

# Relazione per l'elaborato di Programmazione di Reti

Mrclean69

22 luglio 2021

# Indice

<b>1</b>	<b>Introduzione</b>	<b>2</b>
<b>2</b>	<b>Descrizione</b>	<b>3</b>
	Device (Smart IoT Meter) . . . . .	3
	Gateway . . . . .	3

# Introduzione

La traccia scelta è la numero 1, ovvero realizzare una simulazione di uno scenario IoT dove ci sono diversi Smart IoT Meter che rilevano la temperatura e umidità del terreno in cui sono installati. Questi si connetteranno una volta al giorno per inviare ad un gateway tramite una connessione UDP tutte le misure fatte durante il giorno. Il gateway, a sua volta, invia tutte le misure raccolte ad un server, il quale mostra su console tutte le misure dei vari device.

## Descrizione

### Device (Smart IoT Meter)

I device sono stati realizzati su 4 moduli separati (*device1.py*, *device2.py*, ...) dentro ai quali sono definiti, mediante costanti, i seguenti dati:

- Il numero di misure random (*N\_MEASURES*) che si vogliono generare per poi inviare al gateway
- L'indirizzo IP del device (*IP\_ADDRESS\_DEVICE*)
- La subnet mask del device (*SUBNET\_MASK\_DEVICE*)
- L'indirizzo del gateway (*GATEWAY\_ADDRESS*)
- La porta del gateway (*GATEWAY\_PORT*)

Successivamente viene creato un oggetto **IP\_Address** (definito nel modulo *IP\_Address*) che descrive la configurazione IP del dispositivo (Indirizzo IP e subnet mask).

In seguito viene creato l'oggetto **device** al quale viene passato l'*ID* del dispositivo e l'oggetto *IP\_Address* definito prima. Vengono poi chiamate le funzioni **generate\_random\_measures**, la quale genera delle misure randomiche, la funzione **print\_info**, la quale stampa le informazioni del dispositivo creato, e la funzione **send\_data**, la quale invia i dati letti al server. Nella funzione **send\_data** si creerà un messaggio (codificato) composto nel seguente modo:

**ip + subnet mask + tempo di inizio invio del pacchetto + misure da inviare**

L' *IP* e la *subnet mask* sono codificati in un byte array grazie alla funzione *encode\_ip\_and\_subnet* dichiarata nel modulo *IP\_Address*, il tempo di inizio invio del pacchetto è acquisito grazie alla funzione *perf\_counter* del modulo *time* per poi essere codificato, sempre in byte array con la funzione *pack* contenuta nel modulo *struct*. Le misure da inviare, invece, vengono lette dal corrispondente **file csv** per poi essere codificate.

### Gateway

L'implementazione del gateway è all'interno del modulo **gateway.py**. Qua vengono istanziati due oggetti **IP\_Address** che descrivono le due "interfacce" di rete del gateway, una rivolta verso la rete dei device e una verso la rete del *server*. Verrà creato un socket UDP, con porta specificata nella costante *GATEWAY\_DEVICE\_SIDE\_PORT*, al quale poi si collegheranno i vari **device**. Quando verrà ricevuto un pacchetto da un dispositivo, si procederà a "spacchettarlo" nel seguente modo:

- I primi 4 byte contengono l'indirizzo IP del dispositivo
- I successivi 4 byte contengono la subnet mask del dispositivo
- Gli 8 byte successivi contengono il tempo di inizio di invio del pacchetto da parte del dispositivo
- I restanti byte contengono le misure inviate

Con i primi 8 byte, quindi, si crea l'oggetto **IP\_Address** del dispositivo che ha inviato i dati grazie alla funzione **bytes\_to\_IP** (contenuta sempre nel modulo *IP\_Address*). Gli 8 byte del tempo, invece, vengono convertiti in **double** grazie alla funzione **unpack** contenuta nel modulo *struct*. Una volta estratti i dati, il gateway verifica se il dispositivo che ha inviato i dati è all'interno della stessa sottorete e, per fare questo, si avvale della funzione **is\_in\_same\_network** presente nel modulo *IP\_Address*, passandogli l'oggetto *IP\_Address* creato in precedenza. All'interno di questa funzione viene eseguito un **AND logico** tra l'indirizzo IP e la subnet mask del gateway, in modo da ricavare l'indirizzo di rete, e viene confrontato, a sua volta, con l'indirizzo di rete del dispositivo che ha inviato i dati (ottenuto in modo analogo). Se i due indirizzi di rete sono uguali, il gateway accetterà le misure inviate dal dispositivo, altrimenti le scarterà e mostrerà un messaggio di errore.

Il gateway ripeterà questa operazione finchè il numero di dispositivi univoci che hanno inviato i dati è uguale alla costante **NUMBER\_OF\_DIFFERENT\_CLIENTS**. Se un dispositivo che ha già inviato le proprie misure cercherà di rinviarle, il gateway scarterà il messaggio.