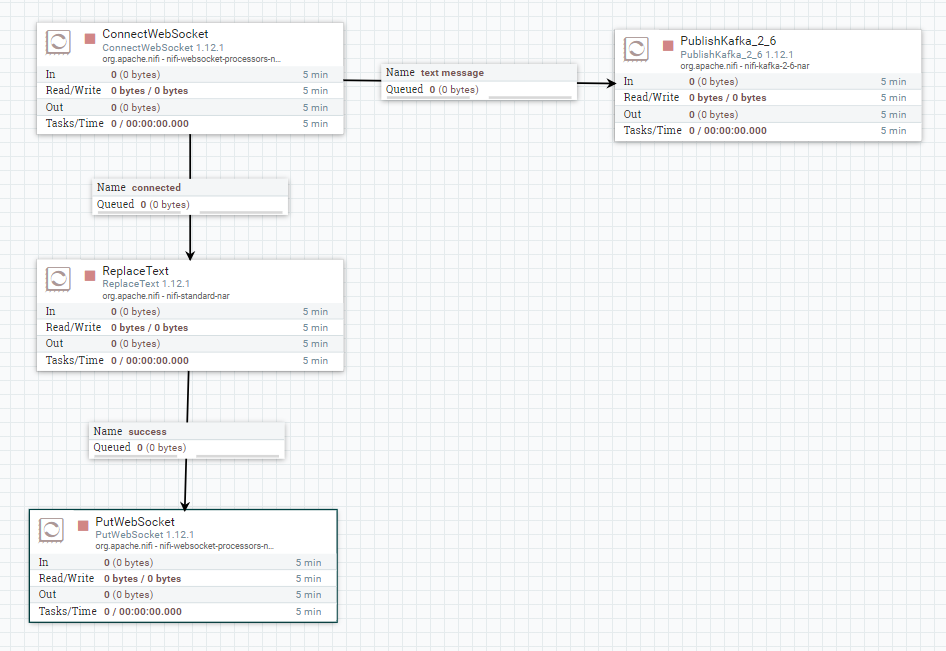
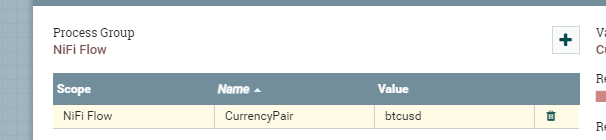
# NIFI flow

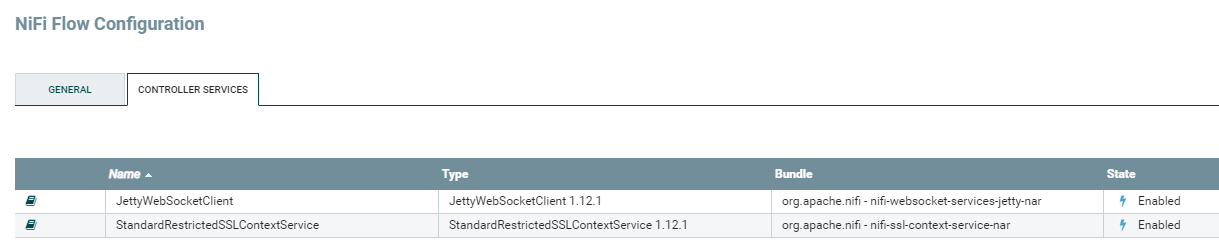
* 1. Import template “sandbox\_repo\training\GLBig\_Data\_ProCamp\home\_work\HW2\nifi\_kafka\_hw2\_template.xml”. Create flow with imported template.



* 1. Create NIFI variable “CurrencyPair” with value “btcusd”



* 1. Configure and enable controller services: “JetyWebSocketClient”, “StandardRegisteredSSLContextService”



# Scripts preparation on GCP data proc node.

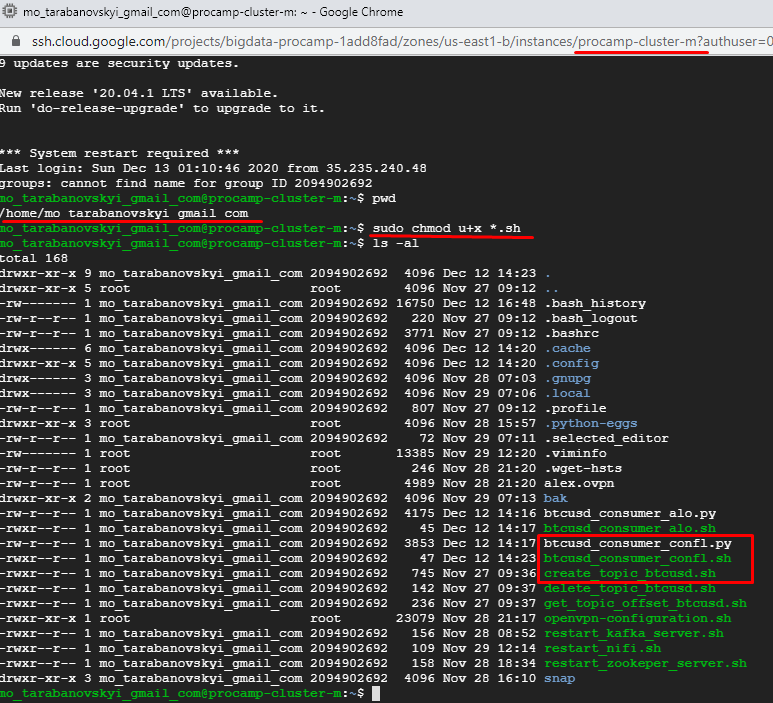
* 1. Copy shell and python scripts in home directory on GCP main node and grant permission:

Git Folder: “sandbox\_repo\training\GLBig\_Data\_ProCamp\home\_work\HW2”

Scripts:

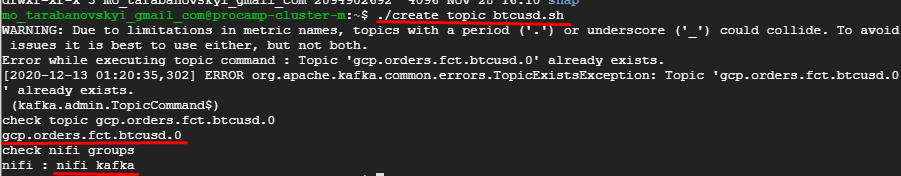
* “btcusd\_consumer\_confl.py”
* “btcusd\_consumer\_confl.sh”
* “create\_topic\_btcusd.sh”

Execute command on main node (to grant needed permissions): “sudo chmod u+x \*.sh”



# Create Kafka topic.

Launch shell script “create\_topic\_btcusd.sh” to create topic “gcp.orders.fct.btcusd.0” and add current and “nifi” users to group “kafka”:



# Install python modules “confluent-kafka”, “pandas”.

* 1. Install python module “kafka-confluence”:

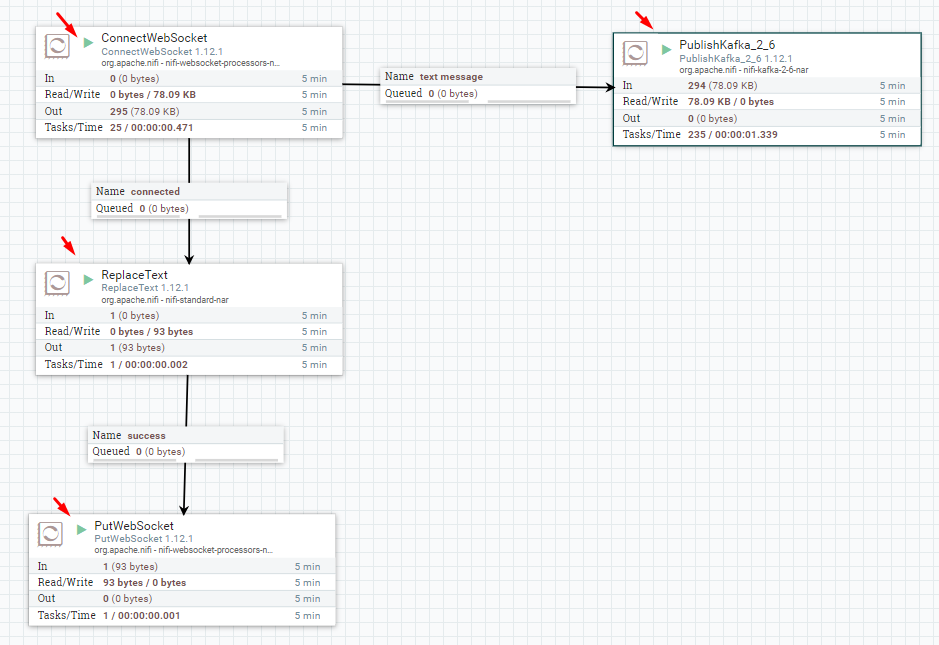
sudo /opt/conda/default/bin/python -m pip install confluent-kafka

* 1. Reinstall python module pandas (to install actual version)

sudo /opt/conda/default/bin/python -m pip uninstall pandas  
sudo /opt/conda/default/bin/python -m pip install pandas

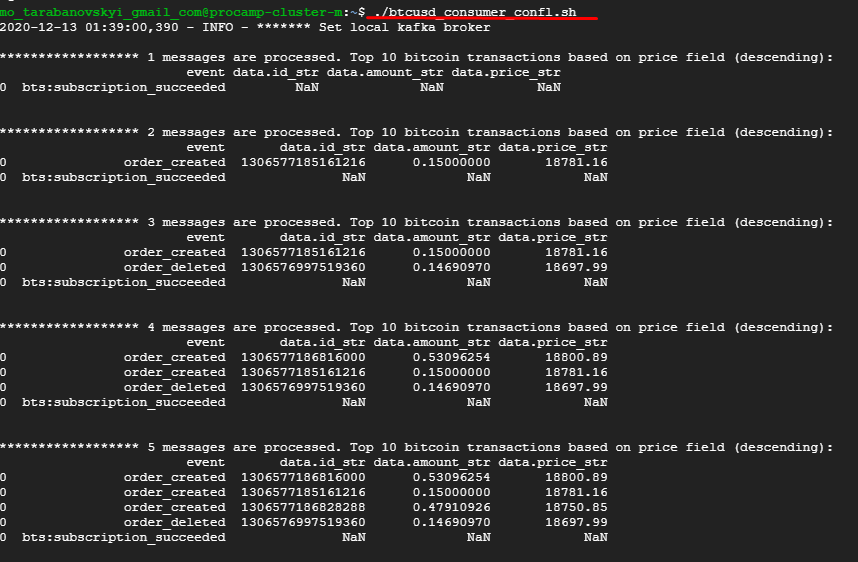
# Start kafka producer

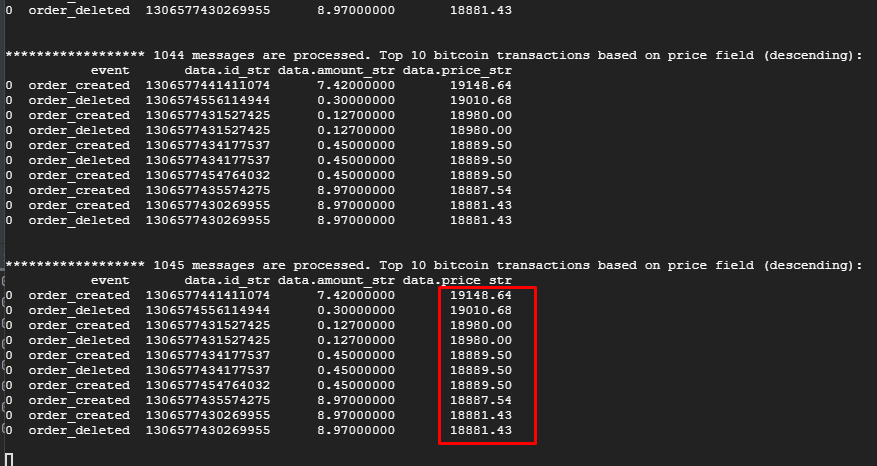
Start all Apache NIFI flow processors:



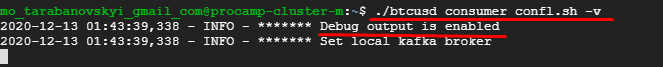
# Launch kafka consumer on GCP node.

Execute shell script “btcusd\_consumer\_confl.sh” on GCP master node.





Note: if something went wrang, launch script with option “-v” for verbose output:



# Launch kafka consumer on local PC. Windows OS example.

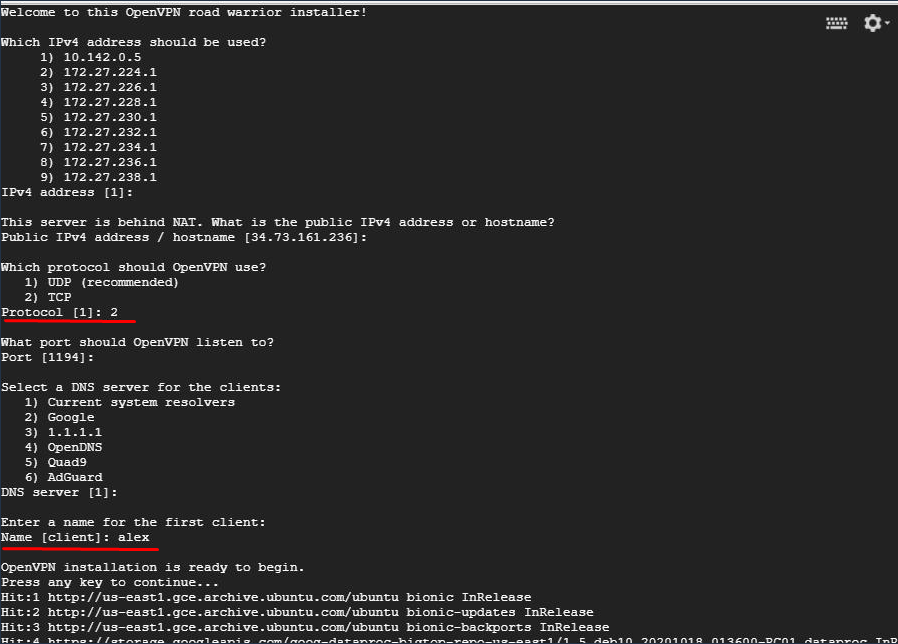
# **(optional step, can be useful for debug).**

* 1. Install python modules on local-PC:

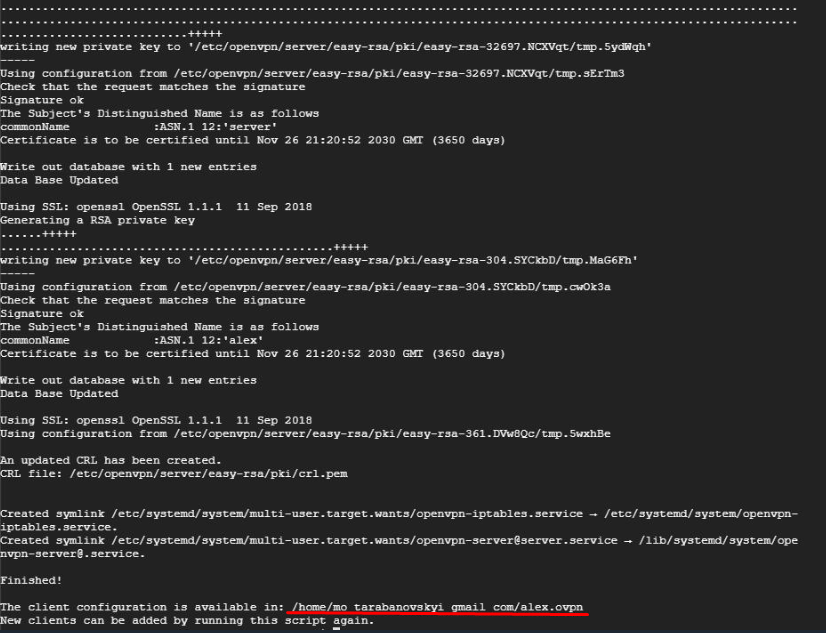
“pandas”, “kafka-consumer”.

*“kafka-consumer” can be installed on python 3.7 from “whl” file “sandbox\_repo\training\GLBig\_Data\_ProCamp\infra\confluent\confluent\_kafka-1.4.1-cp37-cp37m-win\_amd64.whl” (modules for version < 3.7 can be downloaded from confluent site).*

* 1. Up open VPN server on GCP master node
     1. Copy script on master node in home directory “sandbox\_repo\training\GLBig\_Data\_ProCamp\infra\vpn\openvpn-configuration.sh”
     2. Launch script “sudo openvpn-configuration.sh” and fill parameters by default values (except: protocol - choose “TCP” and set client name)



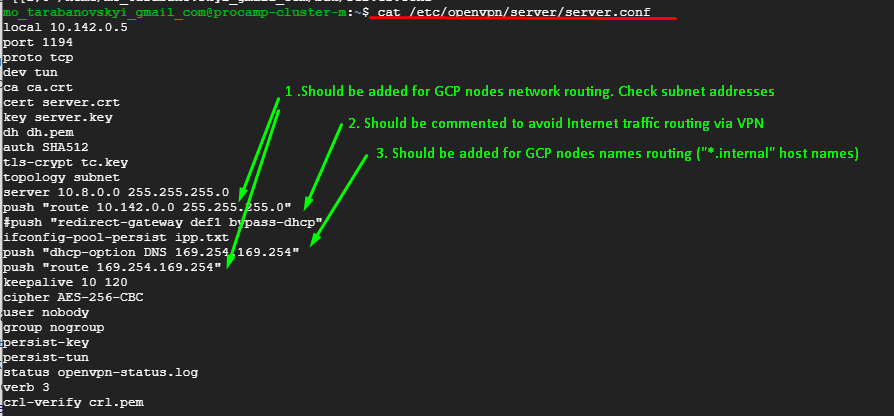
* + 1. Copy client configuration on local PC



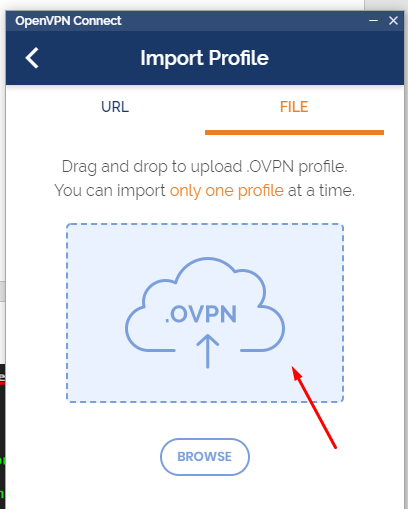
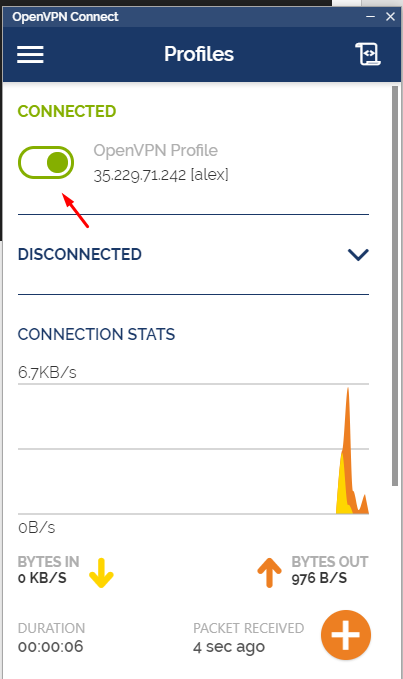
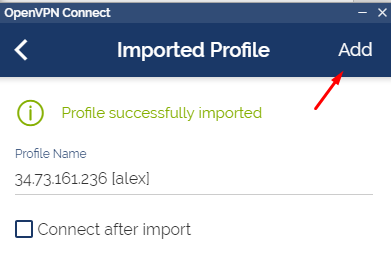
* + 1. Change vpn server configuration and reboot master node

Server configuration example: “sandbox\_repo\training\GLBig\_Data\_ProCamp\infra\vpn\server.conf”

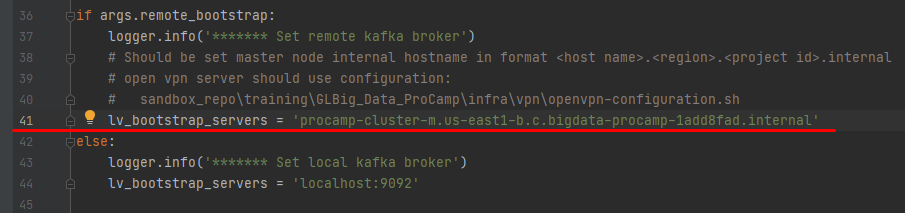
The configuration location is “/etc/openvpn/server/server.conf” on GCP master node.



* 1. Install Open VPN client and import client configuration file (described in 7.2.3) and enable connection

* 1. Change GCP master node name in python script “sandbox\_repo\training\GLBig\_Data\_ProCamp\home\_work\HW2\btcusd\_consumer\_confl.py”:



* 1. Launch consumer with option “-r” (remote broker) – “sandbox\_repo\training\GLBig\_Data\_ProCamp\home\_work\HW2\btcusd\_consumer\_confl.py”

