For Raspberry PI On-Premise IoT / Apache Kafka / TML Deployments:

Author: Sebastian Maurice

Below are suggested configurations – some fields may differ or may not apply

Server environment:zookeeper.version=3.6.1--104dcb3e3fb464b30c5186d229e00af9f332524b, built on 04/21/2020 15:01 GMT
Server environment:java.version=1.8.0_144

MAIN SITE: https://github.com/smaurice101/raspberrypi

1. **Install Java**

    a. **sudo apt-get update**
    b. **sudo apt install default-jdk**

2. **install TMUX: https://linuxhint.com/install-tmux-ubuntu/**

    a. **sudo apt update**
    b. **sudo apt-get install tmux**
    c. **pkill -f tmux (kills all tmux sessions)**
    d. **tmux a -t <session name>**

3. **Cpu monitoring:**

    a. **sudo apt update**
    b. **sudo apt install -y htop**

4. **Install MySQL:**

    a. **Sudo apt update**
    b. **Sudo apt upgrade**
    c. **sudo apt install mariadb-server**
    d. **sudo mysql_secure_installation**
    e. **enter: sudo mysql -u root**
    f. **SET PASSWORD FOR 'root'@'localhost' = PASSWORD('new_password');**
        i. **Replace 'new_password' with raspberry**
        ii. **GRANT ALL PRIVILEGES on *.* to 'root'@'localhost' IDENTIFIED BY 'raspberry';**
        iii. **FLUSH PRIVILEGES;**
        iv.

5. **Install Python (follow steps a-j)**

    a. **sudo apt-get install -y build-essential tk-dev libncurses5-dev libncursesw5-dev libreadline6-dev libdb5.3-dev libgdbm-dev libsqlite3-dev libssl-dev libbz2-dev libexpat1-dev liblzma-dev zlib1g-dev libffi-dev tar wget vim**

b. sudo mkdir Python
c. sudo apt-get install libatlas-base-dev
d. sudo chown pi:pi /home/pi/Python
e. cd Python
f. wget https://www.python.org/ftp/python/3.9.15/Python-3.9.15.tgz
g. sudo tar -zxvf Python-3.9.15.tgz
h. cd Python-3.9.15/
i. ./configure --enable-optimizations
j. sudo make altinstall
k. confirm install:
    i. python -version
    ii. IF WANT TO REMOVE PYTHON:
        1. sudo apt-get clean
        2. sudo apt-get autoremove --purge
        3. sudo apt-get remove python3.9
        4. sudo apt-get autoremove

6. IF REPOSITORY does not install:
    a. Use: sudo apt-get install software-properties-common
    b. Sudo apt-get update
7. install MAADSTML python library and other libraries
    a. pip install maadstml
    b. pip install requests
    c. pip install nest_asyncio
    d. pip install joblib
    e. pip install asyncio

8. Install TML Technologies for ARM chipset If using Pi 3+, for PI 4 using ARM64 binaries:YOU KEEP ALL THESE IN USB stick
    a. https://github.com/smaurice101/raspberrypi
        i. Download TML binaries
        ii. Put each unzipped files in their folders:
            1. Sudo mkdir viper
            2. Sudo mkdir hpde
            3. Sudo mkdir viperviz
    b. Copy viper-env-file from https://github.com/smaurice101/raspberrypi/tree/main/viper-env-file
        i. Replace viper.env file with one above
            1. Copy/replace to Viper and Viperviz folder
    c. IotSolution - copy all Python scripts from Github: https://drive.google.com/drive/folders/1wJEufLVtOzu8R-yYh6ybMASAaKkvMaTw?usp=sharing
        i. Sudo mkdir IotSolution
            1. produce-iot-customdata.py
            2. preprocess-iot-monitor-customdata.py
            3. preprocess2-iot-monitor-customdata.py

4. Raw IoT Device Data: IoTData.zip and unzip data
5. IoT Lat/Longs: downlaofdsntmlidmain.csv

9. Start Apache Kafka: ZOOKEEPER (Store on USB stick)
    a. Sudo mkdir Kafka
    b. Sudo chown pi:pi Kafka
    c. Cd kafka_2.13-3.0.0
    d. Download Kafka: wget https://archive.apache.org/dist/kafka/3.0.0/kafka_2.13-3.0.0.tgz
    e. tar xzf kafka_2.13-3.0.0.tgz
    f. export KAFKA_HEAP_OPTS="-Xmx512M -Xms512M"
    g. Edit zookeeper/conf/zoo.cfg
        i. tickTime=2000
           initLimit=10
           syncLimit=5
           dataDir=/media/pi/sebusb/zookeeper # change to your /mnt/usb
           clientPort=2181
    h. PATH="$PATH:/home/pi/Kafka/kafka_2.13-3.0.0/bin"
    i. cd bin
    j. Enter: zookeeper-server-start.sh /home/pi/Kafka/kafka_2.13-3.0.0/config/zookeeper.properties <enter>
        i. Make sure to change zookeeper.properties: dataDir=/mnt/usb/zoodata (or wherever the usb is mounted to)

10. Start Kafka in another terminal
    a. Kill port: sudo kill -9 $(sudo lsof -t -i:3000)
    b. export KAFKA_HEAP_OPTS="-Xmx512M -Xms512M"
    c. ./kafka-server-start.sh ../config/server.properties
        i. Make sure to change: log.dirs=/mnt/usb/kafka-logs

11. To SSH in to raspberry pi
    a. Use Raspberry Pi imager and flash OS
        i. https://www.raspberrypi.com/software/
    b. Click setting and enable SSH and Wifi
    c. To manually enable wifi/ssh follow blog here:
        i. https://roboticsbackend.com/enable-ssh-on-raspberry-pi-raspbian/

12. Mount USB
    a. Type: `sudo ls -l /dev/disk/by-uuid/`
    b. Capture uuid i.e. 61AA-D040 (for /sda1 which is usb)

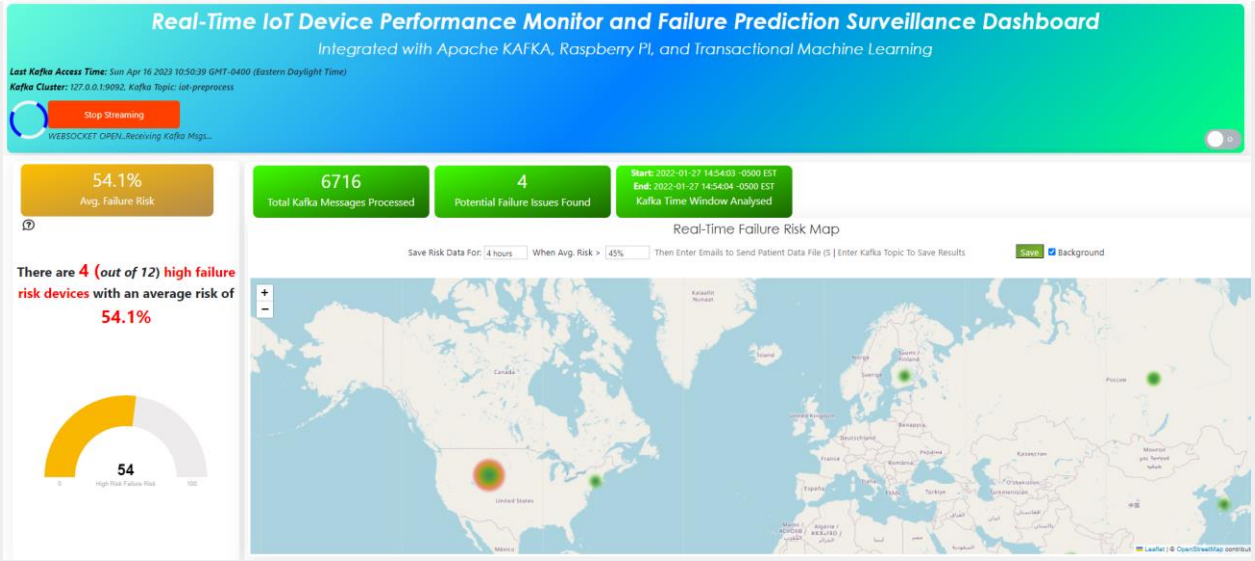
```
total 0
lrwxrwxrwx 1 root root 15 Feb 20 23:08 37CA-39EC -> ../../mmcblk0p1
lrwxrwxrwx 1 root root 10 Feb 20 23:08 61AA-D040 -> ../../sda1
lrwxrwxrwx 1 root root 15 Feb 20 23:08 a4af13c6-d165-4cbd-a9f6-c961fef8255d -> ../../mmcblk0p2
```

c.
d. Create mount folder: `sudo mkdir /mnt/usb`
e. Auto mount usb:
f. `Open file: sudo nano /etc/fstab`
g. Add line at the end of /etc/fstab:
    i. `UUID=61AA-D040 /mnt/usb vfat uid=pi,gid=pi 0 0`
    ii. `Save and exit`
    iii. `Reboot: sudo reboot`

To Execute the IoT Solution

13. **Cd to /mnt/usb/tmux**
14. **Run: ./tmux.sh**
    a. **Note you may need to edit**

15. **VISUALIZATION URL: Make sure to change IP address to the one given to your Raspberry PI:**
    a. http://192.168.2.29:9005/iot-failure-seneca.html?topic=iot-preprocess2,iot-preprocess&offset=-1&groupid=&rollbackoffset=500&topictype=prediction&append=0&secure=1
    b. You should see a dashboard similar to this:



16. **Increase swap file**
    a. `sudo nano /etc/dphys-swapfile`
    b. `CONF_SWAPSIZE=2000`
    c. `sudo reboot`
    d. `free -m`

ISSUES

If you run in to issues if the solution does not start do the following to re-start solution:

1. Enter: pkill -f tmux
2. Then restart and run:
    a. cd to tmux folder
    b. run: ./tmux.sh <enter>
    c. check solution is running by checking the tmux windows:
        i. list window – run: tmux ls
        ii. attach to window: tmux a -t preprocess2-data-python-8002
        iii. attach to window: tmux a -t preprocess2-data-viper-8002
3. if USB stick turns to READ-ONLY:

a. unplug USB and plug back in and reboot pi
    i. sudo reboot