

# Programmazione distribuita I

(01NVWOV)

AA 2014-2015, Esercitazione di laboratorio n. 4

## Cattura di sessioni HTTP

In ambiente Linux utilizzare il programma “curl” che consente di inviare richieste HTTP e ricevere le relative risposte. E' possibile salvare separatamente la parte di header inviata come risposta su HTTP dal server tramite l'opzione -D nomefile.txt . Dopo ogni test verificare il contenuto di tale file e di quello che contiene la risposta. Utilizzare anche il programma wireshark (gia' utilizzato nei precedenti laboratori) per monitorare lo scambio di dati completo sul canale TCP. E' necessario utilizzare l'interfaccia eth0.

Effettuare i seguenti tests:

1) Richiesta di una risorsa html generica

```
curl -D header.txt 'http://media.polito.it' -o response.html
```

2) Richiesta ad un motore di ricerca con parametro codificato nella stringa dell'URL

```
curl -D header.txt 'http://search.yahoo.com/search?p=hello' -o response.html
```

Notare la risposta che cerca di impostare un cookie tramite l'apposito header (quale?)

3) Richiesta ad un motore di ricerca con parametro inserito nel corpo del messaggio (metodo POST)

```
curl -D header 'http://search.yahoo.com/search' -d 'p=hello' -o response.html
```

Ripetere le precedenti richieste dall'interno del browser Chrome o Chromium, visualizzando la finestra di “Inspector” (istruzioni per l'uso: <https://developer.chrome.com/devtools>) usando la combinazione CTRL + ALT + I , e focalizzandosi sui tab “Elements” e “Network”

1) Accedere a <http://media.polito.it> digitando l'indirizzo sulla barra del browser. Verificare che e' possibile evidenziare i vari elementi della pagina, corrispondenti al codice HTML, semplicemente evidenziando il codice HTML stesso. Inoltre, utilizzando il comando “inspect element” del menu contestuale (tasto destro del mouse) su vari elementi della pagina, viene visualizzato il codice HTML corrispondente.

2) Accedere a <http://search.yahoo.com/search?p=hello> digitando l'indirizzo sulla barra del browser. Notare nel tab “Network” che e' possibile visualizzare il diagramma temporale delle varie transazioni HTTP, e il contenuto degli headers di richiesta e di risposta di ciascuna richiesta cliccando sulla richiesta stessa (anche cliccando su “View source”). Questo e' possibile anche se il protocollo utilizzato e' https.

## Creazione manuale di HTML e CSS

Per scrivere file HTML occorre usare un normale editor di testo (es. Notepad, Blocco note, vi). Il file può avere indifferentemente estensione **.htm** o **.html**.

Se le pagine sono di tipo statico e sono scritte correttamente con link relativi, allora è possibile posizionarle in una qualunque cartella del filesystem e navigarle direttamente aprendole con un qualunque browser.

Se le pagine sono di tipo dinamico (server-side) allora occorre installarle su un server web vero e proprio, cosa che verrà fatta in una prossima esercitazione.

Per tutti gli esercizi proposti, visualizzare sempre le pagine con almeno due browser diversi (es. Firefox e Chrome) e cambiando la dimensione della finestra al fine di verificare la qualità grafica della soluzione realizzata.

### Esercizio 4.1.1 Pagine personali

Sviluppare un insieme di pagine web personali, tra loro collegate mediante link:

- una home page dello studente, coi suoi dati sintetici (nome, cognome, luogo e data di nascita)
- una pagina che contenga un sintetico CV (Lista delle competenze in campo informatico)
- una pagina che contenga nomi e voti di alcuni esami superati (da realizzarsi obbligatoriamente con una **tabella**)

Tutte le pagine devono contenere nella stessa posizione:

- un menù che permetta di andare ad una qualunque delle tre (home, CV, voti)
- un footer che identifichi l'autore e ne fornisca l'indirizzo mail

Si faccia attenzione ad usare quanto più possibile una formattazione logica ed **indipendente dalle dimensioni** dello schermo su cui le pagine vengono visualizzate.

Si consiglia di creare prima le singole pagine e verificarne la funzionalità di base; solo successivamente modificarle aggiungendo il menù comune.

Si usino sempre (quando possibile) link relativi e si inserisca anche qualche link a pagine esterne pertinenti (es. al Politecnico di Torino, alla propria città di nascita).

### Esercizio 4.1.2 (creazione automatica di HTML)

Si implementi almeno una delle pagine dell'esercizio 4.1.1 tramite un sistema di composizione di pagine HTML (ad esempio, MS-Word od OpenOffice salvando il risultato in HTML). Si confronti il codice HTML come quantità e qualità con quello generato a mano nell'esercizio precedente.

### Esercizio 4.1.3 (validazione di HTML)

Si controlli la correttezza di tutte le pagine generate sfruttando il servizio di validazione del W3C:

<http://validator.w3.org>

Si correggano eventuali errori.

### Esercizio 4.2.1 (elemento grafico)

Inserire nel footer delle pagine web personali sviluppate nell'esercizio 4.1.1 il logo del Politecnico di Torino (disponibile alla URL <http://security.polito.it/img/polito.gif>) senza scaricare il corrispondente file in locale e facendo in modo che il logo resti nella parte sinistra del footer, col testo del footer scritto alla sua destra.

### Esercizio 4.2.2 (formato comune a più pagine)

Modificare le pagine web personali sviluppate nell'esercizio 4.1.1 in modo da rispettare le seguenti specifiche:

- tutte le pagine devono usare uno stesso font “senza grazia” (cioè sans-serif, quale per esempio “Arial” o “Helvetica”) per il testo, tranne il footer che deve usare un font “con grazia” (cioè serif, quale per esempio “Times”) ed italico (cioè in corsivo)

- lo sfondo di tutte le pagine deve essere nero ed il testo bianco (si consiglia di leggere la definizione del tag `<body>` nel manuale HTML 4.01, disponibile online <http://www.w3.org/TR/html401> )
- la tabella degli esami superati deve avere i nomi ed i voti degli esami scritti in rosso (se il voto è inferiore a 24) oppure in verde chiaro (se il voto è almeno pari a 24) ed i voti devono essere centrati nella loro colonna

### **Esercizio 4.3.1 (HTML strict + CSS)**

Modificare le pagine web personali sviluppate nell'esercizio 4.2.2 in modo che rispettino la specifica HTML di tipo STRICT e la formattazione sia realizzata esclusivamente tramite un CSS esterno. Le specifiche del CSS sono disponibili all'URL <http://www.w3.org/TR/CSS21>

### **Esercizio 4.3.2 (CSS layout)**

Modificare le pagine web personali sviluppate nell'esercizio 4.3.1 in modo da rispettare le seguenti specifiche:

- il layout delle pagine (header, menù, contenuto e footer) deve essere realizzato tramite CSS (aiutarsi con guide e tutorials online disponibili, per esempio qui: <http://www.cssmastery.com/Examples.zip> (vedere, per es., il contenuto di chapter 8), oppure [http://www.w3schools.com/Css/css\\_float.asp](http://www.w3schools.com/Css/css_float.asp)
- l'header deve avere un bordo rosso in rilievo
- il footer deve avere un bordo bianco nella sua parte superiore

Verificare che il layout venga mantenuto anche quando l'utente restringe la dimensione orizzontale della finestra del browser.

### **Esercizio 4.4 (Facoltativo)**

Si esplorino alcune parti del codice HTML e CSS fornito con gli esempi di un framework per la programmazione web, quale per esempio "Bootstrap", disponibile all'URL:

<http://getbootstrap.com/getting-started/>