

Procedure documentation

Assessment candidate : Sriram Radhakrishna

Date : 28th of May, 2022

Aim : To convert the time zones of the event_timestamps in the given CSV files from IST to UTC for compatibility with Logstash & Kibana and push the processed files to Logstash.

Algorithm :

1. Take a CSV file.
2. Convert the 'event_timestamp' entries to UTC from IST.
3. Convert the date/time format into one that is compatible with the platform.
4. Go back to step 1 and iterate for all CSV files.
5. Manually prepare the .conf files for each push.
6. Push the data & configuration files from local to the VM.
7. Push the data using the .conf files from the VM to Logstash for each CSV file.

Implementation of Algorithm :

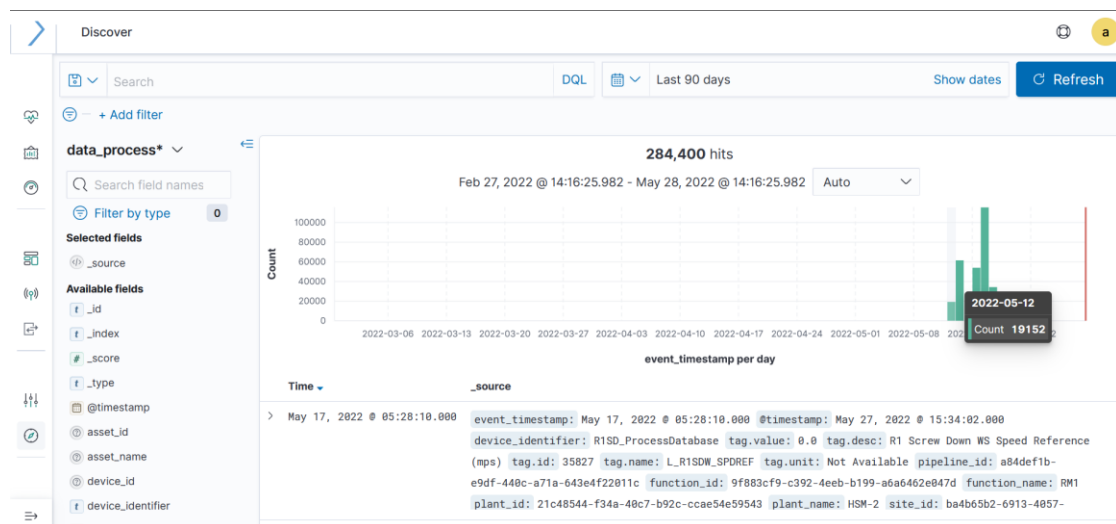
1. The CSV files were stored in a common directory.
2. A python script was written to iterate through them, change the time zone by subtracting 05:30 hours from the 'event_timestamp' cells to convert them from IST to UTC, and save them in an intermediate directory in each iteration.
3. The condition files were modified by a co-worker for removal of a specified substring, which were then updated in the intermediate directory.
4. A second python script was written to convert the same date/time objects from the files in the intermediate directory to a form that was compatible for onboarding the data to the platform. These were stored in a final directory with appropriate names.
5. Configuration files were manually prepared for each modified CSV with value changes to the path, sinceb_path, topic_id & bootstrap_servers keys as required by convention (further details described in the 'Sequence of events' section).

Sequence of events :

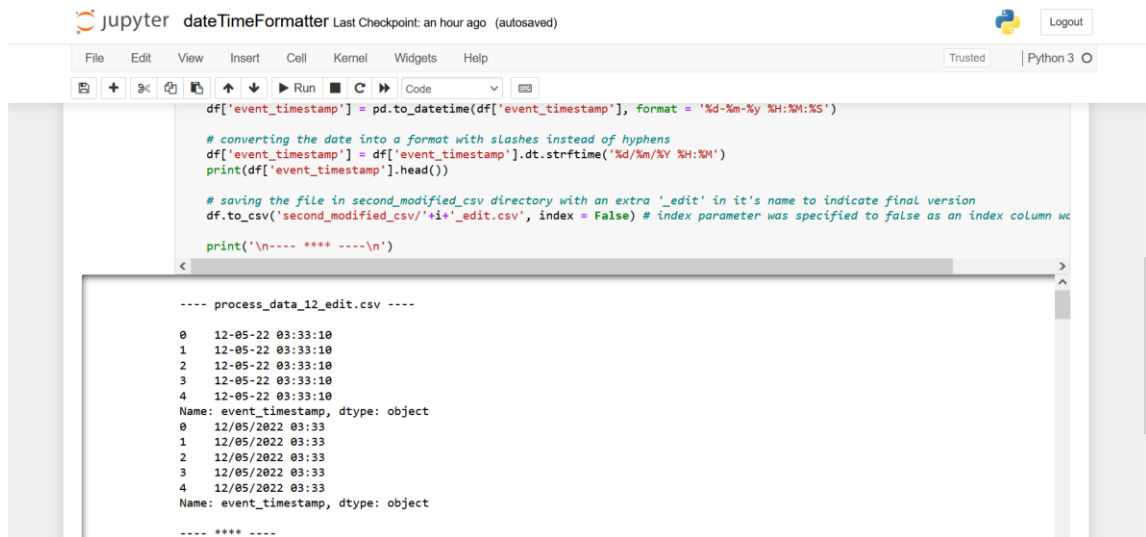
1. The CSV files given by Mr. Ashish Arun on Slack were stored in the 'og_csv' directory, which was not meant to be written into or modified as a precautionary measure.

2. The python notebook 'CSVDateTimeToUTC_allFiles.ipynb' was created for time zone conversion with another notebook 'CSVDateTimeToUTC_1file.ipynb' stored locally for prototyping & rough work.
3. As a first approach, an attempt was made to convert the 'event_timestamp' column of the dataframe imported from the CSV files into a pandas timestamp object, after which the tz_convert method was to be called to change the timezone, followed by conversion back into a series object. However, this caused some formatting issues as pandas automatically converted the date and time in the CSV file from '%d-%m-%y %H:%M:%S' to '%m-%d-%y %H:%M:%S'.
4. Realizing that this was a relatively convoluted approach leading to many ValueErrors due to date/time formatting problems, the approach was changed to one where the 'event_timestamp' column was imported in its exact datetime format and 5 hours & 30 minutes were manually subtracted from its records using the pd.Timedelta method. The entries were converted to '%d/%m/%Y %H:%M:%S' form before saving the CSV files into the 'first_modified_csv' directory with an '_edit' attached to the file name to indicate one modification.
5. The CSV files 'condition_data_de_edit.csv' and 'condition_data_edit.csv' from this directory were handed over to Ms. Sneha Tyagi for removal of the '_max' substring from the columns specified to her. When the files were returned, they were updated in the 'first_modified_csv' directory itself.
6. The first test pipeline for Test Site JK and Test Plant JK was started for the push (later to be found as an error point as the pipeline details weren't specified initially).
7. A .conf file was prepared to push the data from the CSV for process 12 as a test case in the build environment itself. The CSV file for 12-05 and its corresponding .conf file was pushed to the VM using a PUT request on SFTP. The push to logstash from there failed (as indicated by the lack of entries under 12-05-2022 on the data_process* index) due to the incompatibility of the 'event_timestamp' entries.
8. The page on data onboarding in 4pointX documentation was referred to at this point and the date/time format was changed to '%d/%m/%Y %H:%M' from the excel default of '%d-%m-%y %H:%M:%S' (an unsolicited conversion on the side of MS Excel for which the explanation could not be ascertained) using the newly created 'dateFormatter.ipynb' script, which followed a very similar structure to the 'CSVDateTimeToUTC_allFiles.ipynb' script, except that it had no further time zone changes implemented. Another '_edit' was added to the name to indicate that it was the final version.
9. A second push was attempted with partial failures due to already running instances of logstash, which was circumvented by changing the -path.data variable as shown in the demo.
10. A failure of this attempt led to further investigation of the other fields in the .conf file.
11. On the advice of Mr. Ashish Arun & Ms. Sneha Tyagi, I contacted Ms. Mayssa Nour, who reviewed the same file and pointed out that the kafka bootstrap server IP ('34.67.252.170.9092') taken from an example .conf file provided for reference earlier was incorrect. The correct IP was '34.135.68.182', which was backed up by the 'Build' page in the 4PointX documentation on confluence.

12. After changing this, a third push failure led to further inquiry which revealed that the topic_id field in the .conf file was incorrect. The 4PointX documentation specified that the pipeline ID was to be provided in this field.
13. Not knowing where to find the pipeline ID, the 4PointX build was explored and it was found under a document in the data_process* index under the discover tab on the platform with Ms. Sneha's help. This was verified by matching it with the topic_id field in the .conf file example given by Mr. Ashish earlier.
14. After repeated push failures attempted around 11:00 to 12:00 pm on 27th of May, further exploration of the 4PointX platform was done. Due to a lack of knowledge of the site and plant names being serviced by the platform, It could not be ascertained that the push failures were being caused by the fact that the pipeline wasn't running. It could not be enquired with anyone at the time due to the odd work hour and neither could anything be found on the 4PointX 'Onboarding Data' documentation page. Hence, the attempt was closed around 2:00 am on 28th of May.
15. Enquiry with Mr. Ashish Arun earlier today revealed the pipeline problem pointed out in point 6. The site was then set to 'JSW VJNR' and the plant to 'HSM-2'. The HSM-2 process pipeline was started and the graph corresponding to 12-05-2022 on the data_process* index was observed on the discover page of the 4PointX build. The count was at zero.
16. After pushing the data at roughly 11:30 am today, the count went up to 19152, from which it was concluded that the push worked, albeit with low confidence due to inconsistencies between the entry count in the CSV file and the platform, as observable from the screenshot below –



17. A confirmation attempt with the process data for 19-05-2022 was made which failed for reasons unknown at the moment. All guesses point to incorrect configuration file contents or push methods as the date/time formats were doubly verified by reiterating the entire process, as can be inferred from the final python script output screenshot below –



```
df['event_timestamp'] = pd.to_datetime(df['event_timestamp'], format = '%d-%m-%y %H:%M:%S')

# converting the date into a format with slashes instead of hyphens
df['event_timestamp'] = df['event_timestamp'].dt.strftime('%d/%m/%Y %H:%M')
print(df['event_timestamp'].head())

# saving the file in second_modified_csv directory with an extra '_edit' in it's name to indicate final version
df.to_csv('second_modified_csv/' + i + '_edit.csv', index = False) # index parameter was specified to false as an index column was not required
print('\n---- **** ----\n')
```

```
---- process_data_12_edit.csv ----
0    12-05-22 03:33:10
1    12-05-22 03:33:10
2    12-05-22 03:33:10
3    12-05-22 03:33:10
4    12-05-22 03:33:10
Name: event_timestamp, dtype: object
0    12/05/2022 03:33
1    12/05/2022 03:33
2    12/05/2022 03:33
3    12/05/2022 03:33
4    12/05/2022 03:33
Name: event_timestamp, dtype: object
---- **** ----
```

18. Following the procedures to obtain the topic ID and server IP learned earlier, the remaining config files were prepared and stored in the 'config_files' directory.
19. The remaining process, condition and FFT CSV files are yet to be pushed. Their corresponding .conf files can be found in the 'config_files' directory as follows :
 - a. condition_data_de_edit_edit.csv : process_config_cde.conf
 - b. condition_data_edit_edit : process_config_cd.conf
 - c. fft_data_edit_edit : process_config_fft.conf
 - d. process_data_12_edit_edit : process_config_p12.conf
 - e. process_data_13_edit_edit : process_config_p13.conf
 - f. process_data_16_edit_edit : process_config_p16.conf
 - g. process_data_17_edit_edit : process_config_p17.conf
 - h. process_data_18_edit_edit : process_config_p18.conf
 - i. process_data_19may_edit_edit : process_config_p19.conf

Notes :

- The assessment was started with knowledge of Python, Pandas, Elasticsearch, Git version control and the Linux command line interface.
- During the assessment, knowledge on the procedure of writing configuration files for logstash pushes, SSH & SFTP interfaces as well as familiarity with the 4PointX platform were gained by reading the documentation, assistance from mentors and practical experimentation.