

Lavori in corso su TCP/http

L'evoluzione dei contenuti sul Web e la crescente diffusione di applicazioni multimediali hanno messo in crisi la tipica comunicazione Web via TCP e http. Vediamo cosa offre il protocollo Spdy



La comunicazione sul Web avviene oggi principalmente attraverso i due noti standard TCP e http. Il primo (Transmission Control Protocol), introdotto nel 1974, definisce un protocollo di connessione affidabile a livello di trasporto con funzionalità, tra le altre, di controllo di flusso, gestione della congestione di rete, distribuzione in-order dei messaggi. Il secondo (Hypertext Transfer Protocol), invece, opera a livello applicativo sulla base di un meccanismo di comunicazione di tipo richiesta/risposta instaurato tra nodi client e server di una connessione di rete. L'evoluzione dei contenuti sul Web, con la crescente diffusione di applicazioni multimediali, ha tuttavia recentemente messo in crisi questo approccio, evidenziando una serie di limiti che non potevano certo essere immaginati 10 e più anni fa. Tra le limitazioni oggi più significative di http si segnalano, in particolare, la gestione di una sola richiesta per ogni connessione (dal 2008 molti dei browser hanno ridotto l'incidenza di questo, accettando connessioni multiple a un server), la possibilità di iniziare richieste dal solo nodo client, l'assenza di compressione dell'header dei messaggi (in alcuni casi l'header può superare i 2 kB). Per questi motivi, nell'ultimo periodo è cresciuto significativamente l'interesse verso protocolli alternativi o soluzioni in grado di adattare http alla configurazione attuale del Web. Tra questi, uno dei più interessanti è certamente il protocollo Spdy (pronunciato 'SpeeDY'), introdotto in via sperimentale da Google.

Recentemente (febbraio 2012) è stata riemessa una versione Internet-Draft dello standard scaricabile gratuitamente dalla rete (si veda link (1) fra le fonti). Gli Internet-Draft sono documenti di lavoro della left (Internet Engineering Task Force) aventi validità di soli 6 mesi, non riferibili come 'work in progress', aggiornabili o sostituibili in qualsiasi momento.

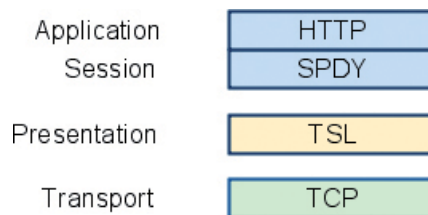
L'obiettivo dello standard Spdy è, dunque, la definizione di un protocollo a livello applicativo che riduca la latenza di trasmissione e la banda richiesta nella trasmissione su Web. In realtà, Spdy si basa sul livello TSL (Transport Secure Layer) e introduce uno strato ulteriore tra questo e http per migliorarne le prestazioni (si veda figura 1). Il livello TSL è un protocollo

di crittografia (standard left - RFC 5246) basato sul vecchio standard SSL di Netscape e pensato per garantire la sicurezza nella comunicazione su rete da eavesdropping e tampering. Il livello di trasporto resta invece TCP; modifiche in questo senso avrebbero al momento impatti troppo grossi sull'intera infrastruttura di rete.

L'left Working Group sta anche lavorando alla revisione 2.0 del protocollo http (in prospettiva soprattutto del crescente accesso al Web dalle piattaforma mobile) e Spdy è tra i principali candidati a essere incluso nella nuova specifica, in contrapposizione con la tecnologia Speed+Mobility proposta da Microsoft. La diffusione dello standard Spdy è ovviamente ancora piuttosto limitata. Secondo alcune fonti (si veda link (2) fra le fonti) a oggi alcuni servizi di Google (come Google Search, Gmail e simili) adottano il protocollo quando disponibile,

mentre Twitter lo ha abilitato sui propri server in marzo 2012, come pure hanno fatto altri progetti open source come Jetty Web Server, Apache o node.js.

Figura 1: la posizione del livello Spdy nella comunicazione su Web



Come funziona il protocollo

Nello specifico, Spdy tenta di conservare la semantica esistente del protocollo http, adattandosi in particolare alle revisione 1.1 di quest'ultimo, ridefinendo soltanto il modo in cui i dati vengono scambiati sulla rete. Definisce due diversi livelli: quello di framing specifica le modalità di multiplexing di più stream su una singola connessione TCP; in questo modo, l'efficienza nell'utilizzo del canale migliora, riducendo il numero di connessioni e incrementando la densità dei pacchetti scambiati. Il livello http, invece, descrive il meccanismo con il quale



Fonte: www.bloggerspath.com

il protocollo di connessione richiesta/risposta di http viene mappato sul livello di framing di Spdy.

Nello standard Spdy, poi, la comunicazione client/server avviene mediante stream, ovvero sequenze bidirezionali di dati strutturati in frame e trasmessi al di sopra di una connessione persistente, ovvero il server lascia aperta la connessione TCP dopo l'invio di una risposta e i successivi messaggi vengono spediti su questa stessa.

A differenza di quanto avviene nel protocollo http, gli stream possono essere creati da client o server ed essere trasmessi in maniera concorrente su una stessa connessione e possono essere cancellati. La gestione degli stream avviene mediante speciali frame di controllo, come 'SYN_Stream' per l'apertura dello stream,

supportata per i soli frame di controllo e garantendo comunque il minimo di 8.192 B. A ogni stream è associato un identificativo univoco a 31 bit, incrementato monotonamente, e un livello di priorità, compreso tra 0 e 7. Se tutti gli identificativi sono utilizzati, non è possibile aprire un nuovo stream sulla connessione; il nodo ricevente deve ovviamente implementare una capacità di riconoscimento di richiesta di apertura di stream già attivi e rispondere a questa mediante un messaggio di errore, inviato attraverso un frame di controllo di tipo 'Protocol_Error'. Allo stesso modo, deve essere segnalata una condizione di errore nel caso di ricezione di frame di dati per stream non aperti.

L'identificativo dello stream viene incluso nell'header dei frame dati a esso associati; il livello di priorità serve così a indicare al nodo destinatario quale stream processare prima nel caso di ricezione di stream concorrenti. Oltre all'identificativo proprio nell'header è possibile specificare l'ID di uno stream in qualche modo associato; ciò è usato nel meccanismo di push delle risorse da parte di un server. Dopo la creazione di uno stream è possibile inviare immediatamente header e dati per quello, senza attendere l'autorizzazione dall'endpoint connesso; un'eventuale successiva risposta di rifiuto di apertura dello stream deve quindi essere opportunamente gestita. Una volta creato, uno stream resta attivo fino a che non viene cancellato in condizioni nominali, perché entrambi i nodi connessi non hanno necessità di inviare ulteriori dati, o non nominali, in caso di errore o per chiusura (teardown) della connessione TCP stessa.

Nel primo caso, l'ultimo frame di dati inviato da ogni nodo deve essere segnalato mediante opportuna flag attivata nel campo di header. Ovviamente dopo tale segnalazione il nodo non è più autorizzato a inviare ulteriori frame dati sullo stream corrente; lo stream può comunque essere lasciato aperto dall'altro nodo, se ne ha necessità.

Alcuni meccanismi di Spdy

Il protocollo Spdy introduce anche alcuni interessanti meccanismi basilari di controllo di flusso. Ogni nodo, per esempio, può segnalare sulla rete di non essere disponibile ad accettare ulteriori stream sulla connessione corrente, mediante un opportuno frame di controllo di tipo 'Goaway'. Ciò consente di evitare che il nodo sia intasato e non riesca a rispondere a tutte le richieste ricevute. È tuttavia consentito inviare stream ulteriori su altre

connessioni, quando queste siano disponibili. Lo stesso numero massimo di stream che possono essere tenuti aperti concorrentemente su una stessa connessione può essere modificato mediante frame di controllo di tipo 'Settings', utilizzato pure per impostare altri parametri della connessione, tra i quali la banda di download/upload stimata sul canale o il tempo di round trip. Anche il numero di byte di frame dati che possono essere inviati sullo stream è soggetto al controllo di flusso. Il nodo trasmettitore implementa per questo un contatore a 32 bit che indica il numero di byte di dati che gli è consentito inviare. Al momento della creazione dello stream



L'evoluzione dei contenuti sul Web, con la crescente diffusione di applicazioni multimediali, ha evidenziato i limiti di dello standard TCP/ http, pensato 10 e più anni fa

'SYN_Reply' per la conferma di ciò, con la possibilità da parte del nodo ricevente di rigettare l'operazione segnalando l'evento mediante l'invio di un frame di tipo 'Refused_Stream', e 'SYN_RST' per la chiusura dello stesso.

I frame dati sono utilizzati per scambiare le informazioni vere e proprie. Entrambi i tipi di frame, di controllo e dati, sono composti da una parte di header, avente lunghezza di 8 byte, e da una parte di payload di lunghezza variabile (specificata dall'header), che arriva a un massimo di 16 MB. Per ridurre le richieste di memoria locale è tuttavia previsto che i singoli nodi possano imporre limitazioni alla dimensione massima

Fonte: www.techtimely.files.wordpress.com

questo viene impostato al valore di default di 64 kB, ma tale valore può essere modificato inviando un frame di controllo di tipo 'Settings'.

Dopo l'invio di un frame dati, il nodo trasmettitore decrementa il contatore suddetto del numero di byte spediti; ovviamente, nel momento in cui il contatore raggiunge il valore 0, il nodo non è più abilitato a trasmettere dati sullo stream specifico. Il valore del contatore può essere quindi incrementato dal nodo ricevente inviando il frame di controllo 'Window_Update', così da accordare ulteriore credito di trasmissione. Il meccanismo funziona soltanto tra i due end-point di un hop della rete, non viene propagato lungo l'intera connessione, e non si applica ai frame di controllo: ogni nodo deve essere in grado di ricevere tutti i frame di controllo che gli vengono inviati.

Altra caratteristica interessante del protocollo Spdy è la compressione dei meta-data di header. Tali meta-data possono essere trasferiti all'interno dei campi 'Name/Value' che sono presenti sia nei frame di controllo di tipo 'SYN_Stream'/'SYN_Reply', usati per l'apertura di uno stream, sia in quelli di 'Headers', dedicati proprio a tale scopo durante la normale comunicazione. Per ogni frame possono essere presenti più di uno di tali campi. L'adozione di un meccanismo di compressione assume rilievo in tutte quelle comunicazioni in cui



Fonte: www.webmonkey.com

Per una prima implementazione di Google Chrome si può avere una riduzione del tempo di caricamento del 60% rispetto a quanto avuto con il solo protocollo TCP

l'header http è di elevata lunghezza; l'algoritmo adottato è zlib. L'header proprio dei frame Spdy di controllo e dati non viene invece compresso, vista la sua ridotta dimensione, di soli 8 B, con un overhead dello 0,6% per frame con payload di dimensione di 1.452 B. La compressione non viene neppure applicata ai byte di payload, in quanto per essere efficiente dovrebbe assumere ipotesi non sempre soddisfatta sulla natura di questi. Lo standard Spdy consente dunque di mappare il protocollo request/response dello standard http in maniera più efficiente sul canale di connessione. Il meccanismo è piuttosto semplice, ma trae beneficio da alcuni accorgimenti derivati proprio dalle caratteristiche intrinseche di Spdy. In corrispondenza di una richiesta inoltrata dal client al server,



Fonte: www.techsanity.com

L'obiettivo di Spdy è definire un protocollo a livello applicativo che riduca la latenza di trasmissione e la banda richiesta nella trasmissione su Web

per esempio, il livello Spdy apre uno stream inviando il frame di controllo 'SYN_Frame' e copiando, in formato compresso, all'interno dei campi 'Name/Value' di questo sia l'header della richiesta http, sia le informazioni accessorie, tra le quali la porta host e lo schema dell'URL richiesto. Il server risponde quindi con un frame di controllo 'SYN_Reply', indicando nel campo 'Name/Value' di questo lo status code della richiesta e la versione del protocollo http supportata. Nel creare la connessione, il protocollo Spdy supporta l'autenticazione del server in base agli schemi Basic, Digest, Ntlm e Negotiate, i primi due sono di tipo stateless, i secondi di tipo stateful. La possibilità di avere stream multipli su una stessa connessione e soprattutto di richiedere l'apertura di uno stream anche da parte del server consente di implementare nello standard Spdy un meccanismo di push delle risorse per ridurre il numero delle richieste di un client al server. In altri termini, ove ne riconosca l'utilità e la validità il server può inviare risposte multiple al client in corrispondenza di una sola richiesta ricevuta; ogni risposta sarà associata a uno stream diverso. Per consentire la corretta associazione, i frame dati di ogni stream conterranno nello header Spdy, oltre al proprio identificativo, l'ID dello stream della richiesta inoltrata dal client. Il pushing delle risorse evita che il client debba inoltrare richieste ulteriori dopo aver ricevuto la risposta primaria e riconosciuto la necessità di ricevere risorse aggiuntive dal server.

Vantaggi alla mano

La diffusione dello standard Spdy è ancora piuttosto limitata, ma i risultati delle prime implementazioni mostrano risultati incoraggianti. Per esempio, per una prima implementazione di Google Chrome, che adotta http su Spdy (si veda link (3) fra le fonti, dove è disponibile anche il link per il download del codice sorgente del browser), si è visto che, eseguendo test di download di 100 'top Website' su connessioni di rete simulate con percentuale di pacchetti persi dell'1%, si ha una riduzione del tempo di caricamento fino al 60% rispetto a quanto avuto con il solo protocollo TCP. La sola compressione degli header comporta una riduzione di tale tempo anche superiore a 1 s.

(*) fonte: <http://tools.ietf.org/html/draft-mbelshe-httpbis-spdy-00> (1); <http://en.wikipedia.org/wiki/SPDY> (2); <http://dev.chromium.org/spdy/spdy-whitepaper> (3)