Seminari Eucip, Esercizio e Supporto di Sistemi Informativi

Servizi di Rete

Alessandro Farinelli

Dipartimento di Informtica e Sistemistica
Università di Roma “La Sapienza”

Sicurezza su Rete

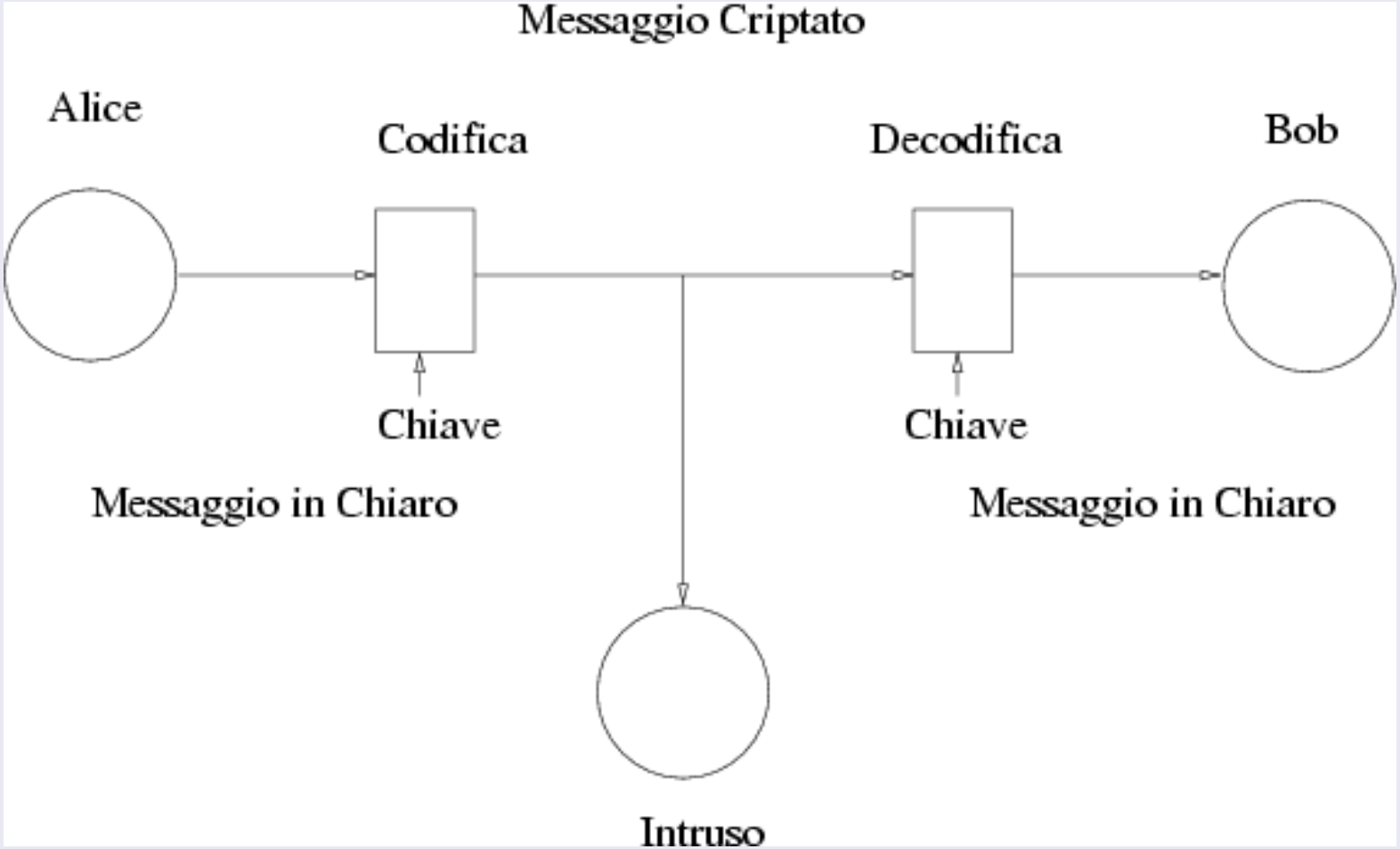
La rete è inerentemente insicura
Lo sforzo per ottene/manipolare dati bilanciato con il valore dei dati stessi

Caratteristiche della Sicurezza

- Riservatezza e Autenticità: Sniffing e Spoofing
- Disponibilità: Denial-of-service, worm
- Integrità: Man-in-the-middle

Crittografia

scopo: garantire la riservatezza



Crittografia Simmetrice e a Chiave pubblica

Principi di Base

- Simmetrica: due utenti usano la stessa chiave
  - Chiave deve essere segreta
  - Deve essere difficle da identificare
  - Sicurezza $\Rightarrow$ chiavi monouso
  - Come le distribuisco (Diffie-Hellman-Merkle) ?
- Chiave pubblica (RSA)
  - Due Chiavi per ciascun utente: Pubblica e Privata
  - La chiave Pubblica **deve** essere nota a tutti.
  - A manda a B un messaggio codificato con $PubKey_b$
  - B decodifica messaggio utilizzando la sua chiave segreta
  - Sicurezza garantita dalla non reversibilitá algoritmo di criptazione (chiave 512 a 4096 bit)

Firma digitale

Principi di Base

Bob vuole essere sicuro che è veramente Alice che ha mandato il messaggio

- RSA utilizzato per firma
- Alice crea un **impronta** del messaggio (Hash)
- Alice cripta l'impronta con chiave privata
- Bob utilizza la chiave pubblica di Alice con RSA per ottenere l'impronta in chiaro
- Bob dal messaggio genera l'impronta (Hash)
- Se le due impronte sono uguali solo Alice può aver mandato il messaggio
- La chiave pubblica deve essere assicurata da una Authority

Autenticazione utenti

X.509

- X.509 Standard per autenticazione utenti
- basato su certificati
- chiave pubblica certificata da authority

Secure Socket Layer (SSL)

- Standard di sicurezza livello trasporto
- Valido per tutte applicazioni che utilizzano TCP
- Autenticazione, integrità confidenzialità

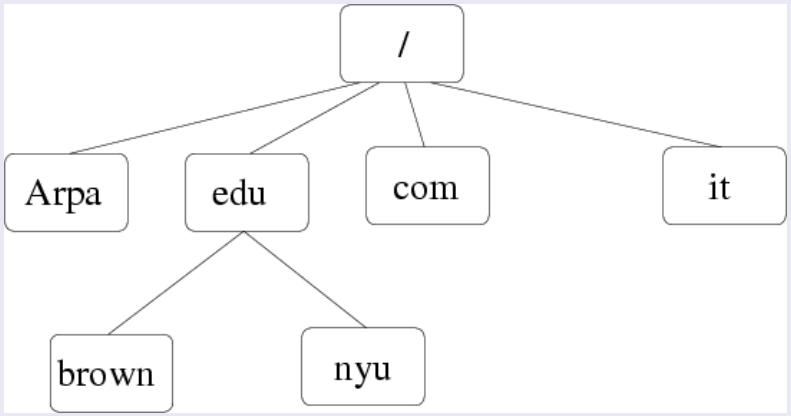
Funzionalità DNS

perchè serve un DNS ?

- Nodi (terminali) su internet individuati da IP a 32 bit
- In pratica più comodo utilizzare indirizzi simbolici
- 216.239.59.103 $\Leftrightarrow$ www.google.com
- DNS = Domain Name System
  - Database distribuito che traduce indirizzi IP in nomi simbolici
  - Protocollo (protocollo DNS) usato dagli host per richiedere traduzioni di nomi simbolici in indrizzi IP

Gerarchia dei Nomi

Gerarchia



Caratteristiche Gerarchia

- Gerarchia va da Destra a Sinistra
- I Top Level Domain (es .com) assegnati da autorità apposita IANA[1]
- TLD non sempre rel. a nazione geografica (Es .com)

[1] International Assigned Number Authority

DNS modalità funzionamento

Gerarchia

- Gerarchia Nomi $\Rightarrow$ Gerarchia dei server dei nomi
- Traduzione basata su interazione client-server
- host richiede indirizzo a server DNS **locale**
- se richiesta non soddisfacibile server DNS richiede a DNS **radice** indirizzo del nameserver del dominio
- protocollo basato su UDP
- **Nota**: relazione IP Nomi $\Rightarrow$ uno a molti

Applicazioni su rete

Caratteristiche generiche

- Applicazioni distribuite
- Coinvolgono scambio di messaggi tra processi dislocati su macchine differenti
- WWW, posta elettronica, trasferimento file.
- In genere seguono il modello client/server

Il World Wide Web

Applicazione per scambiare documenti

- Formato standard documenti (HTML)
- Processi Server (Web Server)
- Processi Client (Browser)
- Protocollo (HTTP)

Il protocollo HTTP

Caratteristiche principali

- Gestisce scambio di oggetti (pagine Web)
- Client apre connessione ed invia richieste di oggetti
- Server aspetta richieste e le gestisce
- Basato su TCP/IP
- Messaggi testuali basati su pochi comandi (GET, POST, HEAD)
- Connessione non persistente (1.0) o persistente (1.1)
- Autenticazione utenti basata su username e password o su cookie (anche entrambe possibili)
- Web caching

Pagine Web

Cosa sono

- Insieme di oggetti, testo e riferimenti ad altre pagine Web
- Sono visualizzate dai browser
- Specificate in HTML (Hyper Text Markup Language)
- Indirizzate da un URL (Universal Resource Locator)

URL

Schema generale di un URL

- Dove si trova la risorsa (www.dis.uniroma1.it:80)
- Nome risorsa (index.html)
- protocollo da usare per ottenere la risorsa (http://)
- http://www.dis.uniroma1.it/index.html = http://dis.uniroma1.it
- HTTP, file, FTP
- un URL $\Leftrightarrow$ un file $\Rightarrow$ possibile sovraccarico server

HTML

Il linguaggio HTML

- HTML usa annotazioni (MarkUp) per formattare il testo (es bold)
- Browser interpreta HTML e visualizza la pagina
- pagina html = <html> <head> testata pagina </head> <body> corpo pagina </body>
- Form $\Rightarrow$ trasmettono dati al server
- metodo POST e GET

Non solo HTML...

Programmazione server e client side

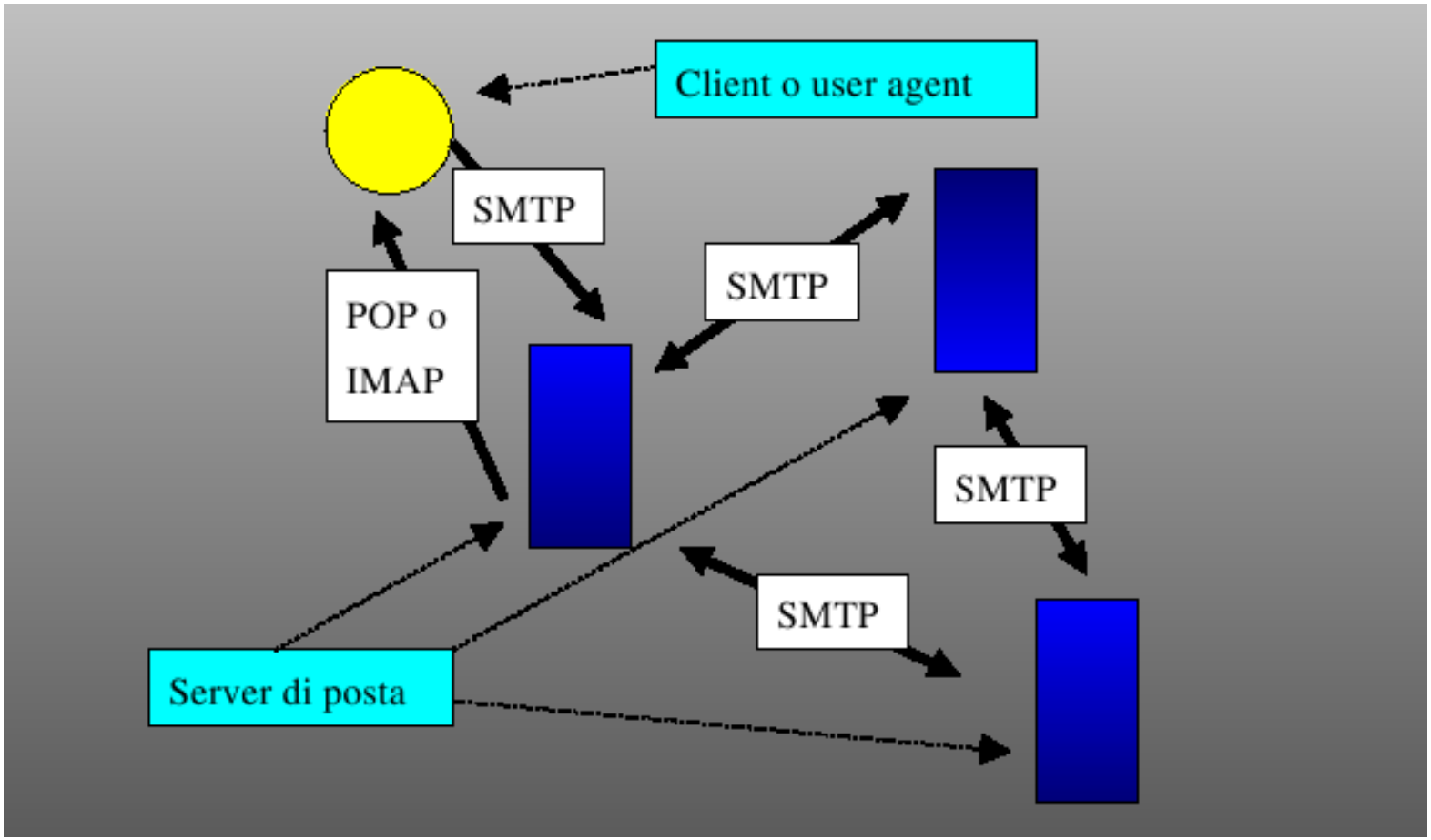
- Scambio pagine web processo **statico**
- Sia server che client possono anche eseguire programmi durante l'interazione
- Lato client:
  - script interpretati direttamente dal browser (JavaScript)
  - plug-in esterni attivati da tag HTML (es <OBJECT> <APPLET>)
- Lato server:
  - JSP, PHP, ASP attivati da tag specifici HTML
  - CGI, servlet totalmente esterni alla pagina Web

Common Gateway Interface

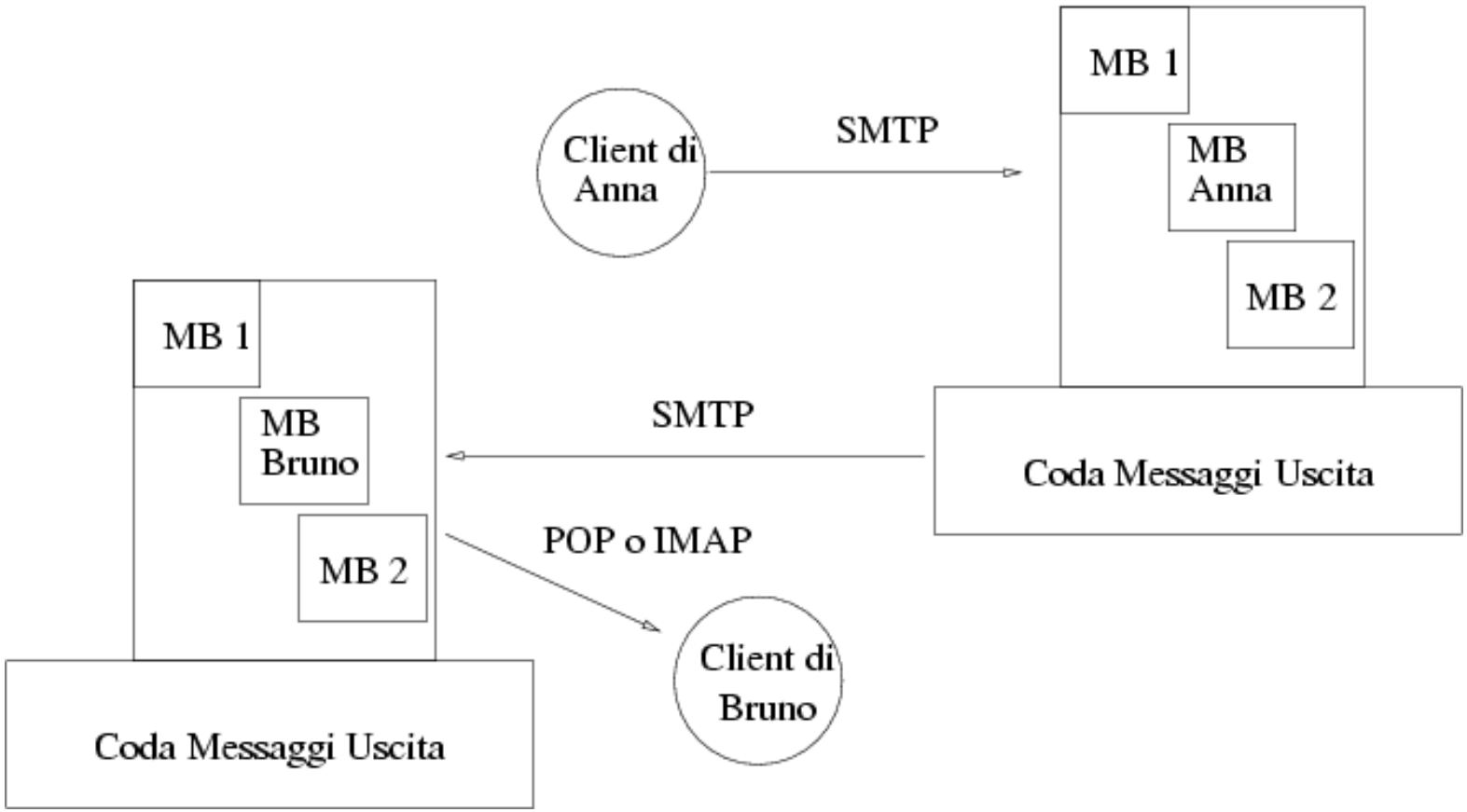
CGI

- Interfaccia con programmi esterni con il server Web
- Il Web server rileva l'invocazione di un programma CGI e lo esegue
- I programmi CGI possono essere scritti in un qualsiasi linguaggio eseguibile dal server
- In genere si usano script Python, Perl, comandi shell Unix.

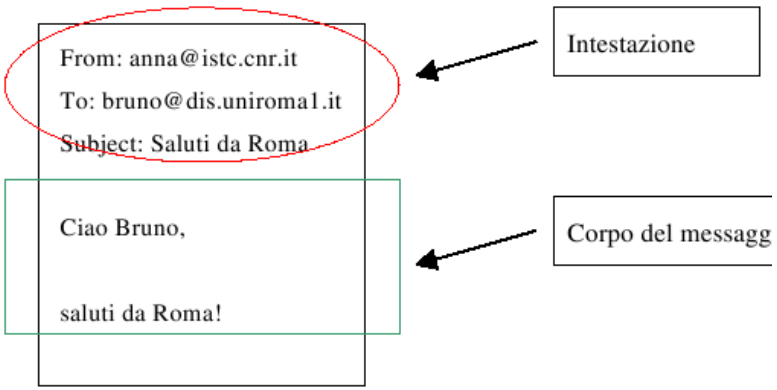
Componenti e Protocolli

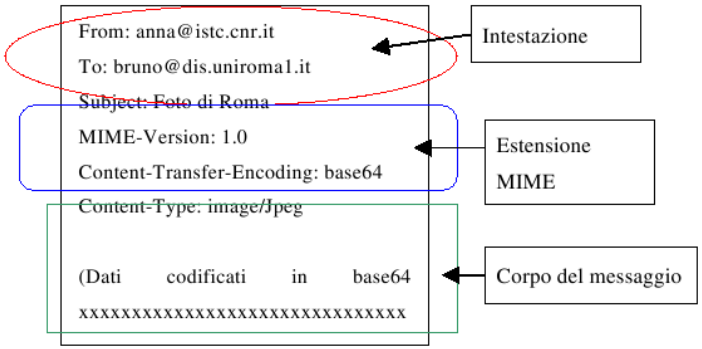


Transazione esempio



Formato messaggi posta





SMTP

<handshacking>
S1 220 webmail.dis.uniroma1.it ESMTP Postfix on UnitedLinux (i
C2 HELO smtp.cs.cmu.edu
S3 250 webmail.dis.uniroma1.it
C4 MAIL FROM: anna@cs.cmu.edu
S5 250 Ok
C6 RCPT TO: bruno@dis.uniroma1.it
S7 250 Ok
<handshacking>
<trasferimento messaggio>
C8 DATA
S9 354 End data with <CR><LF>.<CR><LF>
C10 Ciao Bruno,
C11
C12 saluti da Pittsburgh!
C13 .
S14 250 Ok: queued as 184DE8DEB1
<trasferimento messaggio>
<chiusura trasmissione>

concetti base

- client/server
- client= client posta o server posta, server = server posta
- Hand-Shacking, Trasmissione, Chiusura
- Basato su TCP/IP
- Richieste = comandi Testuali
- Risposte = codici di stato

POP3

Post Office Protocol

- Protocollo per scaricare posta
- client di posta = client, server di posta = server
- basato su TCP e comandi testuali
- Tre fasi: Autenticazione, Trasferimento, Aggiornamento
- Autenticazione basata su username e passwd
- Trasferimento client manda comandio per trasferire posta
- Aggiornamento server aggiorna la mailbox in corrispondenza dello scambio precedente

IMAP

Internet Mail Access Protocol

- Feature di base come POP (TCP, comandi testuali, scaricamento posta etc.)
- Molto più complesso
- Permette gestione di cartelle sul server

Posta via Web

Introdotta di recente

- Google, Yahoo, hotmail, etc.
- Un server Web funge da client di posta
- Utente utilizza un browser che colloquia con il sever Web tramite HTTP
- Molto utile per utenti che non accedono la posta sempre dallo stesso PC

Applicazioni Multimediali in Rete

Tipiche applicazioni multimediali

- Ascolto brani musicali
- Visione filmati
- Voice over IP (VoIP)

Requisiti e tecniche per le Applicazioni Multimediali su Rete

Impatto sulla rete

- Flussi dati molto consistenti
- Congestione $\Rightarrow$ servizio scadente
- In genere utilizzo UDP
- Molto spesso utilizzano i concetti di **Streaming** e **Buffering**
- VoIP delay è una problematica molto importante
- Largo utilizzo compressione $\Rightarrow$ buone capacità di calcolo o processori dedicati
- Grosse quantità di dati $\Rightarrow$ buone capacità immagazzinamento $\Rightarrow$ dischi grandi e veloci, molta RAM