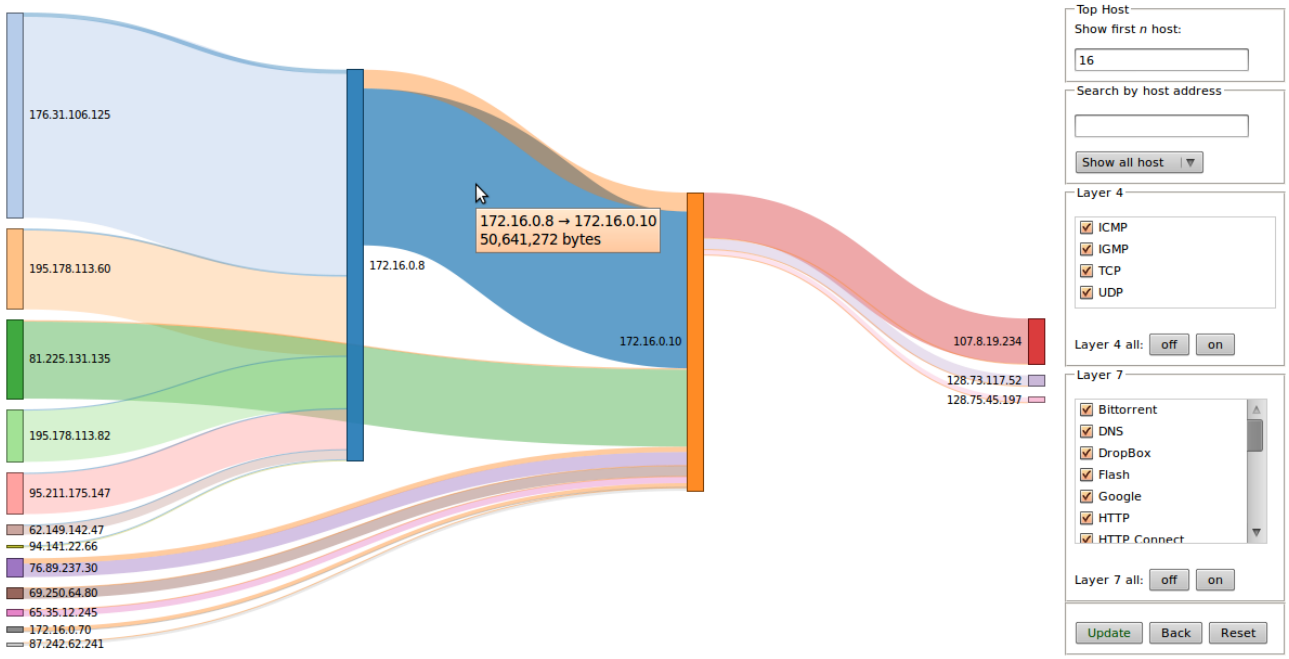
Relazione progetto S.G.R. a.a. 2011/2012

Autore Giulio Tranchida, n° matricola 241732.

Il diagramma “Sankey” è uno specifico diagramma di flusso, nel quale viene mostrata la larghezza delle frecce in proporzione alla quantità di flusso. Generalmente sono utilizzati per visualizzare le quantità di energia o materiali o costi tra i processi.

In questo programma, Network Flows, il diagramma di flusso Sankey viene utilizzato per mostrare le quantità di dati (flussi di rete), espressi in bytes, transitati tra uno o più host.



Un flusso di rete è definito come un insieme di pacchetti legati da proprietà comuni, come ad esempio: stesso protocollo, indirizzo ip, porta, ecc...

Si può facilmente notare che in questo primo diagramma non si evidenziano le informazioni dei protocolli presenti nei flussi. Si è scelto di proposito di dare, in primo luogo, una visione d'insieme dei flussi maggiori della rete lasciando, tuttavia, la possibilità di cliccare su un particolare flusso per aumentare il livello di dettaglio dei dati ed, eventualmente, di filtrarli selezionando solo i protocolli voluti.

In questo primo diagramma tutti i flussi di rete tra due host sono accorpati, le uniche informazioni mostrate sul flusso sono la sua direzione e dimensione.

Il programma Network Flows nella sua versione dimostrativa, acquisisce le informazioni da visualizzare da un file, dove ogni riga è definita come:

host sorgente | host destinazione | bytes in uscita | bytes in ingresso | protocollo layer 4 | protocollo layer 7

Questo file viene elaborato dallo script python *sankey.py* che produrrà un output in formato json. I dati sono organizzati in:

- una lista di nodi, dove ogni nodo è definito da un nome rappresentante l'indirizzo ip dell'host, altrimenti il suo nome risolto, seguita da
- una lista di archi, dove per ogni arco è definito il suo nodo di origine, il suo nodo di destinazione, il suo peso e la sua direzione.
- meta informazioni, che definiscono:
  - il numero massimo di host che si è scelto di visualizzare nel diagramma;
  - l'ordine di misura del peso di un arco;
  - i protocolli layer 4 e layer 7 presenti nei flussi dati;
  - i sottoinsiemi dei predetti protocolli che si è scelto di visualizzare nel diagramma;
  - più alcune variabili di stato.

Lo script sankey.py computa i dati letti da file creando, per ogni nuova tupla di indirizzi degli host sorgente/destinazione, un nuovo elemento nell'hashmap *archi*, se la tupla è già presente i dati verranno sommati a quelli esistenti.

Per ogni chiave (sorgente/destinazione) vengono memorizzate più informazioni:

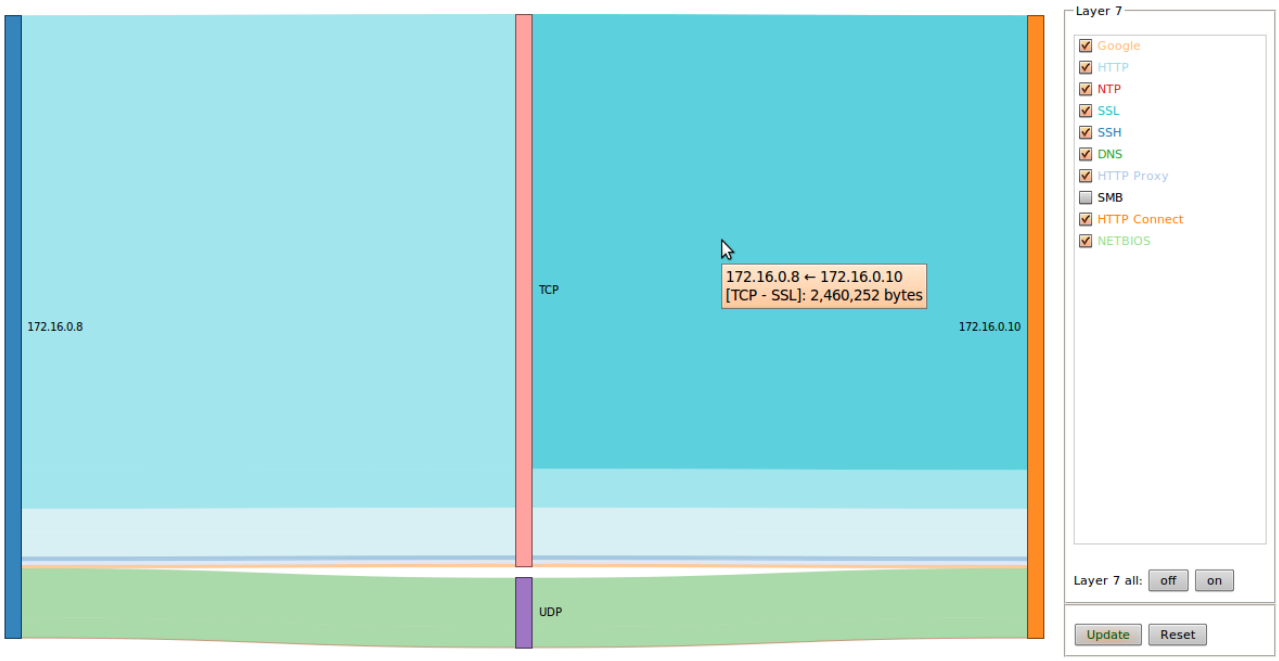
- la somma dei byte in ingresso e in uscita; dato che poi verrà usato per rappresentare il peso di quell'arco, sia per ordinare i flussi in ordine decrescente,
- i byte in ingresso;
- i byte in uscita;
- un hashmap, rappresentante i protocolli layer 4, la cui chiave è il nome del protocollo del livello di trasporto, che, a sua volta, memorizzerà:
  - la somma dei byte in ingresso e uscita per quel protocollo e
  - un'ultima hashmap, rappresentante i protocolli layer 7, le cui chiavi sono i nomi dei protocolli del livello applicativo, che, a sua volta, memorizzerà:
    - la somma dei byte in ingresso e in uscita;
    - i byte in ingresso;
    - i byte in uscita per ogni protocollo layer 7.

La ragione di questo gioco di matriosche è di poter dare all'utente, una volta valutata la visione d'insieme dei flussi di rete, la possibilità di avere delle informazioni dettagliate su un particolare flusso, tra due specifici host, cliccandovi sopra.

Il diagramma di flusso sankey ha dunque due visualizzazioni.

La prima, nella quale ogni host è definito come un rettangolo a cui viene associato un nome e un colore che lo identificano. In questa visualizzazione ogni rettangolo avrà almeno due archi uno uscente e l'altro entrante (almeno di non trovarci nel caso limite di un flusso in un'unica direzione *one-way flow*), in cui l'arco dello stesso colore rappresenterà sempre i dati in uscita per quel nodo e l'arco di colore diverso rappresenterà il flusso di dati in ingresso avente come origine il suo host di destinazione.

Nella seconda modalità di visualizzazione si potranno vedere uno o più archi, dove ogni arco rappresenta il protocollo applicativo, mentre, il colore non identifica più la direzione del flusso bensì il protocollo layer 7. I rettangoli al centro del diagramma identificano, invece, i protocolli del livello di trasporto.



È comunque sempre possibile selezionare un sottoinsieme di protocolli o definire un numero massimo di nodi da visualizzare al fine di raffinare la rappresentazione dei dati nel diagramma di flusso in base alle esigenze dell'utente.

Il compito di visualizzare i dati, una volta che lo script sankey.py ha finito di elaborarli, è lasciato alla libreria grafica *d3.js*, la quale produrrà un grafico in formato *svg*.

É opportuno far notare che il codice che disegna il grafo di flusso sankey non è attualmente presente nella libreria d3, poiché è ancora in fase di sviluppo. Lo sviluppatore della libreria, Mike Bostock, prevede in futuro un possibile miglioramento dell'algoritmo al fine di minimizzare la sovrapposizione dei collegamenti o per supportare grafi ciclici[1], come i cicli al primo hop o connessioni di loopback.

In merito agli archi di ritorno, per ovviare al problema ho scelto di confrontare, come stringhe, per ogni arco la sua sorgente e la sua destinazione. Se la sorgente è maggiore della destinazione la computazione dei dati rimane invariata, altrimenti, viene eseguito il complementare dei dati, invertendo la tupla sorgente/destinazione e viene eseguita la somma dei dati in ingresso alla variabile che conta i dati in uscita e viceversa.

Pisa, li 09/07/2012

F.to Giulio Tranchida

1 http://bost.ocks.org/mike/sankey/