Progetto Gestione di Rete: Ntopng/InfluxDB/Grafana

Raccolta dati e definizione grafici di rete

Cosa ho realizzato?

Il progetto si propone di definire grafi su statistiche di rete attraverso il web tool Grafana sfruttando la raccolta dati realizzata con ntopng. Si è scelto di salvare i dati sotto forma di database InfluxDB.

Ntopng

Ntopng è un software open source per il monitoraggio attivo e passivo della rete. E' la versione moderna di ntop ed introduce degli aspetti innovativi:

- E' meno soggetto ai crash(Robustezza).
- Permette di effettuare cambiamenti di configurazione a runtime senza dover riavviare.
- Aggrega i dati per intervalli di tempo (riesce ha dare velocemente un idea dell'andamento della rete, soprattutto attraverso la media).
- Rispetto ad ntop risulta più veloce e usa meno risorse.
- Categorizza gli host e permette di classificarli per determinati comportamenti assunti.
- Fornisce una chiara visualizzazione di Alerts attaverso una lista di notifiche.
- Permette di definire filtri di rete,pacchetti da scartare, situazioni di allarme etc.

Come utilizzare Ntopng

Installazione:

Dopo aver installato tutti i prerequisiti (**libcap, mysql, etc.**) attraverso il gestore “apt” (Debian) ed aver creato una cartella dove copiare i file ho scaricato il codice di ntopng e ndpi da github .

ho utilizzato la sequenza di comandi da terminale:

git clone <link github>;

./autogen.sh;./configure; make;

/*Per installare una qualsiasi repositories di github*/

Avvio:

Per avviare ntopng ho effettuato l'accesso da root, mi sono recato nella cartella dove sono residenti i file ed ho digitato il comando:

./ntopng -i <nome_interfaccia da monitorare> -w <porta da utilizzare> --dont change user

/* 1. ho specificato l'nterfaccia che volevo monitorare “-i <nome_interfaccia da monitorare>”

2. ho specificato una porta diversa da quella specificata in ntopng.conf in quanto è la stessa porta del servizio Grafana(active daemon) -w <porta da utilizzare>
3. --dont change user (solo per il debug)

   */

Accesso GUI web e Utilizzo:

All'avvio del demone ntopng ho aperto il browser ed scritto nella barra degli indirizzi:

<mio_indirizzo_locale>:<porta_specificata_-w>

ed ho inserito le credenziali di default.

N.B.

Se si utilizza l'indirizzamento dinamico non si ottiene sempre lo stesso indirizzo locale.

La formula alternativa è :

<indirizzo_di_loopback>:<porta_specificata_-w> /*funziona con qualsiasi indirizzo locale*/

Dopo aver esplorato le possibilità di ntopng :

- Categoria Flussi .
- Categoria Host (tiene traccia di utenti, disposivi, reti remote, etc.).
- Categoria Interface (grafi e statistiche sull'interfaccia in esame).
- Categoria System (visualizza lo stato del sistema e del database).
- Alerts (avvisi su possibili misbehaving: flussi terminati, flussi attivi).
- Traffic (grafici su: top talkers, host, application, port, asn. Mappa Host locali).
- Impostazioni (Permette di configurare ntopng e di categorizzare e riconoscere il tipo di traffico).

Attraverso le **impostazioni** ho specificato **Timeseries driver= InfluxDB** (tipo di database su cui si andranno a salvare i rilevamenti).

Ho settato ad **ON** tutti i tipi di informazioni da raccogliere ed ho specificato di raccogliere i dati sul **livello 7 per ogni tipo di classificazione possibile** (più utilizzo di memoria).

A questo punto per avere dati rilevanti ho cercato di far "*parlare la rete*" ovvero far crescere il numero flussi attivi.

Raccolta dati per testing

Per avere una soddisfacente mole di dati su cui lavorare ho alternato stadi di "*apperente quiescenza*" (solo mozilla sulla pagina di ntopng) della rete a intervalli di tempo in cui ho effettuato le seguenti attivita:

- ping :
    1. Repubblica.it
    2. chess.com
    3. facebook.com
    4. Instagram.com

5. Google.com
6. Github.com

   ...

- Streaming:
  1. Siti Mainstream : Netflix,VVVid
  2. Altri (Eurostreaming, Altadefinizione, etc.).
- Navigazione web:
  1. Grafana
  2. Github
  3. Google

     ...

InfluxDB

E' un database basato sulle serie temporali costruito per gestire un carico massiccio di scritture e caricamenti di query ed è adatto per analisi real time. E' in ascolto sulla porta 8086.

Particolarmento usato per analisi sul traffico dati.

InfluxdbQL

E' il linguaggio di query utilizzato su Influxdb è simile ad SQL ma ha delle differenze,soprattuto nella terminologia delle definizioni. Un altra differenza sostanziale è che il tempo è visto come se fosse un indice preselezionato di SQL.

Per effettuare delle query sul database basta utilizzare il comando "**influx**" da terminale

1. Scegliere il Database :
   1. SHOW DATABASES (mosta i database sulla macchina).
   2. USE <nome_database> (imposta come data base nome_database).
2. Esplorazione di Schema
   1. SHOW SERIES (mostra le "righe" salvate).
   2. SHOW TAG KEYS (mostra l'elenco di associazioni tag<--> chiave).
   3. SHOW FIELD KEYS (mostra l'elenco di associazioni field<--> chiave).
   4. SHOW MEASUREMENTS (mostra l'elenco dei nomi delle metriche di misurazione).
   5. CARDINALITY (da postporre alle query sopraindicate per conoscere il numero di righe).

Per l'integrazione con Grafana è importante capire bene come usare la GROUP BY applicata su time (1s) di Grafana e la differenza tra gauge e counter.

- Counter: durante il monitoraggio viene incrementato non appena un altro evento viene filtrato dalla metric (ad esempio conteggio dei byte)

    - non_negative_derivative(metric,1s) GROUP BY time(1s) /* per calcolare l'aumento di velocita su unita di tempo (1s)*/
    - non_negative_difference(metric) GROUP BY time(1s)

      /*per calcolare la differenza (con il prefisso “non_negative_” si considerano solo gli incrementi perchè il decremento avviene nel caso in cui : un contatore viene riazzerato */
- Gauge: una quantita che può aumentare o diminuire nel tempo durante il monitoraggio (ad esempio la morte di un flusso)

Appena ho acquisito un idea d'insieme della morfologia del database ho iniziato una serie di query da CLI per esercitarmi su InfluxdbQL (MySql).

Mi sono concentrato sulla classificazione dei flussi e sui misbehaving che avevo riscontrato, in particolare isolando gli host esterni (volevo capire chi, all'infuori degli host locali, assumesse un cattivo comportamento anche detto *misbehaving*).

Grafana

E' una web application open source che funziona come generica Dashboard e compositore di grafi.

Supporta influxdb. Lavora sulla porta standard 3000.

Ho installato grafana da github con le stesse modalita sopra descritte per ntopng.

Offre un interfaccia grafica al servizio della definizione di grafi temporali attraverso opportune query su un database preimpostato.

Le query possono essere fatte con la ***GUI*** attraverso schemi di query predefiniti, oppure si può optare per ***Toggle mode*** che permette di inserire la query testualmente con le stesse modalita del CLI.

La visualizzazione della risposta alla query si può ottenere oltre che con il grafo temporale anche in formato json.

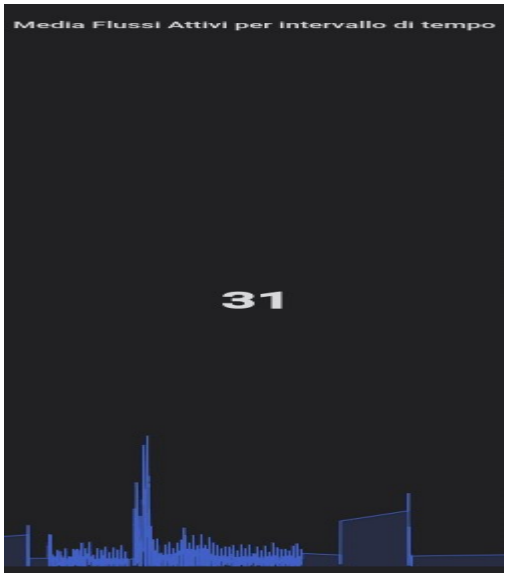
Grafana fornisce vari tipi di grafi (estendibili con appositi plugin).

I grafi che ho utilizzato maggiormante sono:

1. **Grafico di funzione(mostra average di default sull'intervallo campione)**
2. **Single State (La query restiruiste un numero)**
3. **Row (una semplice lista temporale)**

Di seguito alcuni grafici della mia Dashboard:

Flussi Attivi:

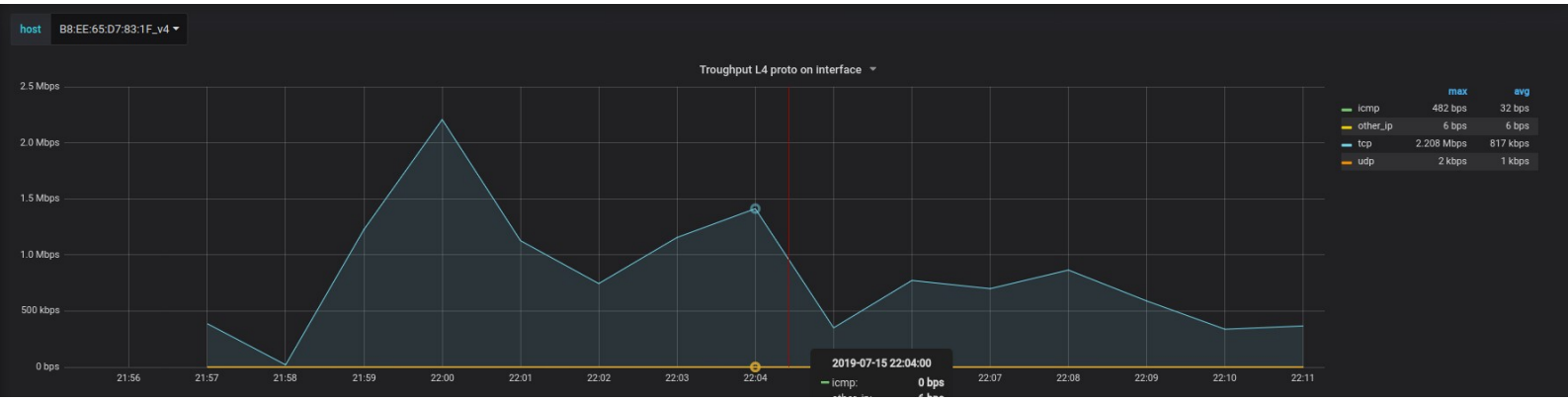


Nuovi flussi nell'intervallo di tempo:

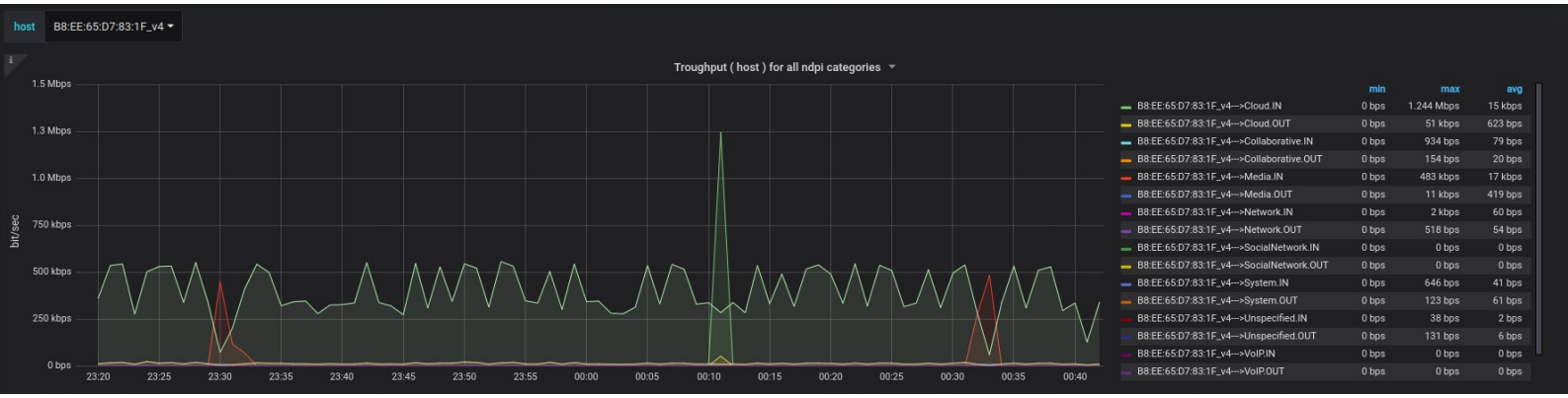
Nuovi Misbehaving Flows nell'intervallo di tempo:

Statistiche Tcp sulla gestione dei pacchetti(Pacchetti per host IN,Out: Out of Order, Lost ,Retransmitted):

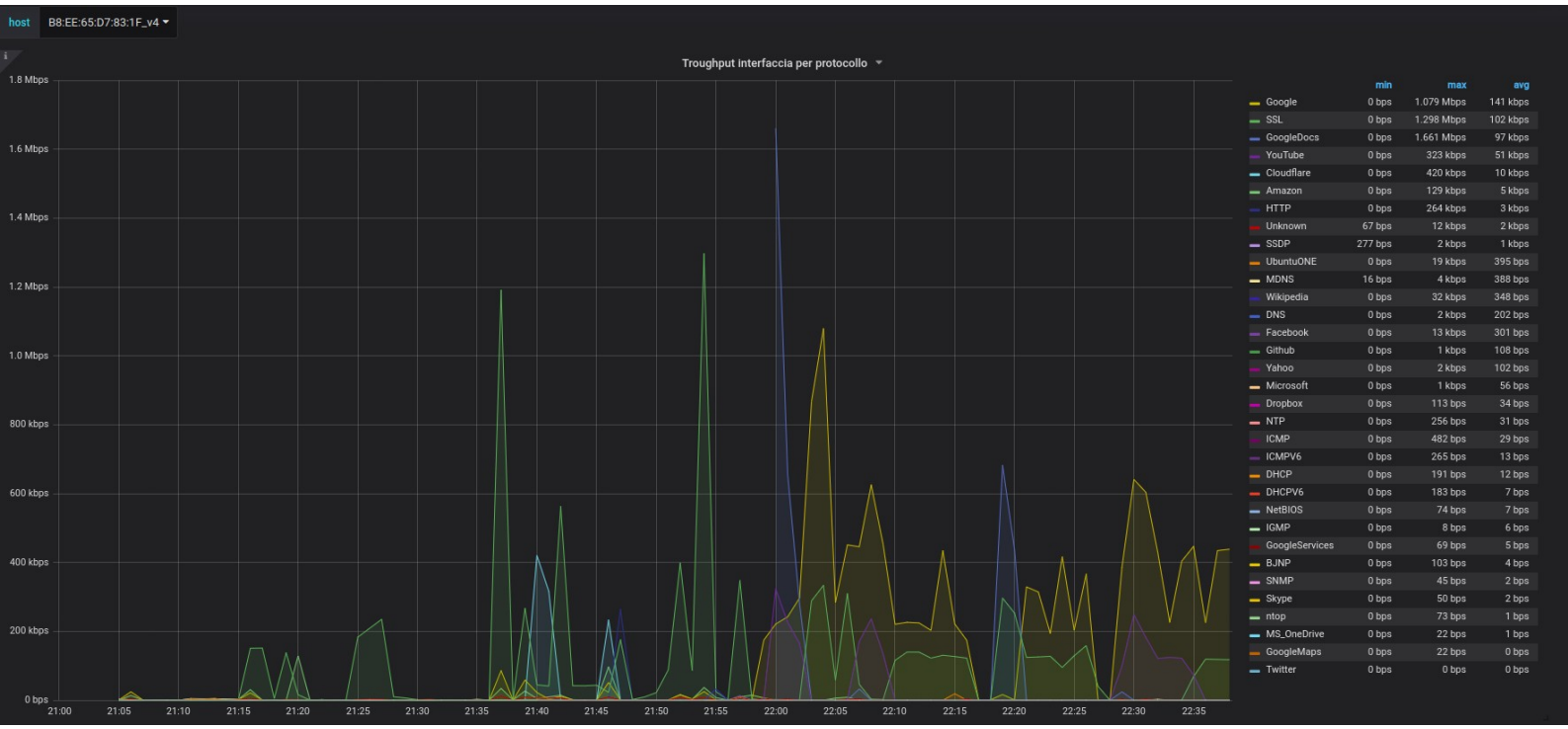
Troughput Protocolli Livello 4 (per 'host da selezionare nel menù tendina'):



Troughput ndpi_categories(per 'host da selezionare nel menù tendina'):



Troughput ndpi_protocols sull'intera interfaccia monitorata:



File di configurazione Dashboard

Dopo aver definitola mia dashboard ho salvato il file di configurazione in formato json attraverso l'operazione:

share dashboard-->export-->save to file

E' possibile anche salvare il file copiando manualmente il contenuto del **json** e incollandolo in un file di testo.

Tale file puo essere importato su un altra macchina e riutilizzato su altri databases .