



UNIVERSITÀ DI PISA

Anno Accademico 2018/2019

Relazione Progetto

Gestione Di Reti

<https://github.com/PieroBook/ContainerTrafficCounter>

Docente Corso

Luca Deri

Studente

Pietro Libro

1.1 MicroK8s

MicroK8s è una micro distribuzione di kubernetes disponibile sotto forma di pacchetto snap, installabile su 42 tipi diversi di distribuzioni linux. Vantaggi principali la compattezza e leggerezza, che lo rendono ideale per testing su una macchina non molto potente come ad esempio un laptop. Sono inoltre presenti una serie di add-ons che implementano servizi ulteriori (e.g. Dashborads).

1.2 Influx DB

InfluxDB è un time-series database, molto performante nella gestione di grandi quantità di dati che devono essere recuperate, usando tags che vengono indicizzati (in modo da rapidizzare al massimo l'efficienza) e field che contengono le metriche che si vuole poi elaborare. In più permette di gestire i dati con ordine temporale, che nel caso di misure/rilevazioni è fondamentale. Il DB è interrogabile attraverso API http e rest che rendono quindi molto rapido l'utilizzo.

1.3 Grafana

Grafana è uno strumento che consente di interrogare una base di dati, nel caso specifico un database influx, gestire le time-series e graficare le informazioni ottenute interrogando il db. Presenta una graphic user interface molto semplice, e permette di creare in poco tempo grafici dalle informazioni estratte.

1.4 Obiettivo

Obiettivo di questo progettino è quello di riuscire a risalire da uno specifico container alla propria interfaccia Virtual Ethernet, associatagli dal sistema, e rilevare le metriche di trasmissione/ricezione dati ovvero: numero di byte inviati/ricevuti. Tali informazioni vengono poi inviati ad un container InfluxDB , letti e graficati da un container Grafana.

1.5 Descrizione e utilizzo dello Script

L'intero codice è stato scritto in linguaggio di Scripting per bash.

In prima battuta sono stati creati dei container di test, che generano traffico di tipo ICMP (ping) ed UDP (nslookup), disponibili nella cartella yaml.

Lo script principale "start.sh" si occupa di avviare microK8s e far partire i servizi aggiuntivi dns e dashboard, che forniscono tra l'altro un container influxdb e un container grafana pronti all'uso.

Viene successivamente creato un db che conterrà le metriche ricevute, e poi mandati in esecuzione i container che genereranno il traffico da rilevare, ed aperta nel browser di default la pagina di visualizzazione dei grafici in grafana.

Alla prima esecuzione è necessario modificare i parametri di default impostati nella gui di grafana come segue:

- GRAFANA -> DATASOURCES -> INFLUXDB -> NOME DB = monitoring
- GRAFANA -> DASHBOARD -> IMPORT -> Network Stats-1563646561719.json

Lo script ad ogni modo dopo l'apertura del browser procede ed avvia lo script "script.sh" che contiene la logica vera e propria.

Questo difatti verifica i parametri con cui lo script precedente lo ha avviato e controlla che sia effettivamente presente un insieme di container in esecuzione, corrispondenti al label richiesto.

Crea quindi un array associativo fra interfacce virtuali e nome container univoco estraendo le prime dai singoli container. Procede poi all'estrazione delle metriche rilevate per ogni virtual interface dalla macchina host e all'invio delle stesse al InfluxDB. Grafana in autonomia provvede quindi a recuperare le informazioni dal DB e rappresentarle correttamente.

Durante l'esecuzione lo script mostra, anche nel terminale, le metriche ricavate istantaneamente.

1.6 Scelte di progetto

Fra le scelte di progetto l'utilizzo dei servizi dns e dashboard è stato preferito principalmente a causa dell'inesperienza nel campo dei micro-servizi. La soluzione proposta risulta quindi un po' macchinosa nell'unica fase in cui effettivamente l'utente dello script deve interagire.

1.7 Sviluppi futuri

Lo script è già predisposto per l'output delle metriche rilevate in formato JSON RFC 4627 nell'eventualità di futuri sviluppi e/o migliorie.

1.8 Test

Il test della soluzione proposta è stato effettuato su sistema operativo Linux Ubuntu 19.04 con l'ultima versione disponibile di MicroK8s. La macchina su cui è stato effettuato il test dispone di 8GB di RAM e di processore i7 di 4° generazione.