Lucas Guichard

Cyber-Physical System Security – ASSIGNMENT (week 4)

Data Preprocessing with Web Server Access Logs

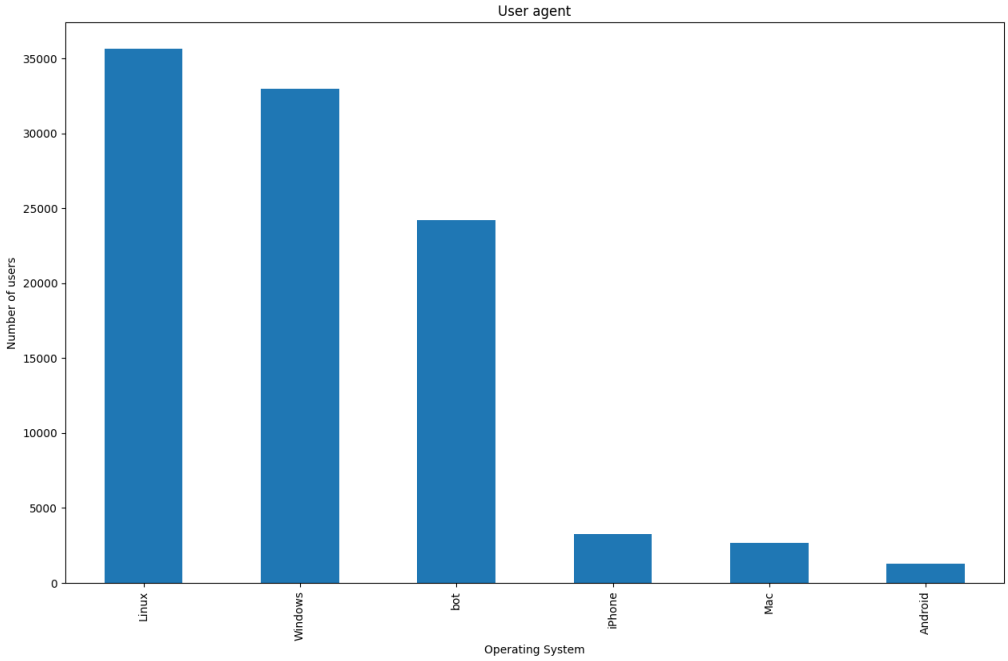
(Github: https://github.com/kounen/CPS_A4)

```
import pandas as pd
import re
import matplotlib.pyplot as plt
from tqdm.auto import tqdm
tqdm.pandas()

# Open log file and load its data into a list of string
with open('short_access.txt', 'r') as log_file:
    raw_data = [line for line in log_file.readlines()]

# Convert raw_data strings list into data frame
df = pd.DataFrame(raw_data, columns = ['log_line'])

## PRE-PROCESSING
# Use regex pattern to parse each log line
# We use axis=1 to loop on each line and not column
pattern = r'^(\S+) (\S+) (\S+) \[(.*?)\] "(.*?)" (\d+) (\d+) "(.*?)" "(.*?)"
"(.*?)"$'
df['client_ip'] = df.progress_apply(lambda x: re.match(pattern,
x['log_line']).group(1), axis = 1)
df['user'] = df.progress_apply(lambda x: re.match(pattern, x['log_line']).group(2),
axis = 1)
df['http_auth_user'] = df.progress_apply(lambda x: re.match(pattern,
x['log_line']).group(3), axis = 1)
df['timestamp'] = df.progress_apply(lambda x: re.match(pattern,
x['log_line']).group(4), axis = 1)
df['request'] = df.progress_apply(lambda x: re.match(pattern,
x['log_line']).group(5), axis = 1)
df['response_code'] = df.progress_apply(lambda x: re.match(pattern,
x['log_line']).group(6), axis = 1)
df['response_size'] = df.progress_apply(lambda x: re.match(pattern,
x['log_line']).group(7), axis = 1)
df['user_agent'] = df.progress_apply(lambda x: re.match(pattern,
x['log_line']).group(9), axis = 1)

# Drop columns containing always the same value
# Drop log_line column (parsing done so its content is not useful now)
# Now, only : ['client_ip', 'timestamp', 'request', 'response_code',
'response_size', 'user_agent']
columns_to_drop = [column for column in df.columns if df[column].nunique() == 1]
columns_to_drop.append('log_line')
df = df.drop(columns_to_drop, axis = 1)

## USER AGENT ANALYSIS
# Create new dataframe specific for this analysis
```

```
agent_df = pd.DataFrame()

# Add the user agent raw date collected from previous parsing
agent_df['raw_data'] = df['user_agent']

# Parse and store browser from this raw_data
agent_df['browser'] = agent_df.progress_apply(lambda x:
x['raw_data'].split('/')[0], axis = 1)

# Parse and store Operating System type
def get_os(agent_info: str):
    match = re.search(r'\((.*?)\)', agent_info)
    # Using only OS section
    if match:
        if 'Windows' in match.group(1):
            return 'Windows'
        elif 'iPhone' in match.group(1):
            return 'iPhone'
        elif 'Mac' in match.group(1):
            return 'Mac'
        elif 'Linux' in match.group(1):
            return 'Linux'
        elif 'Android' in match.group(1):
            return 'Android'
        # Barkrowler is a bot devoloped by eXenSa
        elif 'bot' or 'Barkrowler' in match.group(1):
            return 'bot'
        else:
            return '-'
    # If no match, no OS section detected so we use all the agent information
    else:
        if 'bot' or 'python' in agent_info:
            return 'bot'
        else:
            return '-'
agent_df['os'] = agent_df.progress_apply(lambda x: get_os(x['raw_data']), axis = 1)

# Count the number of occurrences of each OS
# Create a new dataframe from this computation
os_counts = agent_df['os'].value_counts()

## DISPLAY ANALYSIS
# Create a bar chart of the OS counts
os_counts.plot.bar()

# Set the chart title and axis labels
plt.title('User agent')
plt.xlabel('Operating System')
plt.ylabel('Number of users')

# Display the chart
```

```
plt.show()

## CSV WRITING
# Create a csv file for each dataframe
df.to_csv('df.csv', index=False)
agent_df.to_csv('agent_df.csv', index=False)
```

OS analysis figure created thanks to matplotlib:



CSV files screenshots:

- df.csv

client_ip	timestamp	request	response_code	response_size	user_agent
54.36.149.41	22/Jan/2019:03:56:14 +0330	GET /filter/27|13%20%D9%85%DA%AF%D8%A7%D9	200	30577	Mozilla/5.0 (compatible; AhrefsBot/6.1; +http://ahrefs.com/robot/)
31.56.96.51	22/Jan/2019:03:56:16 +0330	GET /image/60844/productModel/200x200 HTTP/1.1	200	5667	Mozilla/5.0 (Linux; Android 6.0; ALE-L21 Build/HuaweiALE-L21) Apple
31.56.96.51	22/Jan/2019:03:56:16 +0330	GET /image/61474/productModel/200x200 HTTP/1.1	200	5379	Mozilla/5.0 (Linux; Android 6.0; ALE-L21 Build/HuaweiALE-L21) Apple
40.77.167.129	22/Jan/2019:03:56:17 +0330	GET /image/14925/productModel/100x100 HTTP/1.1	200	1696	Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.h
91.99.72.15	22/Jan/2019:03:56:17 +0330	GET /product/31893/62100/%D8%B3%D8%B4%D9%	200	41483	Mozilla/5.0 (Windows NT 6.2; Win64; x64; rv:16.0)Gecko/16.0 Firefox/
40.77.167.129	22/Jan/2019:03:56:17 +0330	GET /image/23488/productModel/150x150 HTTP/1.1	200	2654	Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.h
40.77.167.129	22/Jan/2019:03:56:18 +0330	GET /image/45437/productModel/150x150 HTTP/1.1	200	3688	Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.h
40.77.167.129	22/Jan/2019:03:56:18 +0330	GET /image/576/article/100x100 HTTP/1.1	200	14776	Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.h
66.249.66.194	22/Jan/2019:03:56:18 +0330	GET /filter/b41,b665,c150%7C%D8%A8%D8%AE%D8	200	34277	Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot
40.77.167.129	22/Jan/2019:03:56:18 +0330	GET /image/57710/productModel/100x100 HTTP/1.1	200	1695	Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.h
207.46.13.136	22/Jan/2019:03:56:18 +0330	GET /product/10214 HTTP/1.1	200	39677	Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.h
40.77.167.129	22/Jan/2019:03:56:19 +0330	GET /image/578/article/100x100 HTTP/1.1	200	9831	Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.h
178.253.33.51	22/Jan/2019:03:56:19 +0330	GET /m/product/32574/62991/%D9%85%D8%A7%D8	200	20406	Mozilla/5.0 (Linux; Android 5.1; HTC Desire 728 dual sim) AppleWebK
40.77.167.129	22/Jan/2019:03:56:19 +0330	GET /image/6229/productModel/100x100 HTTP/1.1	200	1796	Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.h
91.99.72.15	22/Jan/2019:03:56:19 +0330	GET /product/10075/13903/%D9%85%D8%A7%DB%	200	41725	Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Ge
40.77.167.129	22/Jan/2019:03:56:19 +0330	GET /image/6229/productModel/150x150 HTTP/1.1	200	2739	Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.h
207.46.13.136	22/Jan/2019:03:56:19 +0330	GET /product/14926 HTTP/1.1	404	33617	Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.h
40.77.167.129	22/Jan/2019:03:56:19 +0330	GET /image/6248/productModel/150x150 HTTP/1.1	200	2788	Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.h
40.77.167.129	22/Jan/2019:03:56:20 +0330	GET /image/64815/productModel/150x150 HTTP/1.1	200	3481	Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.h

- agent_df.csv

raw_data	browser	os
Mozilla/5.0 (compatible; AhrefsBot/6.1; +http://ahrefs.com/robot/)	Mozilla	bot
Mozilla/5.0 (Linux; Android 6.0; ALE-L21 Build/HuaweiALE-L21) AppleWebKit/537.36 (KHTML, like	Mozilla	Linux
Mozilla/5.0 (Linux; Android 6.0; ALE-L21 Build/HuaweiALE-L21) AppleWebKit/537.36 (KHTML, like	Mozilla	Linux
Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)	Mozilla	bot
Mozilla/5.0 (Windows NT 6.2; Win64; x64; rv:16.0)Gecko/16.0 Firefox/16.0	Mozilla	Windows
Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)	Mozilla	bot
Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)	Mozilla	bot
Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)	Mozilla	bot
Mozilla/5.0 (compatible; Googlebot/2.1; +http://www.google.com/bot.html)	Mozilla	bot
Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)	Mozilla	bot
Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)	Mozilla	bot
Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)	Mozilla	bot
Mozilla/5.0 (Linux; Android 5.1; HTC Desire 728 dual sim) AppleWebKit/537.36 (KHTML, like Gecko	Mozilla	Linux
Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)	Mozilla	bot
Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/53.0.2785.92 Saf	Mozilla	Linux
Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)	Mozilla	bot
Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)	Mozilla	bot
Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)	Mozilla	bot
Mozilla/5.0 (compatible; bingbot/2.0; +http://www.bing.com/bingbot.htm)	Mozilla	bot