1-Installation

Install OpenPLC Runtime on both Master and Slave. Upload 'elegant_v11.st' program from repository 'https://github.com/bmsousa/elegant' on slave, then set up slave device by following configuration. To capture traffic, install Wireshark Network Analyzer (Version 4.0.2 - v4.0.2-0-g415456d13370).

• Configuration of slave device in master device as shown in figure 1.

• Connect master and slave with cat5 cable with length of 2 m.

First, run OpenPLC Runtime in slave then run it in master, second run OpenPLC Runtime in master, at the end start Wireshark capturing and let it to capture for about 2 min. Make screen shot considering characteristics of experiments as shown in figure 2. Save pcab file with appropriate name (for example including the date and the number of experiment). In addition, export report to csv. Do the same things for more 4 times.

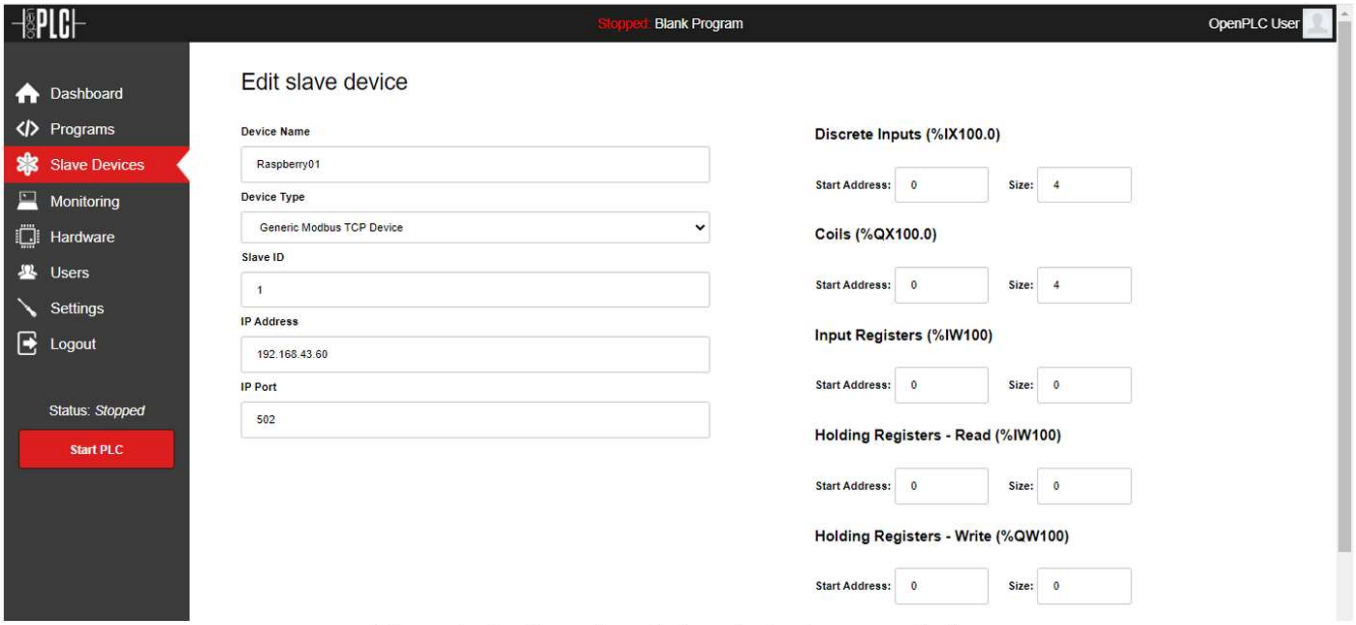

Figure 1: Configuration of slave device in master device

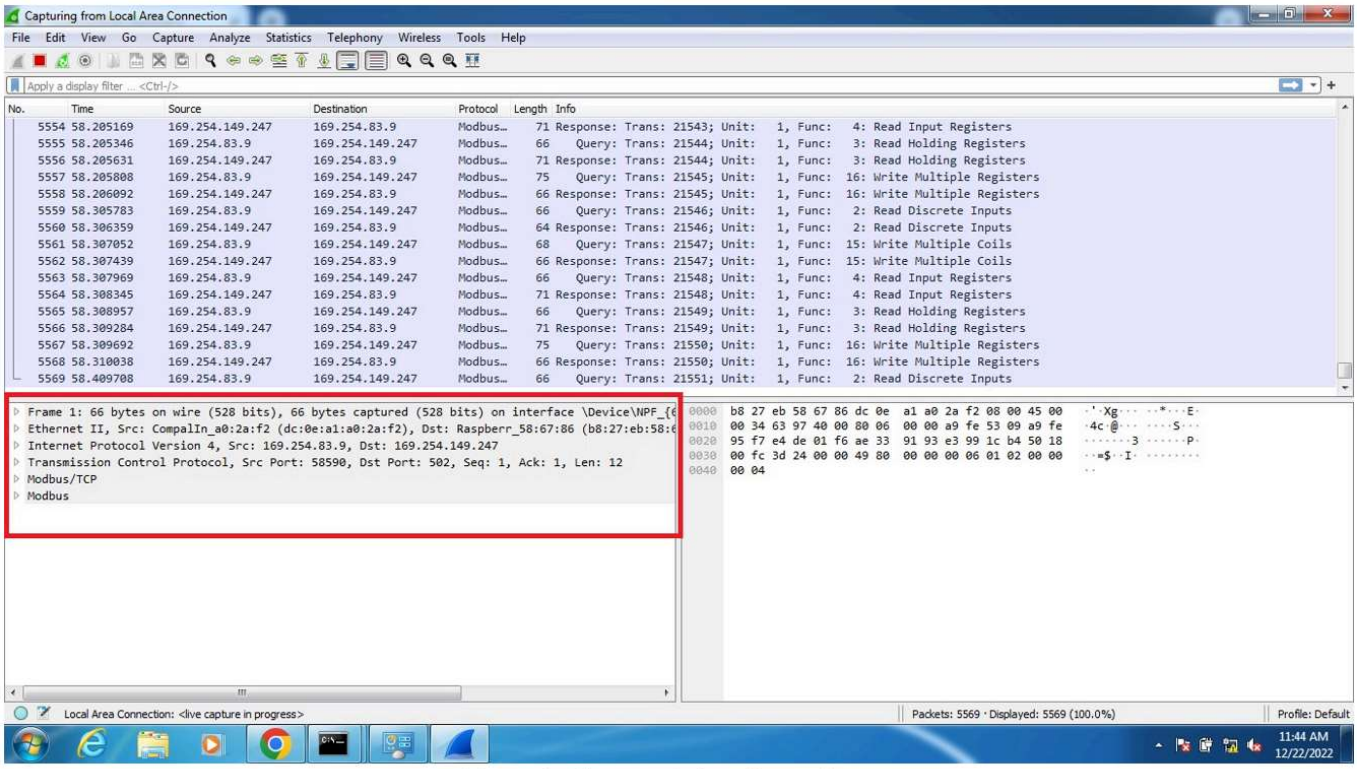

Figure 2: Characteristics of experiments

2-Devices

2-1-Master:

ACER ASPIRE V3-571 (Intel Core i52450M 2.5GHz turbo boost up to 3.1GHz, 8GB DDR3)

2-2-Slave:

Raspberry Pi 3 Model B+ (1.4GHz 1.4GHz)

2-3-Switch:

3Com 3C17304A 4200 28-Port 10/100 Fast Ethernet Switch

3-Preprocessing of dataset

To analyze data, methods and functions of Pandas package is considered as the main tool to deal with raw data. In Table 1 all applied packages are mentioned. In table 1 top 5 rows from experiment number one, is shown to get familiar with the features and cell's data format.

Table 1: Python packages applied to dealing with data

Table 1: Python packages applied to dealing with data.

Package	Description
Pandas	Dealing with data frame
Numpy	Dealing with arrays
Seaborn	Drawing statistical diagrams
Matplotlib	Drawing diagrams in general
OS	Dealing with OS commands
re	Dealing with RegEx
glob	Dealing with files

Table 2: Top 5 rows from experiment number one

No.	Time	Source	Destination	Protocol	Length	Info
1	0	169.254.83.9	169.254.149.247	Modbus/TCP	66	Query: Trans: 7011; Unit: 1, Func: 2: Read Discrete Inputs
2	0.000825	169.254.149.247	169.254.83.9	Modbus/TCP	64	Response: Trans: 7011; Unit: 1, Func: 2: Read Discrete Inputs
3	0.001395	169.254.83.9	169.254.149.247	Modbus/TCP	68	Query: Trans: 7012; Unit: 1, Func: 15: Write Multiple Coils
4	0.002072	169.254.149.247	169.254.83.9	Modbus/TCP	66	Response: Trans: 7012; Unit: 1, Func: 15: Write Multiple Coils
5	0.002511	169.254.83.9	169.254.149.247	Modbus/TCP	66	Query: Trans: 7013; Unit: 1, Func: 4: Read Input Registers

4-Queries and Preprocessing

Before everything, to apply all captured files in the same codes, it is necessary to rename all 'csv' files in precise format. So, a python code 'rename_pcab_files.py' is developed (e.g., 'Pcab 01 221222.csv')

4-1-Main queries from raw data.

As it is shown in table 2, some basic queries regarding captured packages including source Ips, Protocols, Length, etc. could be driven. For doing mentioned basic queries 'pcab_analysis_v02.py' is developed which generate two kinds of files including '*.csv' and* '.txt'. The report is in 'txt' format which is shown in figure 1 (e.g., 'Information Summary 01 221222.txt')

4-2-Preprocessing of raw data

As it is shown in table 1, the last column ('Info') includes different information such as 'Query/Response', 'Trans Number', 'Function Number' and 'Function Description'. To deal with this column, in preprocessing program it is divided into six columns extracted via program 'pcab_analysis_v02.py':

• 'Info_01': it shows that the packet belongs to Query or Response.

• 'Commands': it shows command, in another word functions related to each packet.

• 'Trans': it shows transmission number for each packet.

• 'R_W': it is extracted form 'Commands' which is divided by three and includes: 'Write' or 'Read' as the first part of commands' phrase.

• 'D_M_I_H': it is extracted as the same as 'R_W' and includes: 'Discrete', 'Multiple', 'Input' or 'Holding' as the second part of commands' phrase.

• 'I_C_R': it is extracted as the same as 'R_W' and includes: 'Inputs', 'Coils' or 'Registers' as the third part of commands' phrase.

4-3-Calculating new items

Regarding doing analysis, it is necessary to calculate some and assign new columns which inspire meaning to some raw data. For instance, regarding understanding round trip of packets, times related to precise transmission number; however, the first step is calculation of each packet time length. Some calculation formulas are mentioned in the following equation:

$$\beta_{i+1} = \tau_{i+1} - \tau_i \qquad \text{Equation 1}$$

In which β shows basic time difference, τ shows time related to each one-way transmission and i shows the raw number. Box plot and histogram is drawn by programs 'pacb_histogram.py' and 'pcab_boxplot.py'.

$$\delta_i = \beta_j - \beta_k \qquad \text{Equation 2}$$

In which δ shows time duration of round-trip packet for transmission number i (data in 'Trans' column), j is raw belongs to 'Response' joined with i and k is raw belongs to 'Query' joined with i. Note that, there is not any column including δ_i , instead it is issued by making queries from dataset. Box plot and histogram is drawn as previous. As the same as previous, it is possible to query times interval between each query by sequence of transmission number and each response and box plot and histogram is drawn. It is assumed that difference of times between query and responses for a specific transmission is equal to the time of delay of packet transmission Box plot and histogram is drawn as previous.

$$\theta_i = \delta_j - \vartheta_j \qquad \text{Equation 3}$$

In which θ shows packet consumption' for transmission number i (data in 'Trans' column), δ_j is raw belongs to 'Query' joined with i and ϑ_j is raw belongs to 'Response' joined with i.

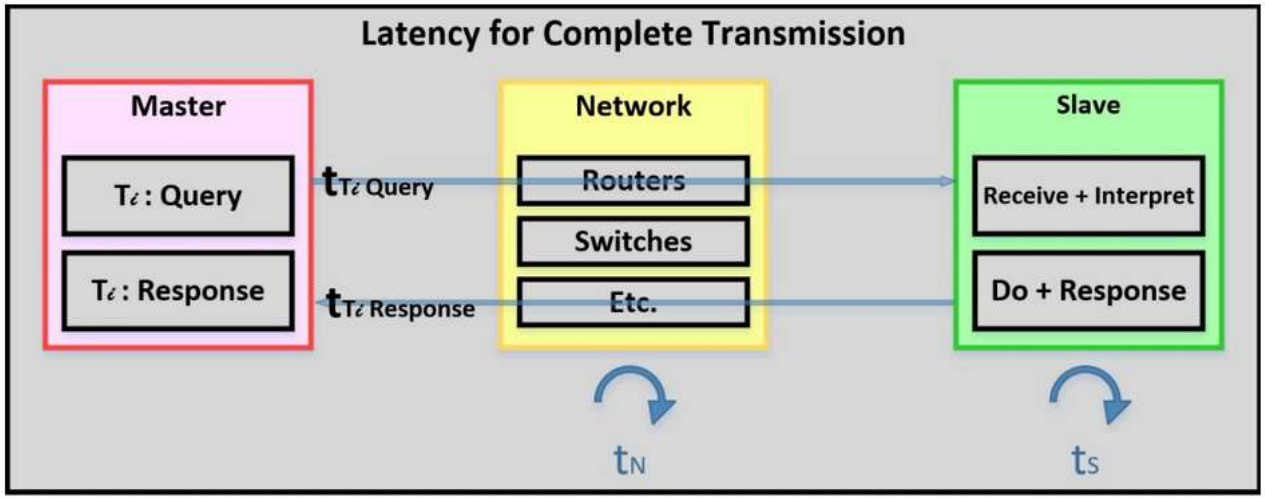

Figure 3: Explanation of Complete Transmission Latency

Figure 3: First parst Explain Complete Transmission Latency following the idea of second part (Reference of below part of figure: Antonio Virdis, Giovanni Nardini, Giovanni Stea and Dario Sabella "End-to-End Performance Evaluation of MEC Deployments in 5G Scenarios", doi:10.3390/jsan9040057)

It is assumed that difference of times between two responses is equal to the time of processing in master plus time consumption to send message from master to slave plus processing time in slave and at the end plus time consumption to response to the master which is considered as 'packet consumption' time in this experiment. Box plot and histogram is drawn as previous.

$$\theta_i = \delta_j - \delta_k \qquad \text{Equation 4}$$

In which θ shows packet consumption' for transmission number i (data in 'Trans' column), j is raw belongs to 'Response' joined with i and k is raw belongs to 'Response joined with i+1.

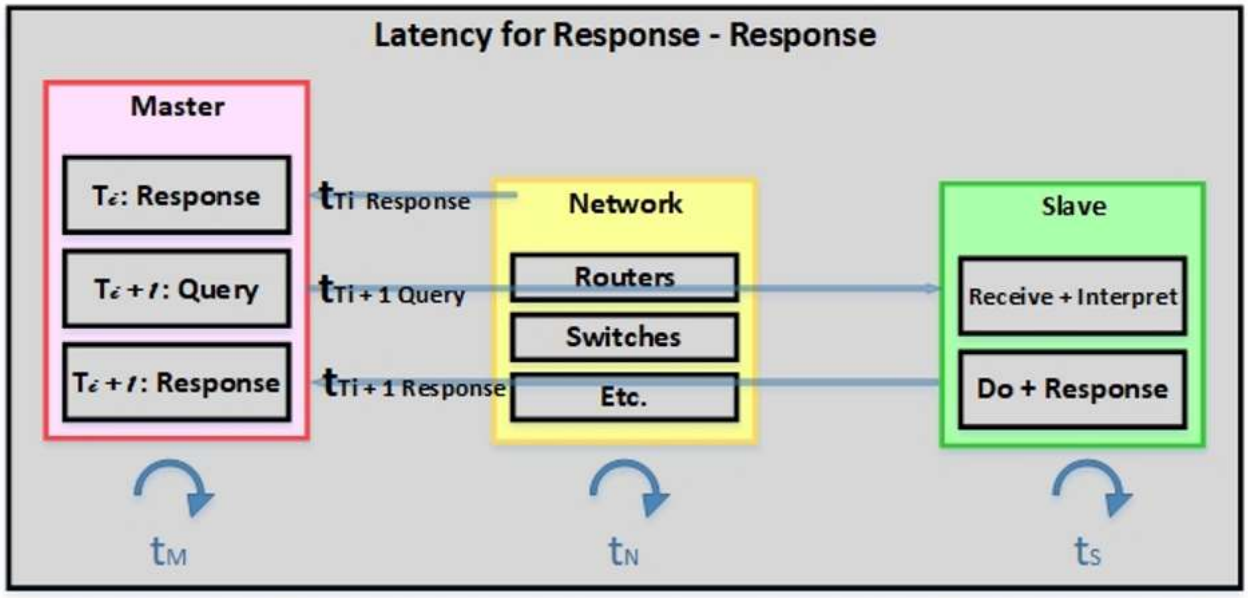

Figure 4: Explanation of Response – Response Latency

5-Remarks

Durations of these Runnings are considered 120 s, in other words capturing could be for a longer period and only 120 s of that extracted to study.

Some transmissions are not completed during the considered time slot which are detected and shown in *.txt report files. In addition, must be omitted for some statistical analysis.

In some selected time slot, data set is started by response which is related to a query that is not in extracted time slot, so this raw must be omitted from dataset.

From this point to end of results, the python functions which generate the results have been described.

```
Information Summary 01 221222
Number of Samples: 11435

..........................................................
Source - Destination              : Total Number
..........................................................
169.254.149.247                   : 5613
169.254.83.9                      : 5617
239.255.255.250                   : 4
255.255.255.255                   : 2
Broadcast                         : 117
CompalIn_a0:2a:f2                 : 5
Raspberr_58:67:86                 : 5
Spanning-tree-(for-bridges)_00    : 60
ff02::1:2                         : 7
ff02::fb                          : 5

..........................................................
Protocols      : Total Number   : Mean of Lenght
..........................................................
ARP            : 127            : 59.29
DHCP           : 2              : 342.0
DHCPv6         : 7              : 145.0
MDNS           : 5              : 103.4
Modbus/TCP     : 11227          : 67.9
SSDP           : 4              : 217.0
STP            : 60             : 64.0
TCP            : 3              : 62.0

..........................................................
Commands                      : Total Number
..........................................................
Read Discrete Inputs          : 2246
Read Holding Registers        : 2244
Read Input Registers          : 2245
Write Multiple Coils          : 2246
Write Multiple Registers      : 2246
is at b                       : 5

..........................................................
               not rounded Transmissions

Type of Transmission: Counts    : Trans Number
..........................................................
Query              : 1          : 2623,
Response           : 2          : 5610, 3535,
```

Figure 5: Report of basic driven information

** **Important Notice** **

1-

- Moving forward, all images required for the report will be generated using code. For instance, the image labeled as Figure 5 above is created using the codes provided below.

2-

- There are various parameters associated with image generation that you can adjust to enhance your understanding of their significance in the graphs.

3-

- For each each fiure based on their number, some fuctiond are developed in the following whit the name of figures respectedly. For instanse "figure_06 refered to fuctions genearting figure 6 of this report.

==========

** **Imprtant Notice to Run Perfectly** **

If you want 'Run All', please consdier put:

For 1st Run

- permission_df_generating = True
- permission_graphs_drawing = False