Proctraffic

Una applicazione pratica delle potenzialità di Sysdig

Introduzione

L'incremento di servizi sempre più difficili da monitorare richiede un cambio di approccio al modo di analizzare il comportamento di una rete e dei suoi componenti.

Negli ultimi anni infatti si sta assistendo ad un aumento esponenziale del numero di servizi cloud based in cui non si ha più una visione così immediata della struttura della rete (un indirizzo IP ormai identifica un servizio e non una macchina fisica a cui ci si va a collegare) e di come questa vada ad evolversi.

Potremmo fare due esempi su tutti:

- ***Elastic Compute:*** in cui si va ad acquistare una certa quantità di risorse le quali vengono adeguate alle reali esigenze per il servizio. Le macchine virtuali necessarie sono variabili in numero e locazione, nella maggior parte dei casi a controllarne l'attivazione è il fornitore stesso.
  Per esempio la richiesta per un servizio durante la notte in Europa potrebbe essere bassa mentre nello stesso momento in America (essendo ad inizio giornata) potrebbe essere più alta, questo fa si che il fornitore automaticamente aumenti la disponibilità di macchine virtuali in America e decrementi quella in Europa.
- ***SDN (Software Defined Network)*****:** sono reti dinamiche che vengono create sopra una rete fisica, realizzando di fatto una astrazione che poggia le sue basi sulla rete realmente esistente.
  Il livello di astrazione detto Control Layer offre quindi una gestione del pacchetto a livello di rete e non più a quello del singolo apparato.
  In questo modo è possibile per un amministrazione di rete cambiare il modo in cui il control layer va a mapparsi sulla rete fisica migliorando lo sfruttamento da parte di essa delle risorse di rete mediante la definizione del comportamento da seguire per ogni possibile flusso.

Risulta evidente che un'analisi della rete che sfrutti i flussi si trasforma in un ardua impresa, basti pensare che nel caso di un servizio di Elastic Compute non è possibile fare alcuna assunzione sulla specifica macchina su cui viene eseguito il nostro servizio e sulla rete a cui questo è collegato essendo queste due caratteristiche legate al gestore del servizio.

System troubleshooting versione 2.0

Il monitoraggio del traffico ha come compito quello di andare ad analizzare la rete in punti nevralgici al fine di andare a migliorare le prestazioni o cercare la causa di malfunzionamenti.

Spesso questo approccio non permette di risalire con precisione alla causa del problema.

Per esempio in un Server, normalmente è possibile sapere come il traffico si sta direzionando verso questo, ma non si riesce a conoscere quali processi o servizi vadano realmente a gestirlo o come questo vada ad impattare sulle risorse disponibili sulla macchina.

In questo tipo di analisi quindi il sistema risulta completamente trasparente, contrariamente a quello di cui si ha bisogno in un sistema Cloud Based, in risulta fondamentale avere informazione su come si sta comportando il servizio indipendentemente dalla rete su cui è locato sulla quale non è possibile fare alcuna assunzione.

Insomma quello di cui c'è bisogno è un sistema che permetta di andare ad analizzare il comportamento dei servizi dall'interno, magari osservando l' interazione reciproca tra i processi.

Sysdig

Sysdig è un piccolo (ma potente) software open source sviluppato da Draios che va ad installarsi nel kernel di Linux andando ad interporsi tra le chiamate di sistema ed il loro ritorno, riuscendo a tracciare ognuna di queste.

La grande forza di Sysdig sta proprio nel poter andare a vedere cosa succede "dietro le quinte", ossia analizzare come i processi (o thread) vadano a sfruttare le chiamate di sistema.

Al fine di rendere meno verboso il reporting di Sysdig è possibile impostare una serie di filtri per la personalizzazione dell' output desiderato.

Dopo questa breve descrizione risulta immediato come Sysdig sia "Cloud Friendly" .

Per poter localizzare un problema nell'erogazione del servizio è possibile analizzare le chiamate di sistema coinvolte nella sua esecuzione.

Chisel

"Sysdig's chisels are little scripts that analyze the sysdig event stream to perform useful actions."

Per poter rendere poi il software ancora più flessibile è stata data la possibilità di creare dei piccoli script in Lua detti "chisel".

Sysdig al suo interno contiene una versione precompilata dell'interprete Lua, ciò garantisce la portabilità degli script su tutti i sistemi in cui è presente Sysdig.

Il chisel ProcTraffic

L'idea

Quello che manca nell'analizzare il traffico è sapere con precisione quali processi lo vanno a generare.

Nel localizzare un problema avere uno strumento che permetta di conoscere quali processi stanno generando traffico in un particolare momento, può aiutare a individuare comportamenti anomali che possono (per esempio) mandare in crisi un servizio o saturare la rete.

Il progetto si basa sulla scrittura di un chisel (PT2.lua) che svolge due compiti: ogni n secondi stampa a video l'ultimo report disponibile e salva su un file tutti i report tracciati.

La realizzazione

Le strutture dati

I dati ricavati da ogni processo vengono salvati in una serie di tabelle <chiave,valore>. Tali dati sono:

- L'elenco dei processi da monitorare
- L'ultima rilevazione di dati in ingresso ed uscita (alla fine di ogni intervallo questa tabella viene svuotata)
- Il totale dei dati ricevuti ed inviati
- La penultima rilevazione disponibile in ingresso ed uscita

Con l'ultima struttura dati si cerca di dare un andamento del traffico mediante confronto con l' ultimo report. In particolare:

- Un valore positivo indica che rispetto all'ultimo report il traffico generato è maggiore (cioè il traffico generato sta aumentando).
- Un valore negativo indica che rispetto all'ultimo report il traffico generato è minore (il traffico generato sta diminuendo)

Parametri

L'unico parametro necessario è Il numero di secondi (ossia la dimensione dell'intervallo) tra un report e l'altro: questo permette all'utente di decidere la granularità dei report. La validità del parametro viene controllata nella funzione on_set_args nalla quale viene anche istanziato.

Quale traffico analizza

Il chisel applica ai risultati di sysdig il seguente filtro:

" evt.type="recv" or evt.type="recvmsg" or evt.type="recvfrom" or evt.type="send" or evt.type="sendto" and fd.type="ipv4" "

- in ingresso analizza le chiamate: recv, recvmsg, rcvfrom
- In uscita invece: send, sendto

Gli output

Il file di output

Ogni file di output è rinominato con la seguente sintassi:

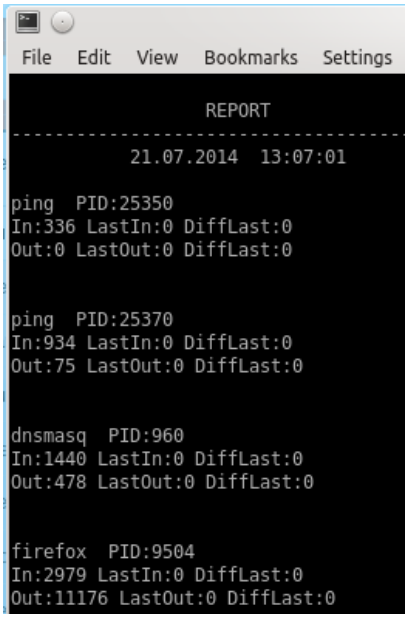
Log_data_ora

dove la data e l'ora sono quelle relative all' avvio del monitoraggio.

Ogni volta che viene creato un nuovo report questo viene aggiunto in append alla fine del file. Ogni report riporta data e ora della rilevazione.

Ogni report oltre ad essere scritto nel file di log è mostrato a video con lo stesso layout.

L'output video



Il chisel va a stampare l'elenco dei processi che hanno generato traffico durante l'ultimo intervallo indicando:

Nome del processo
In/out: totale byte in ingresso/uscita
LastIn/LastOut: totale byte ultima rilevazione
DiffLast: differenza rispetto all'ultima rilevazione

Conclusioni

La realizzazione di questo chisel ha messo in evidenza le grandi potenzialità di Sysdig, con un piccolo script è stato possibile realizzare un programma che altrimenti avrebbe richiesto tempi e risorse di realizzazione sicuramente più elevate.

Sysdig si propone quindi come una idea semplice ma geniale (come quasi tutte le idee vincenti) per l'analisi del traffico. Nonostante vi siano ancora alcune difficoltà derivanti soprattutto dall'uso di un interprete precompilato nel software (un esempio su tutti il fatto che non ci sia nessuna libreria di comunicazione come lua-socket), l'idea di utilizzare i chiesel risulta una semplificazione ben strutturata per creare ottime (e rapide) applicazioni per il monitoraggio.