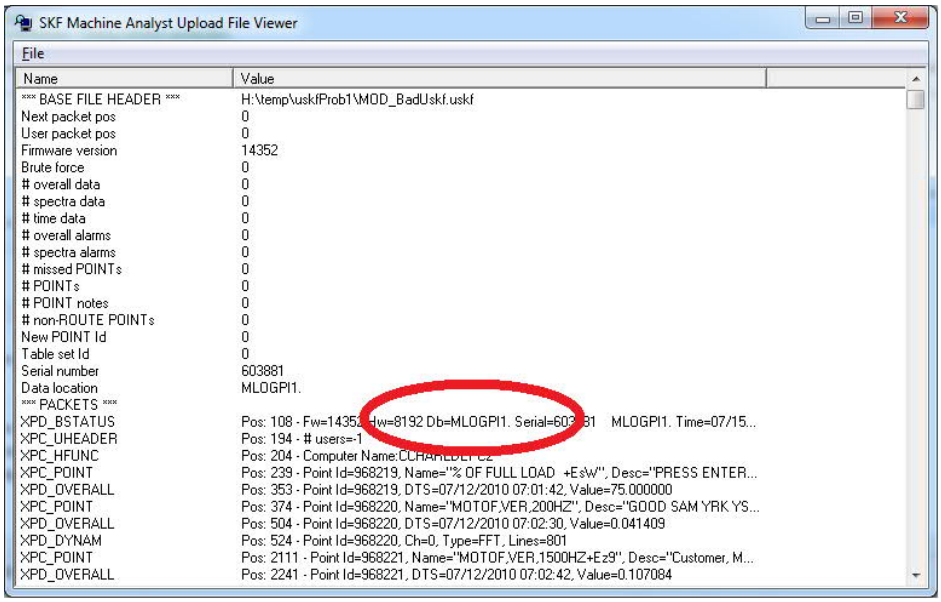
FIX:

Place the file(s) in a directory with Mod_uSKF.exe and run Mod_uSKF.exe from the command line.



The resulting file: MOD_filename.uskf will be correct:

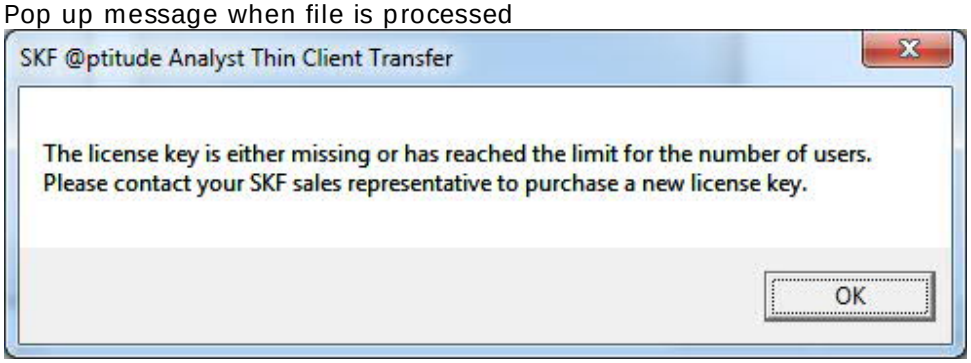




2. All licenses used.

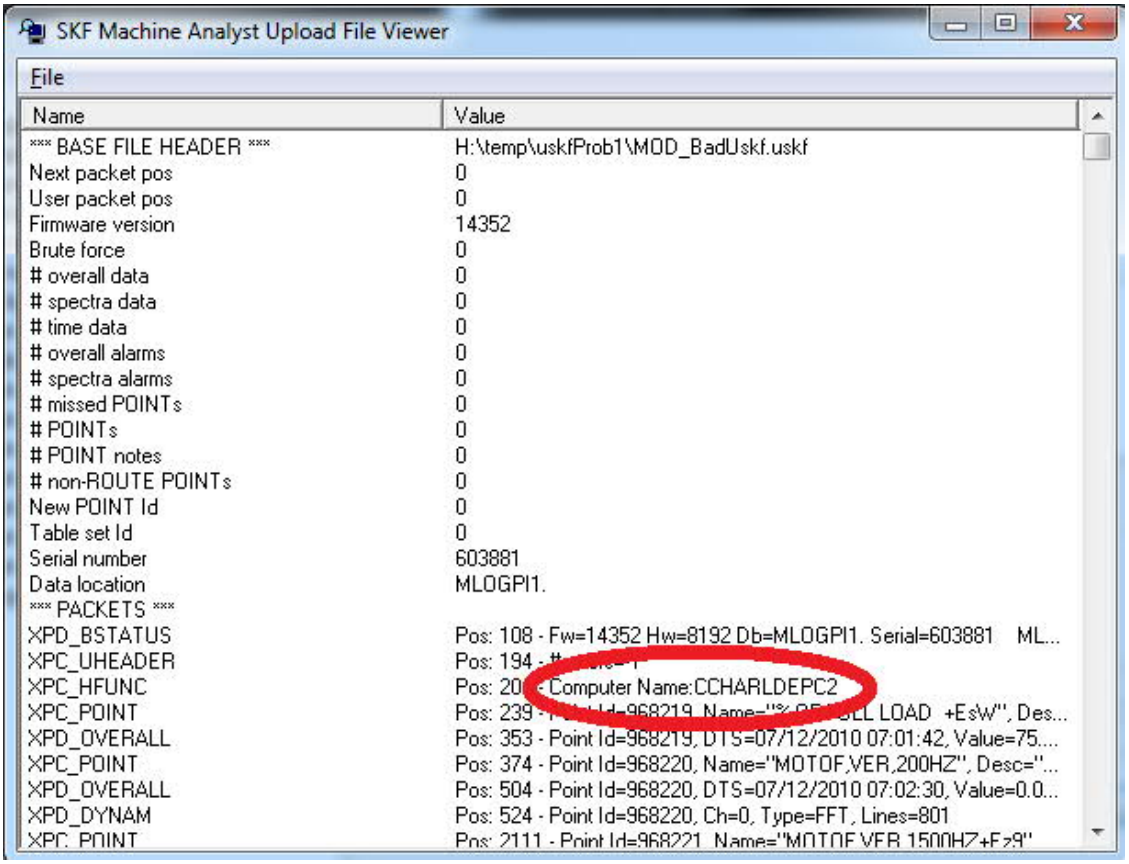
SYMPTOM:

Pop up message when file is processed



ISSUE:

A new unique computer ID is contained in a uSKF file that is being processed after the software has reached the limit of the purchased CMSW 7320 licenses.

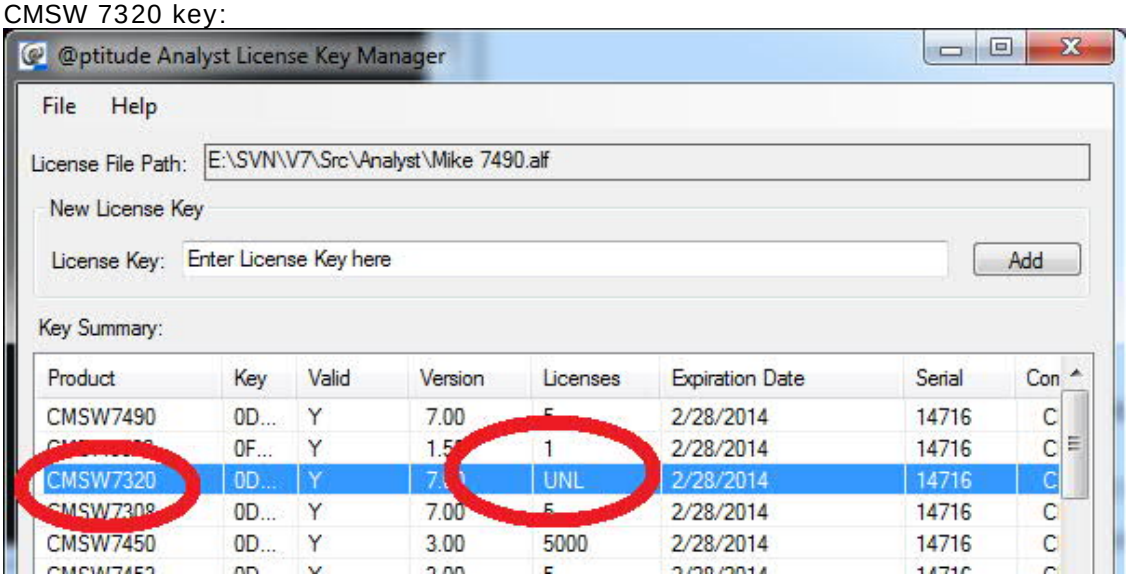


IDENTIFICATION (a):

The pop up message being displayed.

IDENTIFICATION (b):

Compare the number of licenses displayed in License Key Manager for the CMSW 7320 key:



against the count of the number of computer id records currently stored in the database by running the following script:

```
select COUNT(PrefId) from PREFERENCE where PREFID like
'%File%Upload%ClientName%';
```

NOTE: NEVER PROVIDE THIS SCRIPT TO A CUSTOMER OR UNAUTHORIZED SKF PERSONNEL.

PROBABLE CAUSE:

A previously used laptop has been put out of commission and a new laptop has just started to be used. New service personnel have been added to the organization.

FIX:

The licenses can all be cleared out and the counter restarted by deleting all current keys with the SQL script:

```
delete from PREFERENCE where PREFID like
'%File%Upload%ClientName%';
```

NOTE: NEVER PROVIDE THIS SCRIPT TO A CUSTOMER OR UNAUTHORIZED SKF PERSONNEL.

If a specific computer name is known that has been taken out of service, then the script:

```
delete from PREFERENCE where PREFVAL = 'computerName';
```

will target the specific computer.

If the service organization has outgrown their current licenses, then a new license can be ordered to bump up the number of authorized users.

3. Missing (unavailable) License

SYMPTOM:

A: I do not see options for DAD ←→ File or File ←→ Db under Transfer \ Microlog Analyzer.

B: I do not see the option for DAD ←→ Db under Transfer \ Microlog Analyzer

IDENTIFICATION:

User opens up @TCT and selects Transfer \ Microlog Analyzer and does not see the options he expects to see.

PROBABLE CAUSE:

For the proper key (for (A) - CMSW 7320 key; for (B), CMSW 7321 key) has not been installed on the Transaction Service computer; @TCT is not configured to connect to the right Transaction Service; Transaction Service is not running or hung.

FIX:

Verify through the License Key manager on the Transaction Service computer that the proper license is installed. Verify through the SKF @A Configuration Tool on the @TCT computer that it is pointing to the right Transaction Service. Stop and restart the Transaction Service, then start and restart @TCT. In the case of the direct connect (CMSW 7321 key), verify that all licenses are currently not in use.

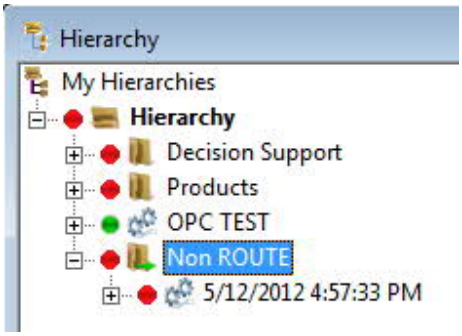
4. Brute force flag set

SYMPTOM:

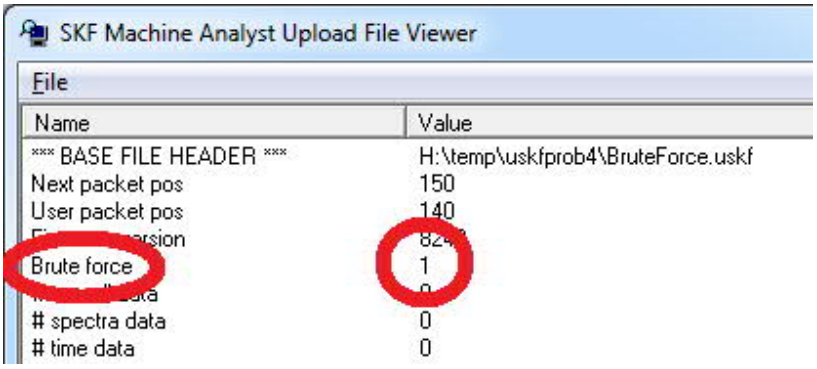
uSKF file processes, but no data appears where the user expects it to.

IDENTIFICATION:

In @ptitude Analyst there is a new Machine under Non ROUTE with the upload date and time.



A review of the uSKF file using the UploadViewer shows the Brute Force flag set:



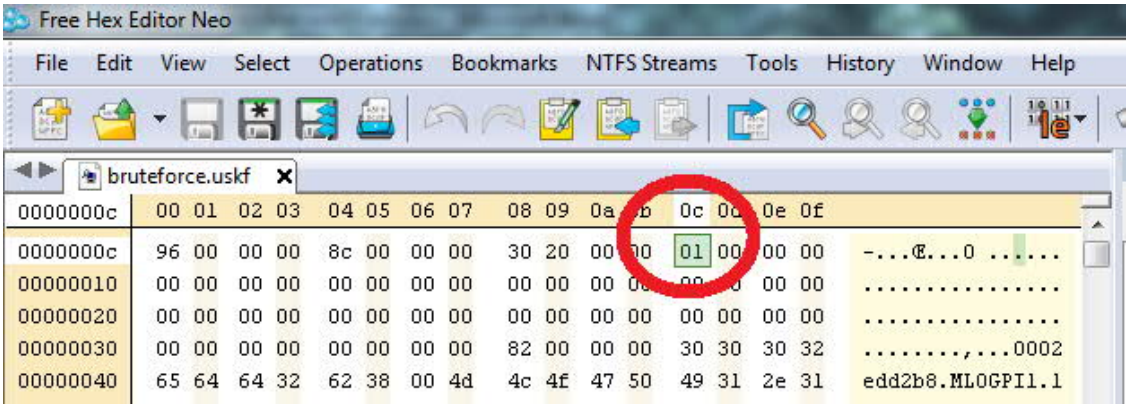
PROBABLE CAUSE:

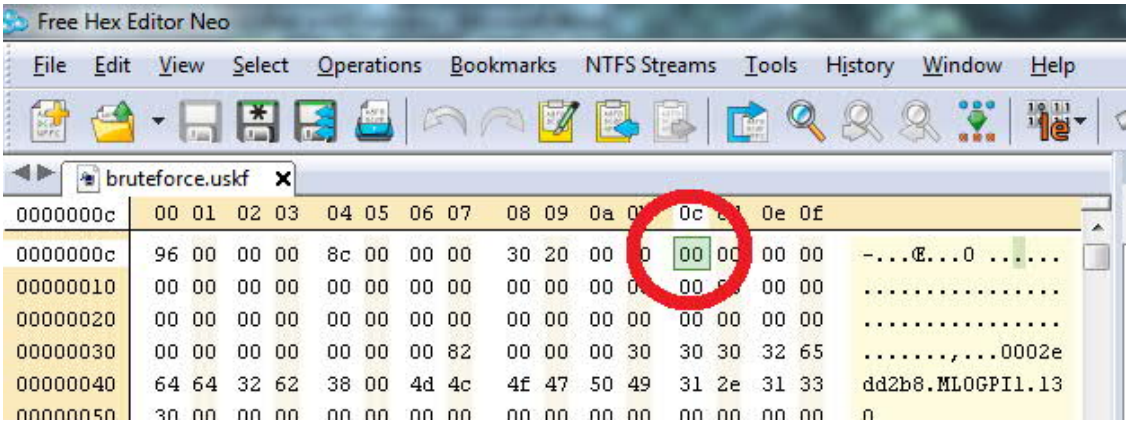
User selected the Brute Force option when they used the Generate Upload File function on the Client Computer while in the Transfer \ Microlog Analyzer \ DAD ←→ File dialog.

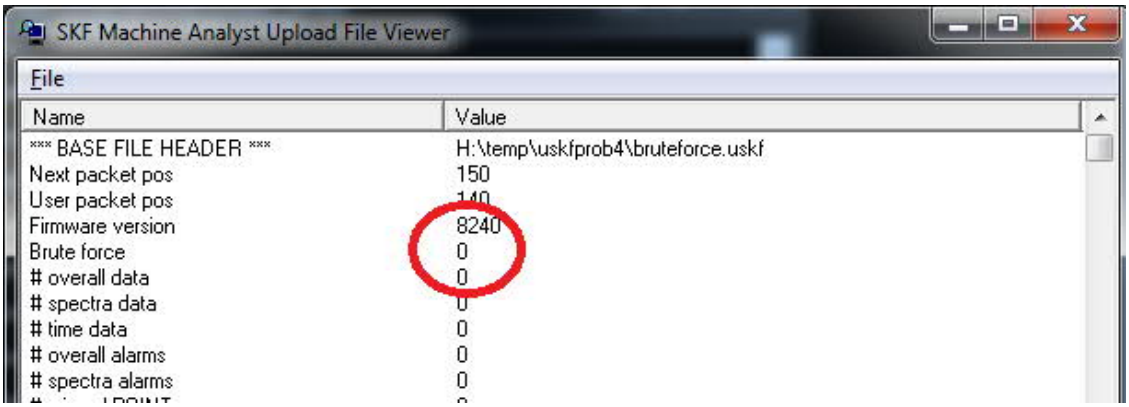
FIX:

The easiest way to fix this is to take the ~GX folder and re-generate the uSKF file without the Brute force flag set.

Assuming that all that is left is the uSKF file a Hex editor can be used to set the flag for Brute force to false:


Free Hex Editor Neo
File Edit View Select Operations Bookmarks NTFS Streams Tools History Window Help
bruteforce.uskf
0000000c 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f
0000000c 96 00 00 00 8c 00 00 00 30 20 00 00 01 00 00 00 -...Œ...0 .....
00000010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
00000020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
00000030 00 00 00 00 00 00 00 00 82 00 00 00 30 30 30 32 ....,...0002
00000040 65 64 64 32 62 38 00 4d 4c 4f 47 50 49 31 2e 31 edd2b8.MLOGPI1.1


Free Hex Editor Neo
File Edit View Select Operations Bookmarks NTFS Streams Tools History Window Help
bruteforce.uskf
0000000c 00 01 02 03 04 05 06 07 08 09 0a 0b 0c 0d 0e 0f
0000000c 96 00 00 00 8c 00 00 00 30 20 00 00 00 00 00 00 -...Œ...0 .....
00000010 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
00000020 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 ....
00000030 00 00 00 00 00 00 00 82 00 00 00 30 30 30 32 65 ....,...0002e
00000040 64 64 32 62 38 00 4d 4c 4f 47 50 49 31 2e 31 33 dd2b8.MLOGPI1.13


SKF Machine Analyst Upload File Viewer
File
Name
Value
*** BASE FILE HEADER ***
H:\temp\uskfprob4\bruteforce.uskf
Next packet pos
150
User packet pos
140
Firmware version
8240
Brute force
0
overall data
0
spectra data
0
time data
0
overall alarms
0
spectra alarms
0

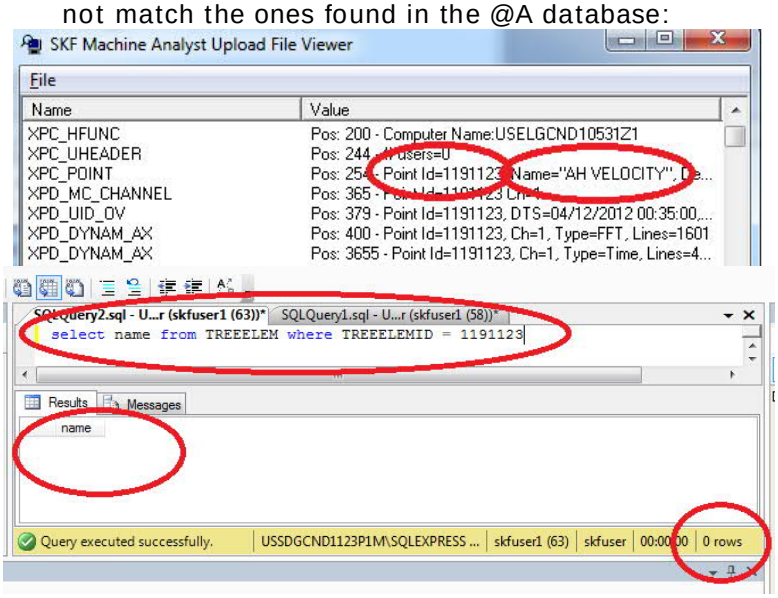
5. Mismatching POINT IDs

SYMPTOM:

uSKF file processes, but no data appears where the user expects it to.

IDENTIFICATION:

- File processes, no Machine appears in the Non ROUTE set;
- The number of POINTs processed in the Event Log does not match the number showing up in the View uSKF file function in @TCT \ File ←→ DB Import Upload File tab;
- Unusual messages in the Event Log such as “Unable to process packet 37”;
- The POINT Names for the Point IDs identified in the uSKF file do not match the ones found in the @A database:



PROBABLE CAUSE:

The Microlog was configured from a dSKF file that was generated from a different database than the one currently being used.

FIX:

NOTE: THIS IS A VERY SERIOUS PROBLEM. 1st off, if the ~GX folder still exists, a new dSKF file can be generated, then loaded into the GX unit to generate a proper uSKF file which can then be uploaded and processed into the @A system. If the ~GX folder does not exist, then it might be possible to change the brute force flag to “1” (see issue 4) and then use the data in the Non ROUTE Machine to capture the data so it can then be placed in an appropriate location.

The administrators of the system will need to identify which dSKF files need to be replaced so that this problem doesn’t continue. In addition, some of the data uploaded may have been loaded into the wrong POINTs. Some database maintenance may need to occur to identify the readings that are incorrect and remove them. This issue might be rectified by restoring from a backup, or a custom script might have to be written to remove the erroneous data.

6. uSKF file is corrupt

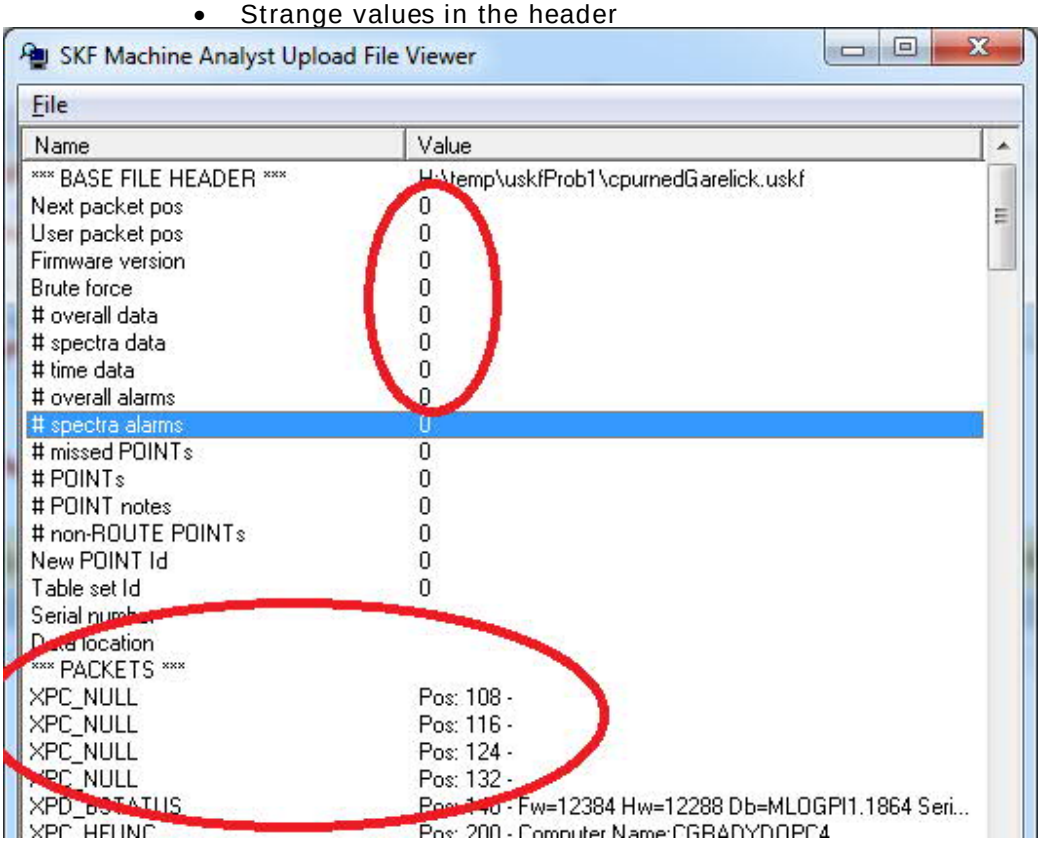
SYMPTOMS:

@TCT (Host) might give a message like “File too large”.
File may show up in TCT as Invalid.
File may fail to process.

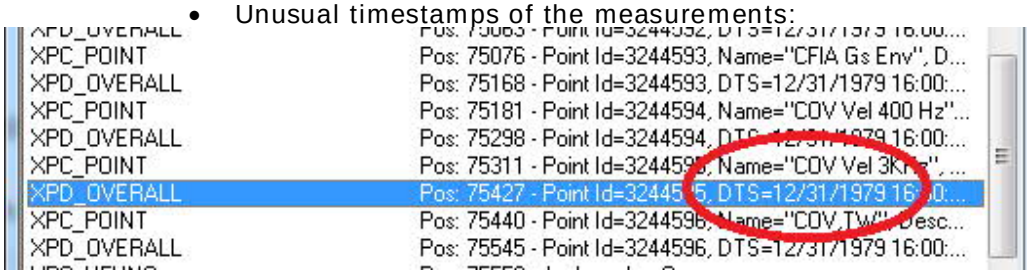
IDENTIFICATION:

Viewing the file in Upload File Viewer shows unexpected results:

- Strange values in the header



- Unusual timestamps of the measurements:



PROBABLE CAUSE:

Unknown. There are many paths for the file to become corrupt. The data on the Microlog might be corrupt; the uSKF file could have become corrupt by a glitch during transfer from the Microlog to the uSKF file; the uSKF file may be sitting on a bad sector on the client computer; the file may have become corrupt during transmission back to the data center.

FIX:

About the only way to fix this issue is to hope that the ~GX folder is not corrupt and then try to re-create the uSKF file using @TCT Client.