UNIX Scripting

Important point

1.Requirement the code should not have the hardcode any file name or path and it should be declared as variable within the script

2.All the tasks should be written as functions

Functions for alert raising, error handling, db2 logging and inserting records as mentioned below, informatica ETL process, email etc.

3. code should be readable and commented as needed for easy understanding.

A new UNIX script, fileLoader.sh script will be developed for the purpose of file processing. The script will reside in the $InfaHome/Scripts directory and will be executed via Autosys. The script will have the following functionality:

1. Accept the source.target.file ID of the file name in as parameter $1 and assign to $FEED\_NAME.  Assign $2 to the $FILE\_NUM. If it does not exist, default to 01.

* so my script is supposed to be parameter driven and it should run as
* ./fileLoader.sh VPAS.ANNTSL. SUBACCOUNT 01
* So $FEED\_NAME will be VPAS.ANNTSL. SUBACCOUNT
* $FILE\_NUM will be 01

1. Using $FEED\_NAME, append a “.txt” value to ensure file exists in the /apps/informatica/server/infa\_shared /Inbound directory.  If the file does not exist, call the ALERT function to raise an error and exit the script with a non-zero return code.  If the file does exist, loop through the file to pick up the names of the expected contents which are the compressed data file and the control file.
2. Once .txt file is detected, copy the .tar file and .txt file to the /apps/informatica/server/infa\_shared /Archive directory and append a timestamp value to avoid the file from being overwritten.
3. Also Copy the files to the /apps/informatica/server/infa\_shared /Temp directory and untar the .tar file

* The tar file(VPAS.ISGHUB.SUBACCOUNT.tar) will have 2 files which are
* Details file - VPAS.ISGHUB.SUBACCOUNT.20180425.01.csv
* Control file - VPAS.ISGHUB.SUBACCOUNT.20180425.01.ctl

1. Details File validation on the .csv file(VPAS.ISGHUB.SUBACCOUNT.20180425.01.csv)
   1. Validation Rules:
2. As a validation step, compare the $FEED\_NAME.txt, tar file, and control file hexadecimal fingerprints to the derived value. The hex validation should be applied to all pertinent files to ensure their integrity. The derived hex value can be obtained by running the following command: /usr/bin/openssl dgst –sha256 <FEED\_NAME>
3. For all the validation records if it pass then continue with the rest of the execution and if it does not satisfy the requirement then notify via email and log in the error log and also write in the auidit table
4. Validate if the name/date of the .csv file and .ctl file is same
   * + 1. ie. below two files should have same name and date
       2. Details file - VPAS.ISGHUB.SUBACCOUNT.20180425.01.csv
       3. Control file - VPAS.ISGHUB.SUBACCOUNT.20180425.01.ctl
5. Validate if the VPAS.ISGHUB.SUBACCOUNT.20180425.01.csv has the Header and Footer records
6. Validate date in the Header record if it is formatted if it is YYYYMMDD Format
7. check for the row count of the .csv file to match with that of the row count value in the .ctl file
8. Once the above all validations are successful then move the below Files to the /apps/informatica/server/infa\_shared/SrcFiles directory and remane to generice name
   1. Details file - VPAS.ISGHUB.SUBACCOUNT.20180425.01.csv
   2. Control file - VPAS.ISGHUB.SUBACCOUNT.20180425.01.ctl
   3. VPAS.ISGHUB.SUBACCOUNT.done

example

VPAS.ISGHUB.SUBACCOUNT.20180425.01.csv has to be changed to VPAS.ISGHUB.SUBACCOUNT.csv

VPAS.ISGHUB.SUBACCOUNT.20180425.01.ctl has to be changed to VPAS.ISGHUB.SUBACCOUNT.ctl

1. Remove the files from the Temp folder.
2. Write any standard out and standard error messages to the $InfaHome/Log file directory as fileLoader.<$Feed\_NAME>.<YYMMDDHHMMSS>.log.
3. Write a function to email to a group of users upon success and failure
   1. Example email a,b,c if success
   2. Email d,e if failure
   3. So we have to have this email function separate and call as needed.
4. Create a function within the script to insert a row to the TSL\_JOB\_ERROR table when any of the following conditions are met:
   * The file fingerprint, represented by the hexadecimal value provided on the file from VPAS, is different than the value received when running the openssl dgst command.
   * The record count within the control file does not match the actual record count on the file
   * Any filename listed within the control file do not exist.
5. When processing successful and all validations have passed, insert a row with key values into the TSL\_JOB\_AUDIT in db2 database table with an event code of 100, a message describing the failure, along with a status code of ‘f’.

10. Create a new function designed to run ETL processing using the File ID. Workflow names will have the following convention:

o wf\_<source>\_STG\_<fileID>

o Example: wf\_VPAS\_STG\_SECANDREIN

10 Pass in a session parameter file which includes the $FILE\_NUM value. This will be used in the CREATE\_PARM\_FILE mapping. Merge session and workflow parameters to make all values visible to the workflow execution.