

Università di Genova
Facoltà di Ingegneria

Livello di Applicazione in Internet

1. Introduzione

Prof. Raffaele Bolla

Ing. Matteo Repetto



Livello di Applicazione Internet: R. Bolla

Caratteristiche del corso: Docenti

- o Docente titolare
 - **Prof. Raffaele Bolla**
 - 4° Piano Edificio E (Opera Pia 13)
 - **Email: raffaele.bolla@unige.it**
 - Tel. 010 3532075
- o Docente
 - **Ing: Matteo Repetto**
 - 3° Piano Edificio E (Opera Pia 13): Lab. di Telematica
 - **Email matteo.repetto@unige.it**
 - Tel. 010 3532057

1.2

Caratteristiche del corso

- o **Valore in crediti:** 3
- o **Durata**
 - Circa 15 ore di lezione diretta (7 lezioni di 2 ore + una lezione di 1 ora)
 - 30 ore di preparazione al tirocinio da effettuarsi con il “tutor” del tirocinio.
- o **Modalità dell'esame:**
Test a risposte multiple sugli argomenti affrontati nelle ore di insegnamento frontale, orale sulla preparazione allo stage.

1.3

Obiettivi

- o Fornire un inquadramento su alcuni dei protocolli del livello di applicazione della architettura protocollare TCP/IP (la tecnologia di Internet) e mostrare alcuni esempi pratici di come le applicazioni che li utilizzano debbano essere configurate. Dare una breve introduzione alla programmazione del livello di applicazione.
- o Preparare al tirocinio

1.4

Programma del corso

1. Introduzione al livello di applicazione

- URL (*Universal Resource Locator*)

2. DNS (*Domain Name System*)

- Servizi forniti
- Struttura
- Archivi e messaggi

3. HTTP (*Hyper-Text Transfer Protocol*) e WWW (*World Wide Web*)

- Introduzione
- Tipologie di connessione
- Formati dei messaggi
- Autenticazione e *cookie*
- Cenni al *cached*

1.5

Programma del corso

4. SMTP (*Simple Mail Transfer Protocol*)

- Introduzione alla posta elettronica
- Il protocollo SMTP
- Formato dei messaggi e MIME (*Multipurpose Internet Mail Extension*)
- Protocolli di accesso alla posta
 - » POP
 - » IMAP

5. Programmazione di applicazioni che utilizzano la rete

- Introduzione all'uso dei socket TCP e UDP.

1.6

Riferimenti

- Bibliografia
 - **Cap. 2 del J. F. Kurose, K. W. Ross, Internet e Reti di Calcolatori, MacGraw-Hill, 2001**

- Sito Web

<http://www.reti.dist.unige.it/applicazioniinternet>

- Il sito contiene copia dei lucidi usati, il programma e permette di accedere alla prenotazione degli esami.

1.7

Introduzione al Livello di Applicazione TCP/IP

Arch. OSI	Arch. TCP/IP	
Applicazione	FTP, Telnet, http SMTP, SNMP, ...	Processi/ Applicazioni
Presentazione		
Sessione		
Trasporto	TCP-UDP	Host-to-Host
Rete	IP	Internet
Linea	Non Specificati	Reti private
Fisico		

1.8

Livello di Applicazione Internet: R. Bolla

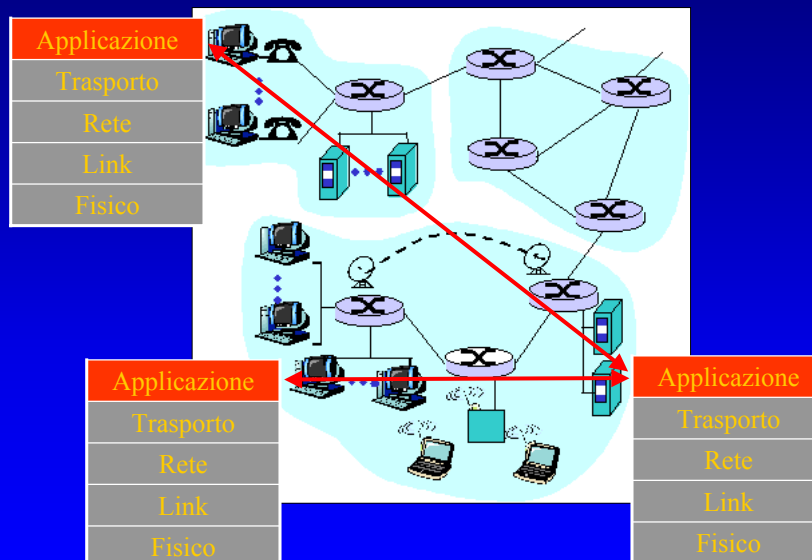
Introduzione al Livello di Applicazione

- La struttura del software/hardware che realizza le comunicazioni nella rete segue una architettura a strati di cui il strato/livello di Applicazione è quello più alto e quindi più vicino all'utente
- Livello di applicazione: protocollo e applicativi
 - Una applicazione “di rete” è composta processi applicativi distribuiti su più nodi terminali (*host*)
 - I processi su *host* diversi comunicano scambiandosi messaggi attraverso la rete, rispettando opportune regole e formati definiti tramite protocolli.

1.9

Livello di Applicazione Internet: R. Bolla

Applicazioni in comunicazione



1.10

Esempi di applicazioni Internet

- Trasferimento di File (FTP)
- Terminale remoto (Telnet)
- **WWW (http)**
- **Posta elettronica (SMTP)**
- File System distribuiti (NFS)
- **Traduzione nomi-indirizzi (DNS)**
- Applicazioni di condivisione dati (Gnutella)
- Sistemi di video-conferenza (H323, RTP)
- Sistemi per la distribuzione di video/audio broadcasting

1.11

I Protocolli del Livello di Applicazione

- Un protocollo del Livello di Applicazione è solo una parte di una Applicazione della rete (http è solo una parte del Web)
- Un protocollo del Livello di Applicazione, in generale, definisce come le entità (i processi) applicative comunicano in rete
- In particolare esso definisce:
 - La tipologia di messaggi da scambiare
 - La sintassi dei messaggi
 - La semantica dei campi
 - Quando e come un processo invia o risponde ai messaggi

1.12

Client e Server

- La comunicazione in rete tra due entità avviene spesso individuando un **Client** ed un **Server**
- Il *Client* è quell'*host* che inizia una sessione di dialogo formulando una richiesta ad un altro *host* (il *Server*), il quale risponderà
- Per molte applicazioni, uno stesso *host* utilizzerà sia il lato *Client* che il lato *Server*



1.13

Client e server

- Un *host* può avere attive sia applicazioni con funzioni di *client* che di *server* per un dato servizio
- Più in generale alcune applicazioni comprendono ambedue le funzioni.
- La distinzione fra *client* e *server*, in questo caso, serve più che altro a definire le modalità di gestione del dialogo, il *client* inizia il dialogo e fa una richiesta, il *server* risponde.

1.14

Interfacce Socket

- I processi *Client* e *Server* comunicano tra loro inviando e ricevendo messaggi tramite le loro interfacce: i *Socket*
- Un *Socket* è definibile come un API (*Application Programmers' Interface*) tra l'applicazione e la rete, si tratta infatti di un'interfaccia fornita dal Livello di Trasporto che permette di utilizzarne i servizi in maniera trasparente.
- I soli aspetti che il gestore dell'applicazione può controllare in relazione al livello di trasporto sono:
 - La scelta dello specifico protocollo di trasporto da usare (TCP o UDP)
 - Eventuale possibilità di stabilire alcuni parametri

1.15

Applicazioni

- Non tutte le applicazioni di rete forniscono servizi direttamente all'utente (umano).
- Alcune applicazioni forniscono loro volta un servizio ad altre applicazioni (ad esempio il DNS, *Domain Name System*) o integrano applicazioni locali estendendone le funzionalità alla rete (ad es. i NFS, *Network File System*)

1.16

User agent

- Lo *User Agent* (Agente d'utente) è un'interfaccia tra l'utente e le applicazioni di rete che forniscono un servizio diretto all'utente.
- I *browsers* (Mozilla, Explorer, Netscape, ...) costituiscono un chiaro esempio di *User Agent* per l'applicazione di rete Web.
- Eudora, Outlook e Messenger sono invece tre diversi *User Agent* per l'applicazione di posta elettronica.

1.17

Indirizzamento

- Affinché un processo di un *host* possa inviare messaggi al processo di un altro *host*, esso dovrà conoscere:
 - L'indirizzo IP dell'*host* di destinazione.
 - L'identificativo del processo (numero di porta), sull'*host* di destinazione, che deve ricevere il messaggio.
- Ai protocolli più comuni del Livello di Applicazione sono stati assegnati [RFC 1700] numeri di porta specifici:
 - HTTP: num. di porta 80, POP3: num. di porta 110
 - SMTP: num. di porta 25, IMAP: num. di porta 143
 - DNS: num. di porta 53

1.18

URL: *Uniform Resource Locator*

- E' un indirizzo particolare che identifica la posizione delle risorse nella rete specificando anche le modalità per accedervi.
- La sua forma è del tipo

$$\langle \text{scheme} \rangle : \langle \text{scheme-specific-part} \rangle$$

dove:

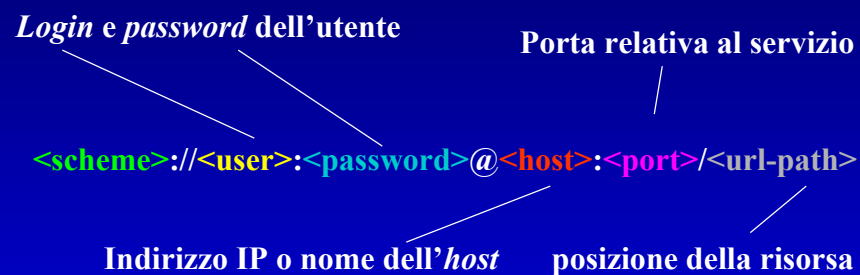
- $\langle \text{scheme} \rangle$ è, in sostanza, la modalità generale di accesso, ossia il protocollo (ftp, http, mailto ...)
- $\langle \text{scheme-specific-part} \rangle$ è nella forma:

$$\| \langle \text{user} \rangle : \langle \text{password} \rangle @ \langle \text{host} \rangle : \langle \text{port} \rangle / \langle \text{url-path} \rangle$$

- $\langle \text{user} \rangle$ e $\langle \text{password} \rangle$ sono la *login* e la *password* dell'utente
- $\langle \text{host} \rangle$ è il nome o l'IP dell'*host* e
- $\langle \text{port} \rangle$ è la porta relativa al servizio
- $\langle \text{url-path} \rangle$ è la posizione della risorsa

1.19

URL: esempio



Esempi:

<ftp://rossi@zeus.com.dist.unige.it:21/lavori>
<ftp://rossi:mr82@zeus.com.dist.unige.it:21/lavori>
<ftp://bianchi@130.251.8.191:21/lavori>
<http://zeus.com.dist.unige.it:8080/~bianchi>
<http://130.251.8.191:8080/~rossi>
<file:///C:/documenti/orario.doc>

1.20

Caratteristiche del servizio di trasporto eventualmente richieste da una applicazione

- Perdita dei dati: alcuni protocolli di applicazioni non tollerano la perdita di dati (ad es. SMTP o HTTP), quindi richiedono un livello di trasporto affidabile (TCP).
- Banda: vi sono applicazioni, soprattutto di tipo multimediale, che richiedono una quantità di banda (di picco o media) garantita nel tempo.
Le applicazioni che non hanno specifici requisiti di banda sono dette Elastiche.
- Ritardi e jitter: requisito fondamentale nelle applicazioni interattive in *Real-Time*. Esse per essere efficaci richiedono una forte limitazione dei tempi di spedizione dei dati.

1.21

Requisiti di alcune Applicazioni

Applicazione	Perdita Dati	Larghezza Banda	Ritardo
FTP	No	Elastica	No
E-mail	No	Elastica	No
Web	No	Elastica (pochi Kbit/s)	No
Audio-Video Real-Time	Tollerabile	Audio: pochi Kb/s - 1Mb/s Video: 10 Kb/s - 5 Mb/s	S ^{cc} , centinaia di ms
Audio-Video in memoria	Scarsa-tollerabile	Audio: pochi Kb/s - 1Mb/s Video: 10 Kb/s - 5 Mb/s	S ^{cc} , pochi sec
Giochi interattivi	Scarsa-tollerabile	Pochi Kb/s-10Kb/s	S ^{cc} , centinaia di ms
Appl. Finanziarie	No	Elastica	S ^{cc} e No

1.22