Progettazione di Servizi Web e Reti di Calcolatori

Politecnico di Torino – Prof. Antonio Lioy

AA 2019-2020, esercitazione di laboratorio n.2

Prompt dei comandi

Per usare l'applicativo **nslookup** è necessario aprire una finestra di **prompt dei comandi** analogamente a quanto fatto per **netstat, wget, ps** e **tasklist** (si faccia riferimento all'introduzione proposta nella esercitazione n.1 per un esempio su come aprire un prompt di comandi). Anche questo comando è disponibile per Windows, macOS e Linux.

Introduzione all'uso di "nslookup"

L'applicativo **nslookup** permette di eseguire interrogazioni al sistema DNS.

Comando per la risoluzione dell'host usando il server di default (modalità non interattiva):

nslookup *[opzioni …] host*

Comando per la risoluzione dell'host usando il server specificato (modalità non interattiva):

nslookup *[opzioni …] host serverDNS*

Dove *serverDNS* può essere specificato sia come indirizzo IP (preferibile) sia tramite il suo Fully Qualified Domain Name (FQDN – cioè il nome "univoco" di un host all'interno della gerarchia DNS). L'opzione del comando nslookup rilevante per questa esercitazione è *-q=tipo*, la quale permette di specificare il tipo di query da effettuare. Il *tipo* può assumere i seguenti valori: A (indirizzo IPv4), AAAA (indirizzo IPv6), A+AAAA (indirizzo IPv4 + IPv6), ANY, CNAME, MX, NS, SOA, PTR, HINFO, TXT, WKS.

Un esempio di comando ammissibile sarà quindi

nslookup *–q=A www.google.com*

Per ulteriori approfondimenti si rimanda alle slide del corso.

Si ricorda che, per eseguire le interrogazioni al DNS, bisogna usare il comando **nslookup** dopo aver aperto una finestra di **prompt dei comandi**.

Introduzione all'applicativo "JPerf"

L'applicativo **JPerf** permette di misurare il throughput di rete per i protocolli TCP ed UDP tra due nodi (uno configurato come *client* e l'altro come *server*). L'applicativo dispone di un'interfaccia grafica che ne semplifica l'uso. In particolare è possibile specificare:

- i parametri della trasmissione dati e le relative opzioni (tra cui l'indirizzo IP del server Jperf a cui collegarsi);
- il protocollo da usarsi (TCP o UDP);
- il numero di "stream paralleli" (cioè numero di trasmissioni indipendenti e contemporanee).

JPerf permette di impostare numerosi altri parametri che esulano dal contesto di questa esercitazione, per cui non dovranno essere modificati.

Nella modalità *client* è necessario specificare:

1. l'hostname oppure l'indirizzo IP del server JPerf;
2. la porta sulla quale il server Jperf è in ascolto;
3. il protocollo da usare, TCP o UDP.

Nella modalità *server* è necessario specificare:

1. la porta sulla quale il server JPerf deve mettersi in ascolto;

2. il protocollo da usare, TCP o UDP.

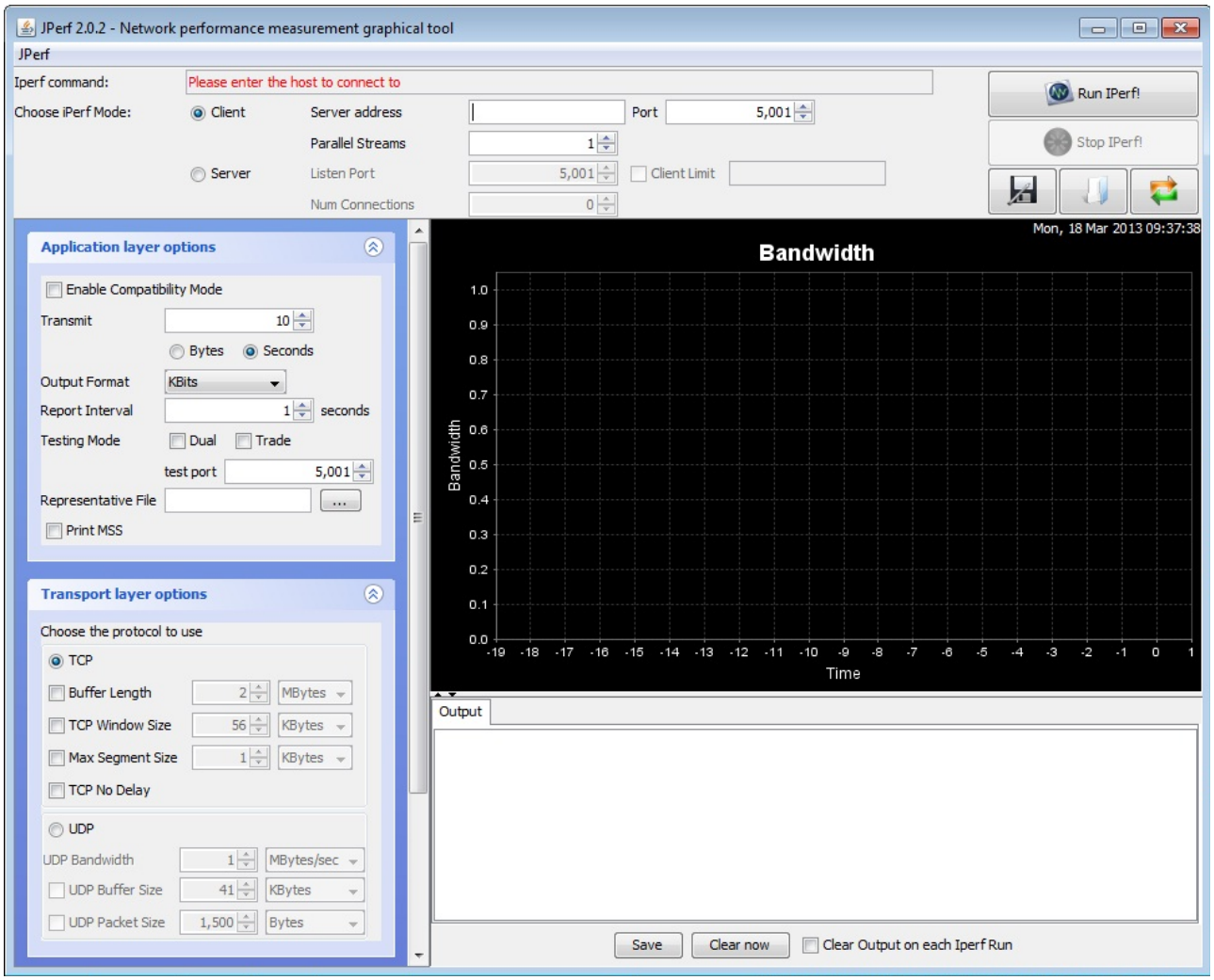

Figura 1: finestra dell'applicazione Jperf.

Per poter utilizzare JPerf, occorre scaricare il package jperf-2.0.0.zip dal link https://sourceforge.net/projects/iperf/files/jperf/jperf%202.0.0/

***NOTA**: Essendo JPerf un applicativo Java, per poterlo eseguire è necessario avere Java Runtime Environment (JRE) installato.*

JRE può essere scaricato, ad esempio, da https://www.java.com/download.

Esercizio 2.1

Il servizio DNS permette di mantenere la corrispondenza tra indirizzi IP e nomi attraverso interrogazioni dirette (da nome a IP) ed inverse (da IP a nome). L'architettura gerarchica e l'implementazione distribuita rendono il servizio DNS robusto e scalabile. Questa struttura influenza la risoluzione, in particolare il flusso query e risposte, secondo quanto visto a lezione.

Al fine di comprendere meglio il processo di risoluzione da nome a indirizzo IP, si eseguano in modalità manuale le query che effettuerebbe un server DNS locale per "risolvere" *www.google.com*.

1. Si effettui la query DNS usando il comando **nslookup** e specificando un root server a scelta (lista completa: http://root-servers.org/) ed il nome "www.google.com.". Quali server possono essere consultati per il "top level domain" (TLD) "com."?
2. Scegliere un server responsabile della gestione del TLD "com.". Effettuare la precedente query specificando il nuovo server DNS. Compilare la lista dei server DNS disponibili:

3. In modo simile ai punti precedente, scegliere un server dalla lista e identificare tutti gli indirizzi IP associati a “www.google.com.”:

   _____

Esercizio 2.2

Usando il comando **nslookup** e relative opzioni

1. Identificare i nameserver del dominio “libero.it” e fornire le seguenti risposte:
   a. comando ed opzioni:_____
   b. nameserver primario: _____
   c. nameserver secondari:_____
2. identificare gli indirizzi e i nomi dei server destinati a offrire il servizio di posta elettronica per il dominio “libero.it”:
   a. comando e opzioni: _____
   b. nome e indirizzo IP del server di posta: _____
3. la risposta del punto precedente proviene da un server *authoritative*? Se così non fosse, come deve essere modificato il comando per ricevere una risposta da un server *authoritative*?
   a. comando e opzioni: _____
4. Come è possibile determinare gli alias per l’host “www.libero.it”?
   a. comando e opzioni: _____
   b. alias: _____

Esercizio 2.3

Usando il comando **nslookup** e relative opzioni, identificare per l’indirizzo IP “130.192.1.34”

1. Nome DNS corrispondente: : _____
2. Comando ed opzioni utilizzate: _____
3. Eventuali alias associati: _____

Trovare poi l’indirizzo IP associato al nome “security.polito.it”: quali considerazioni si possono fare dopo questa operazione?

Esercizio 2.4

Alcuni server applicativi fanno una doppia query al DNS per verificare l'identità dei client prima di accettarne le richieste: sono i cosiddetti *server paranoici* (in inglese: *paranoid server*). Se le risposte di lookup diretto e inverso sono coerenti, allora rispondono alla richiesta.

Seguendo questo principio, elencare i passaggi ed i comandi necessari per eseguire manualmente con il comando **nslookup** i passi effettuati da un server paranoico per verificare la corrispondenza biunivoca nome-indirizzo per i nodi con indirizzi 130.192.181.193 (corrispondente a *webfront-01.polito.it*) e 193.42.161.169 (corrispondente a *www.mondadori.it*). Quali differenze notate?

Esercizio 2.5

Si vogliono analizzare i pacchetti DNS scambiati durante la risoluzione del nome *www.google.com*.

A tal fine si segua la seguente procedura e si risponda alle domande proposte.

Aprire wireshark, predisporlo alla cattura dei pacchetti DNS (specificando l’interfaccia di rete) ed avviare la cattura dei pacchetti.

Attraverso l’applicativo **nslookup** eseguire la query richiesta.

Fermare la cattura dei pacchetti e rispondere alle seguenti domande:

1. Quali protocolli della pila TCP/IP, partendo dal livello “data-link”, sono stati impiegati?
2. Quali “porte” del protocollo di trasporto sono state usate? Il loro identificativo nelle varie prove è fisso o variabile? Il loro identificativo è, in generale, necessariamente fisso/variabile (cioè, quanto riscontrato in queste prove è una regola sempre vera)?
3. E’ possibile identificare la query richiesta al server DNS? In caso affermativo compilare i seguenti campi:
    a. Name:_____
    b. Type: _____
    c. Class_____
4. E’ possibile identificare la risposta del DNS? In caso affermativo compilare i seguenti campi per un indirizzo IP a scelta tra quelli presenti della risposta:
    a. Name: _____
    b. Type: _____
    c. Class: _____
    d. Addr: _____
5. Per identificare più precisamente il traffico DNS è possibile definire un filtro per la visualizzazione?
    - espressione del filtro:_____

Esercizio 2.6

Si vuole analizzare il throughput TCP durante il download di file con diverse dimensioni.

Usando **wget**, scaricare i file proposti (URL http://security.polito.it/~lioy/01nbe/lab2/ seguita dal nome del file) e rispondere alle domande.

1. Eseguire il download del file ‘`test1k.dat`’ e rilevare i seguenti dati:
    a. byte trasferiti: _____
    b. tempo richiesto per il download: _____
    c. throughput medio: _____
2. Eseguire il download del file ‘`test10k.dat`’ e rilevare i seguenti dati:
    a. byte trasferiti:_____
    b. tempo richiesto per il download: _____
    c. throughput medio: _____
3. Eseguire il download del file ‘`test100k.dat`’ e rilevare i seguenti dati:
    a. byte trasferiti: _____
    b. tempo richiesto per il download: _____
    c. throughput medio: _____
4. Eseguire il download del file ‘`test1m.dat`’ e rilevare i seguenti dati:
    a. byte trasferiti: _____
    b. tempo richiesto per il download: _____
    c. throughput medio: _____
5. Eseguire il download del file ‘`test10m.dat`’ e rilevare i seguenti dati:
    a. byte trasferiti: _____

b. tempo richiesto per il download: _____

c. throughput medio: _____

6. Il throughput medio varia all'aumentare delle dimensioni del file? In caso affermativo spiegare per quale ragione.

Esercizio 2.7

Si vuole analizzare (mediante l'utilizzo di un grafico) l'andamento della TCP window durante il download di risorse con dimensioni diverse.

Considerando gli stessi file proposti nell'esercizio precedente, si segua la procedura proposta qui sotto per ciascun file:

1. usando gli applicativi **wget** e **wireshark**, si proceda alla cattura del traffico TCP relativo al download di un file a scelta;
2. salvare il traffico catturato in un file esterno;
3. personalizzare la finestra di visualizzazione aggiungendo una colonna separata che contenga il valore della TCP window;
4. esportare i pacchetti catturati nel formato CSV;
5. usando **Excel**, importare i dati dal formato CSV e costruire il grafico (in ascissa impostare il tempo o il numero del pacchetto, in ordinata impostare il valore della TCP window).

Per inserire una nuova colonna in wireshark si segua la procedura:

1. dal menù di wireshark, selezionare *Edit → Preferences → Columns*;
2. aggiungere una nuova colonna specificando come `Field Type` il valore `Custom`;
3. Come `Field Name` impostare il valore `tcp.window_size`.

Infine rispondere alle seguenti domande:

1. Per ciascuna risorsa qual è la dimensione minima e massima della TCP window?
    a. dimensione minima: _____
    b. dimensione massima: _____
2. Spiegare il diverso andamento della TCP window per risorse con dimensioni diverse.

Esercizio 2.8

Usando **JPerf** si vuole confrontare il throughput UDP con quello TCP. Per poter eseguire questi test, è necessario siano presenti un processo client ed un processo server su due punti di una rete in comunicazione tra loro. JPerf può agire sia nel ruolo di client sia nel ruolo di server, configurabile tramite semplice scelta da interfaccia grafica, ma per semplicità ci limiteremo all'uso tramite client, in quanto abbiamo attivato un server all'indirizzo 130.192.1.117, porta 5001

Per il nodo in modalità client è necessario innanzitutto impostare l'indirizzo IP del server JPerf.

Durante l'esercizio saranno da modificare di volta in volta il protocollo utilizzato (UDP o TCP), la banda massima e/o il numero di "stream paralleli" (ossia numero di connessioni in uso simultaneamente).

NOTA: *In caso di accesso simultaneo di tanti client allo stesso server, è possibile che il server sia sovraccaricato dal numero di richieste contemporanee e quindi il calcolo di banda ne risulti influenzato. In generale, ed a maggior ragione durante le ore di laboratorio, un approccio "statistisco", cioè ripetere i test N volte (con N=5 o 10 ad esempio) permette di ridurre l'influenza di eventuali sovraccarichi. In questo caso, bisogna aver cura di indicare i valori medi di tutte le prove, nelle rispettive caselle sotto riportate.*

Dal client effettuare i seguenti test:

1. Impostare il protocollo TCP (con le opzioni di default ed il numero di "stream paralleli" a 1, svolgere il test e rilevare:

a. throughput: _____

b. percentuale di pacchetti persi: _____

2. Impostare il protocollo TCP (con le opzioni di default) ed il numero di “stream paralleli” a 5, svolgere il test e rilevare:

   a. throughput: _____

   b. percentuale di pacchetti persi: _____

   Quali considerazioni si possono fare in confronto ai risultati del punto precedente?

3. Impostare il protocollo UDP, la banda ad 1 Mbps ed il numero di “stream paralleli” a 1, svolgere il test e rilevare:

   a. throughput: _____

   b. percentuale di pacchetti persi: _____

4. Impostare il protocollo UDP, la banda a 5 Mbps ed il numero di “stream paralleli” a 1, svolgere il test e rilevare:

   a. throughput: _____

   b. percentuale di pacchetti persi: _____

5. Impostare il protocollo UDP, la banda a 10 Mbps ed il numero di “stream paralleli” a 1, svolgere il test e rilevare:

   a. throughput: _____

   b. percentuale di pacchetti persi: _____

6. Impostare il protocollo UDP, la banda a 1 Mbps ed il numero di “stream paralleli” a 5, svolgere il test e rilevare

   a. throughput: _____

   b. percentuale di pacchetti persi: _____

7. Impostare il protocollo UDP, la banda a 5 Mbps ed il numero di “stream paralleli” a 5, svolgere il test e rilevare

   a. throughput: _____

   b. percentuale di pacchetti persi: _____

8. Impostare il protocollo UDP, la banda a 10 Mbps ed il numero di “stream paralleli” a 5. Come varia il throughput rispetto al caso precedente? Ci sono pacchetti persi?

   a. throughput: _____

   b. percentuale di pacchetti persi: _____

9. Come varia il throughput al variare di protocolli e parametri del test? Ed il numero di pacchetti persi? Il throughput del protocollo UDP è superiore a quello del protocollo TCP? Analizzare i test precedenti e motivare le risposte.