

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). Il sistema è stato testato su Lubuntu installato su macchina virtuale.

2 Requisiti e dipendenze

- installazione di InfluxDB 2
- librerie Python Easysnmp, Influxdb-client
- installazione servizio SNMP

3 Installazione SNMP

Aprire il terminale ed eseguire i seguenti comandi:

- *sudo apt-get install snmp snmpd snmp-mibs-downloader*

- aprire il file *etc/anmp/snmpd.conf* e nella sezione "ACCESS CONTROL" abilitare i percorsi per interrogare gli OIDs
- aprire il file *etc/snmp/snmp.conf* e accanto accanto alla parola "mibs" aggiungere "+ALL"
- ricaricare il servizio con *sudo service snmpd restart*
- controllare se il servizio è attivo con *sudo service snmpd status* se attivo fare una walk di prova

4 Installazione API Python

Aprire il terminale ed eseguire i seguenti comandi:

- *sudo apt-get install libsnmp-dev*
- *pip3 install easysnmp*
- *pip3 install influxdb-client*

5 Creazione bot Telegram per alert

- aprire Telegram e iniziare una conversazione con il profilo *BotFather*
- seguire la procedura guidata scegliendo il nome del bot e altri parametri
- salvare e custodire con cura il token che viene rilasciato, questo servirà per controllare il bot dalle api
- aprire da browser il link *https://api.telegram.org/botNOSTRO-TOKEN/getUpdates*
- aprire una chat col nuovo bot e scrivere un qualsiasi testo.
- ricaricare la pagina web prelevando dal json visualizzato il ChatID che ci servirà in seguito.

6 Configurazione UI per monitoraggio

- scaricare Influxdb da *https://docs.influxdata.com/influxdb/v2.0/install/?t=Linux*
- scompattare e copiare i file influx e influxd in *usr/local/bin*
- eseguire il comando *sudo influxd*, NON CHIUDERE questa finestra del terminale, continuare su un'altra tab
- aprire il browser su *http://localhost:8086* e cliccare su "Get Started"
- creare un profilo con nome utente, password, organizzazione e bucket

- Nella barra laterale sinistra cliccare su "Data", andare nella sezione "Tokens", selezionare quello presente (admin) e copiare il token.
- aprire il file *conf.py* e completarlo con i dati appena inseriti nella UI

7 Configurazione alert

Nella web UI andare nella sezione "Task", e creare 3 nuovi task (RAM, CPU, Disco), si aprirà un editor in cui bisogna incollare gli script forniti nella consegna nel file *task-script* avendo cura di sostituire i campi *YourToken* e *YourChatID* con i dati ottenuti alla sezione 5. Dare un nome significativo ai task e impostare la frequenza desiderata.

N.B. Gli alert restano in funzione anche chiudendo il browser finchè lo script rimarrà in esecuzione.

8 Utilizzo del sistema

Dopo aver seguito le procedure di cui sopra, possiamo eseguire il programma:

- da terminale lanciamo *python3 PCMonitor.py*
- si aprirà in automatico la pagina web *http://localhost:8086*
- nella barra a sinistra andiamo su "Boards" e da qui possiamo seguire la procedura grafica per creare qualsiasi grafico prelevando le informazioni che lo script manda al bucket attraverso la API
- nel frattempo i task si occuperanno di controllare che i record inseriti siano al di sotto (o al di sopra) delle soglie selezionate, eventualmente mandando degli alert sul bot Telegram.
- Ovviamente dobbiamo avviare il bot sul nostro telefono avviando una chat e digitando */start*
- per terminare il programma, sul terminale digitiamo *ctrl+c*

Alla chiusura il bucket viene svuotato dei dati accumulati.

9 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.

10 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.

11 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.

12 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/>

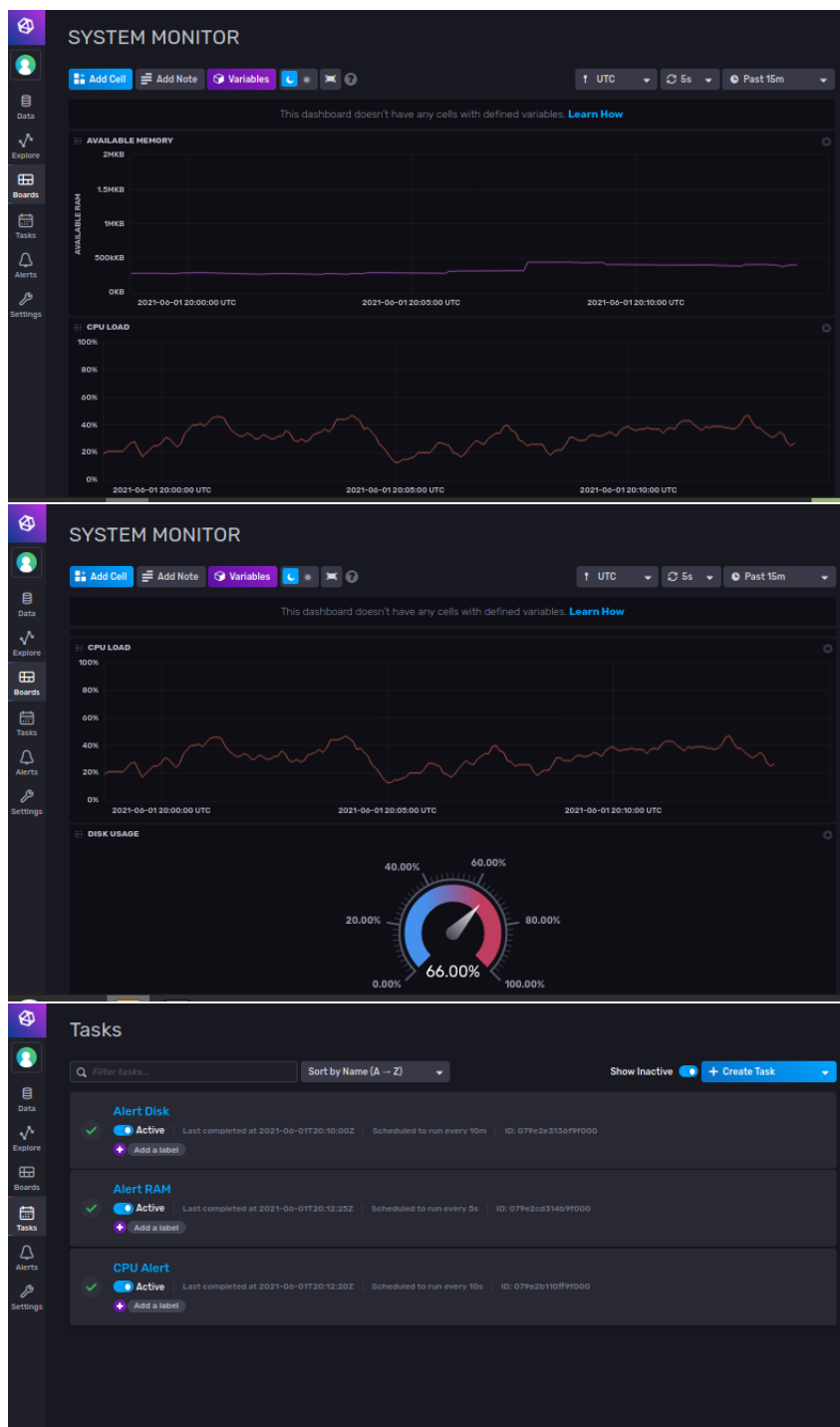


Figure 1: Interfaccia Web di InfluxDB