

Università di Pisa  
Corso di Gestione di Reti A.A.2020/21  
Docente: Deri Luca  
**Monitoraggio risorse su PC**

Candidato: Firera Salvo - 578018

## 1 Panoramica del sistema

Il programma, implementato in Python 3.7, permette il monitoraggio di alcune risorse hardware della macchina rilevando i seguenti parametri:

- Memoria RAM inutilizzata
- Percentuale disco occupata
- Carico della CPU

Per l'implementazione del sistema è stato utilizzato il protocollo SNMP e per la gestione dei dati InfluxDB 2, che integra un'interfaccia grafica di monitoraggio in tempo reale oltre alla possibilità di configurare degli alert personalizzati in base a dei criteri specifici (nella versione 1 il servizio UI era offerto da un tool a parte).

## 2 Requisiti e dipendenze

- installazione di InfluxDB 2 e InfluxDB CLient 2
- che siano attivi i servizi InfluxDB, e Influxd
- librerie Python Easysnmp, Influxdb-client
- installazione servizio SNMP

## 3 Esecuzione

- avviare i servizi Influx e Influxd
- eseguire il comando *python3 PCmonitor.py*

- su Telegram avviare il bot @InfluxdbNotyfybot
- aprire il browser sulla pagina *http://localhost:8086*
- nella barra laterale andare sulla dashboard per il monitoraggio in tempo reale
- per terminare l'esecuzione digitare *ctrl+c*

N.B. Gli alert su Telegram arrivano solamente a un dispositivo per volta che va configurato nello script del task di notifica

## 4 Rilevamento dei dati

Il rilevamento dei parametri delle risorse viene fatto utilizzando il protocollo SNMP visto durante il corso, consiste nell'interrogare ad intervalli regolari degli agenti (MIBS) tramite la funzione *snmpget*. I MIBS in questione sono i seguenti:

- iso.3.6.1.4.1.2021.4.11.0: RAM inutilizzata.
- iso.3.6.1.4.1.2021.9.1.9.1: percentuale disco occupata
- iso.3.6.1.2.1.25.3.3.1.2.196608: percentuale della CPU

Questi OIDs sono stati presi dal sito *oidref.com*. Il protocollo SNMP può essere utilizzato all'interno di applicativi Python tramite la libreria EasySNMP.

## 5 Salvataggio dei dati

I valori rilevati tramite SNMP vengono immagazzinati in un database no-SQL. In questo caso è stato utilizzato InfluxDB, un database a serie temporali che oltre allo storage di dati offre un accesso intuitivo tramite UI dal browser web. A differenza delle precedenti versioni, InfluxDB 2 integra un'interfaccia web sulla porta 8086 del localhost, da qui è possibile creare un database (bucket) con associato un token che permette di interagire mediante la libreria Python dedicata. Collegandosi alla UI è possibile monitorare in tempo reale i dati rilevati dal programma mediante una serie di tool grafici personalizzabili a piacimento. Al termine dell'esecuzione viene fatto un "flush" del bucket.

## 6 Alerts personalizzabili

Per mandare gli alert è stata fatta un collegamento con un bot Telegram, automatizzando l'invio di messaggi quando i parametri superano (nel caso della ram sono al di sotto) di un certo valore. A differenza di Chronograf, qui per utilizzare Telegram è necessario creare degli script in linguaggio Influx per creare dei task di notifica. Il codice dei task viene fornito assieme al programma per eventuale consultazione. Come anticipato sopra, gli alert su telegram possono essere inviati soltanto a un dispositivo, identificato tramite il ChatID nel codice del task. Gli alert su telegram sono attivi anche se si chiude il browser.

## 7 Link utili

EasySNMP: <https://pypi.org/project/easysnmp/>

InfluxDB-client: <https://docs.influxdata.com/influxdb/cloud/tools/client-libraries/python/>

InfluxDB Tools: <https://portal.influxdata.com/downloads/>

OID Ref: <https://oidref.com/>

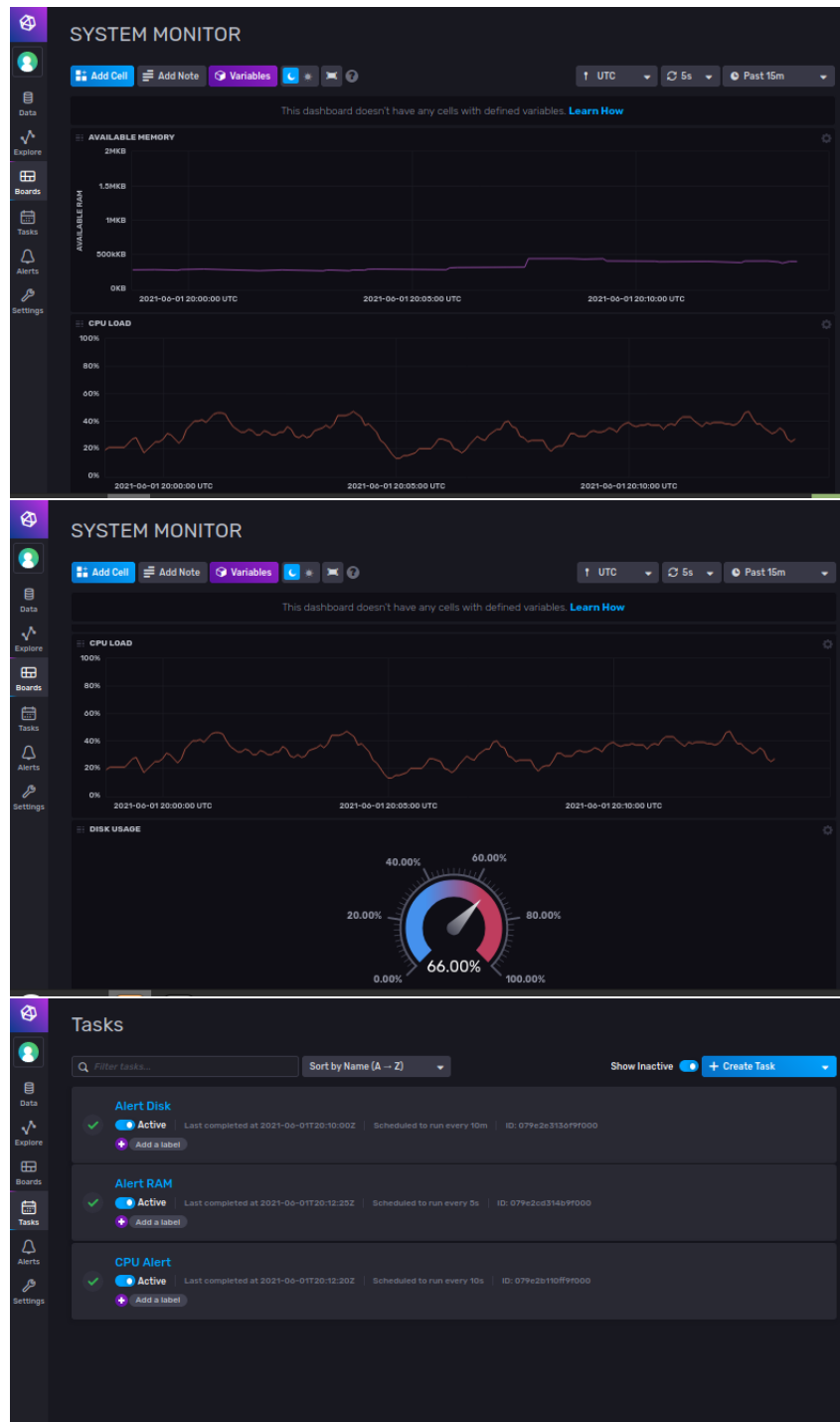


Figure 1: Interfaccia Web di InfluxDB