PARTE 10

SERVIZI DI RETE
“Storici”

Servizi di rete Storici

- **Trasferimento file – FTP ***
- **Accesso remoto a computer – TELNET ***
- **Posta elettronica – SMTP**
- **Accesso a dati di dispositivi in rete – SNMP**
- *** Tendono a non essere più utilizzati perché INSICURI**

Modulo 1: Trasferimento remoto di file (FTP)

Trasferimento file

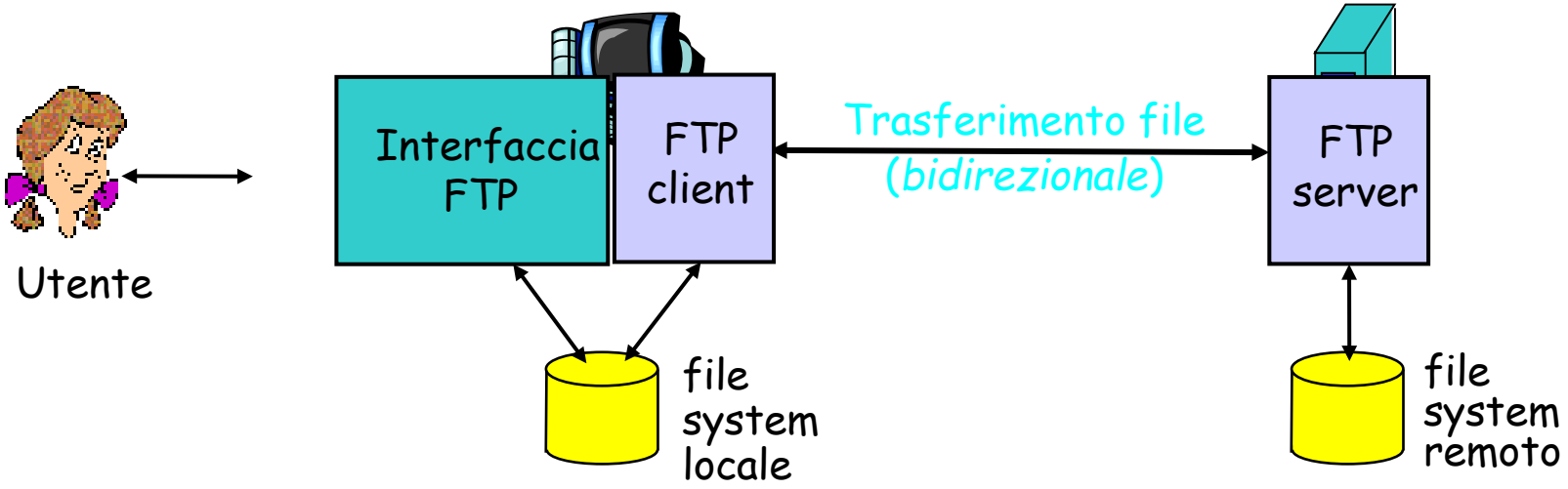
- **Il trasferimento di file da un host ad un altro è una delle applicazioni (dirette o indirette) più importanti che operano su Internet**
  - Il WWW trasporta file
  - La posta elettronica è in pratica un trasferimento file
  - Il peer-to-peer consente di scaricare file (molti MP3 e video …)
  - ecc.

- **In questa sezione si analizza il protocollo “storico” per il trasferimento file: FTP**

File Transfer Protocol

- **Descrizione del protocollo per il trasferimento di file da/verso un host remoto:**
  - FTP [RFC 959]
  - (<1000 = “storico”)
- **Paradigma client/server**
  - client: inizia il trasferimento (sia verso/da remoto)
  - server: host remoto
- **Protocollo di trasporto utilizzato: TCP**
  - Perché?

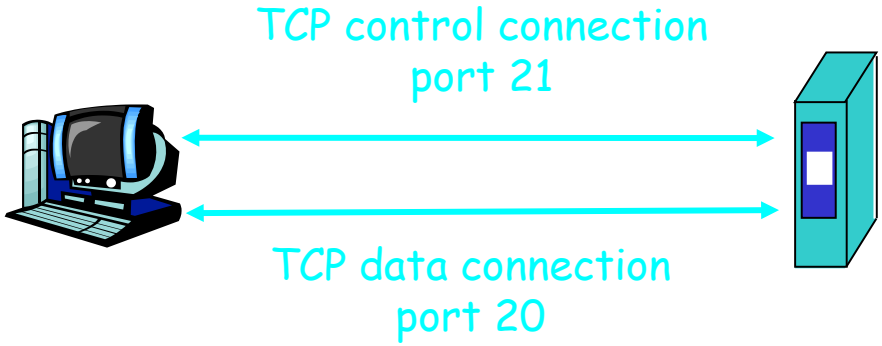
File Transfer Protocol



- Elementi fondamentali:
  - **FTP client**
  - **FTP server**
  - **Interfaccia FTP**
  - **File system locale e file system remoto**
- *Well known ports* dell'FTP del server:
  - ***porta 21:* controllo**
  - ***porta 20:* dati**

File Transfer Protocol

- Il client FTP richiede una connessione TCP sul server FTP sulla *well known port* 21
- Vengono aperte **due** connessioni TCP parallele:
  - **connessione per controllo: scambio dei comandi (put, get), delle risposte tra client e server**
  - **connessione per dati: il client inizia la connessione, ma il trasferimento file può avvenire in modo bidirezionale (da/verso il file system locale). Si apre una connessione dati per ogni file**



FTP è un protocollo stateful.
Il server FTP mantiene lo "stato":
- **directory corrente**
- **autenticazione precedente**

Interfaccia utente

- **Per molti anni gli utenti hanno avuto a disposizione solo un'interfaccia a linea di comando per utilizzare FTP**

- **Attualmente, vi sono molte interfacce grafiche disponibili, ma i comandi sono sempre quelli definiti dal FTP**

FTP

- **Quando un utente attiva**
  - ftp [indirizzo remoto]
- **un programma applicativo sulla sua macchina diventa client e cerca di stabilire una connessione TCP al server identificato con [indirizzo remoto] può essere:**
  - Indirizzo IP dell'host a cui connettersi
  - Hostname dell'host
  - Poiché accetta anche gli indirizzi IP, il comando ftp può essere usato anche se il DNS non funziona
- **Esempi**
  - ftp sturdust.ing.unimore.it
  - ftp 160.80.120.85

Comandi e risposte FTP (linea di comando)

Alcuni comandi:

- inviati come testo ASCII sulla connessione di controllo
- più di 50 comandi
- **`USER`** ***`username`***
- **`PASS`** ***`password`***
- **`LIST`** restituisce la lista dei file nella directory corrente (inviata su una nuova connessione dati)
- **`RETR filename`**: prendi (get) il file dalla directory corrente
- **`STOR filename`**: memorizza (put) il file nella directory corrente dell'host remoto

Alcuni codici di risposta:

(codice dello stato e frase come in HTTP)

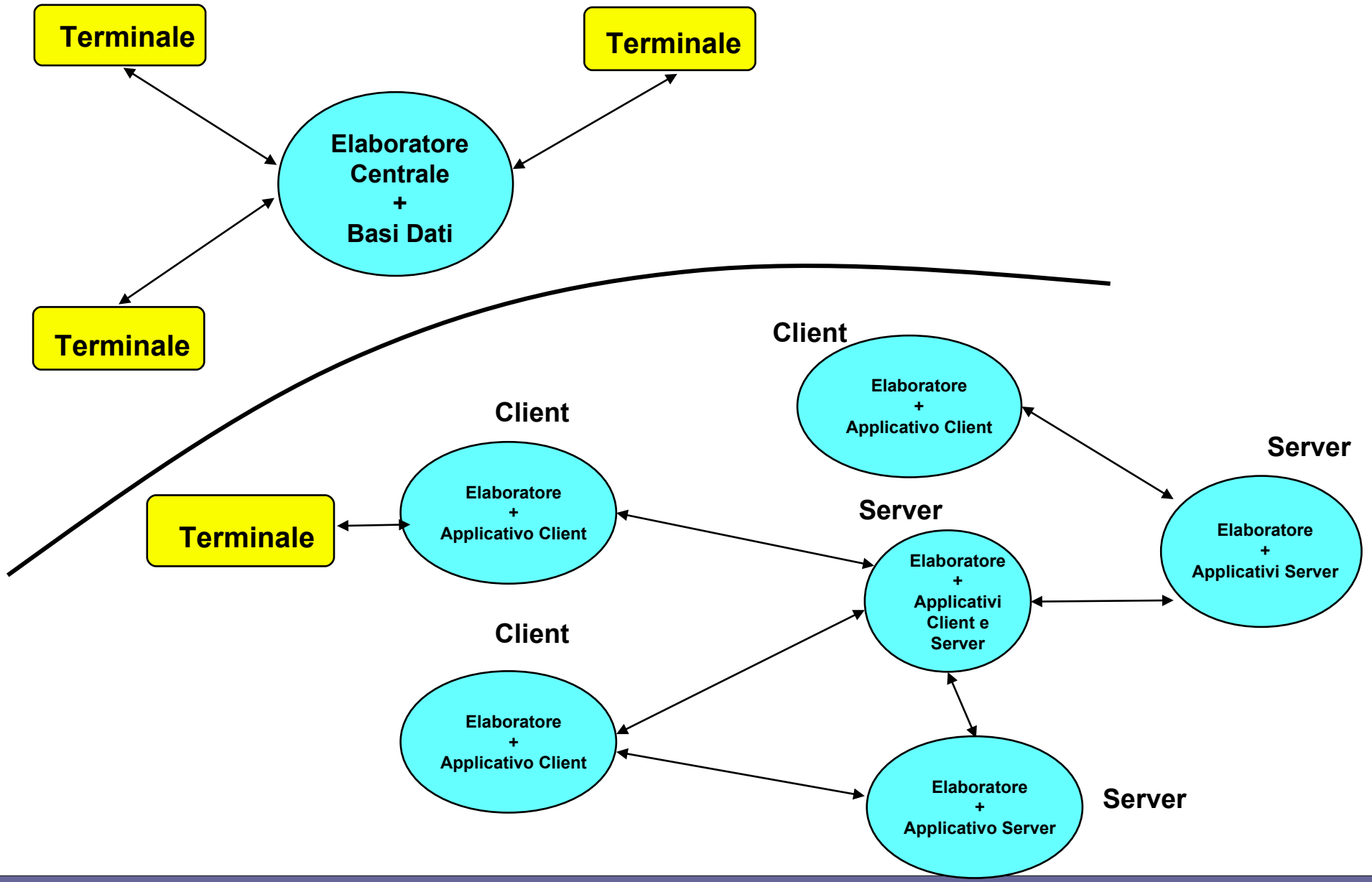
- **`331 Username OK, password required`**
- **`125 data connection already open; transfer starting`**
- **`425 Can't open data connection`**
- **`452 Error writing file`**

Perché della nota iniziale?

- **[I protocolli FTP e TELNET tendono a non essere più utilizzati perché ritenuti INSICURI]**
- **A meno che non sia un server FTP “pubblico” (anonymous), per poter stabilire una connessione con una macchina remota bisogna AUTENTICARSI con Login+Password, quindi FTP sembra sicuro, sembra …**
- **PROBLEMA:Tutta l’informazione “viaggia in chiaro”:**
  - Sia i dati di autenticazione
  - Sia i file trasmessi
- **→ E’ facile intercettare il tutto**

Modulo 2: Collegamento a computer remoto (TELNET)

Motivazioni del collegamento remoto



TELNET o telnet ?

- **TELNET è un protocollo di livello applicativo, descritto nell' RFC 854 che fornisce una modalità di comunicazione remota:**
  - Generale
  - Bi-direzionale
  - Basata su flussi di byte
- **telnet è una applicazione che utilizza il protocollo TELNET operante su TCP**

- **E' possibile realizzare altre applicazioni utilizzando il protocollo applicativo TELNET**

Definizioni di TELNET

“A terminal emulation program for TCP/IP networks such as the Internet.

The telnet program runs on your computer and connects your PC to a server on the network.

You can then enter commands through the telnet program and they will be executed as if you were entering them directly on the server console.

This enables you to control the server and communicate with other servers on the network.

To start a telnet session, you must log in to a server by entering a valid username and password.

Telnet is was **a common way to remotely control Web servers.”**

Cosa fornisce il protocollo TELNET

- **Connessione TCP e trasporto di dati e comandi di controllo sulla stessa connessione (a differenza di FTP)**
- **Concetto di Network Virtual Terminal per gestire l'eterogeneità**
- **Opzioni negoziali**

Login remoto

- **Tramite la funzionalità di login remoto gli utenti hanno accesso a tutti i comandi disponibili sul sistema remoto**
- **Quando un utente attiva**
  - telnet [indirizzo remoto]
- **un programma applicativo sulla sua macchina diventa client e cerca di stabilire una connessione TCP al server identificato con [indirizzo remoto] può essere:**
  - Indirizzo IP dell'host a cui connettersi
  - Hostname dell'host
- **Esempi**
  - telnet sparc.scienze.unimore.it
  - telnet 155.30.10.88

Login remoto

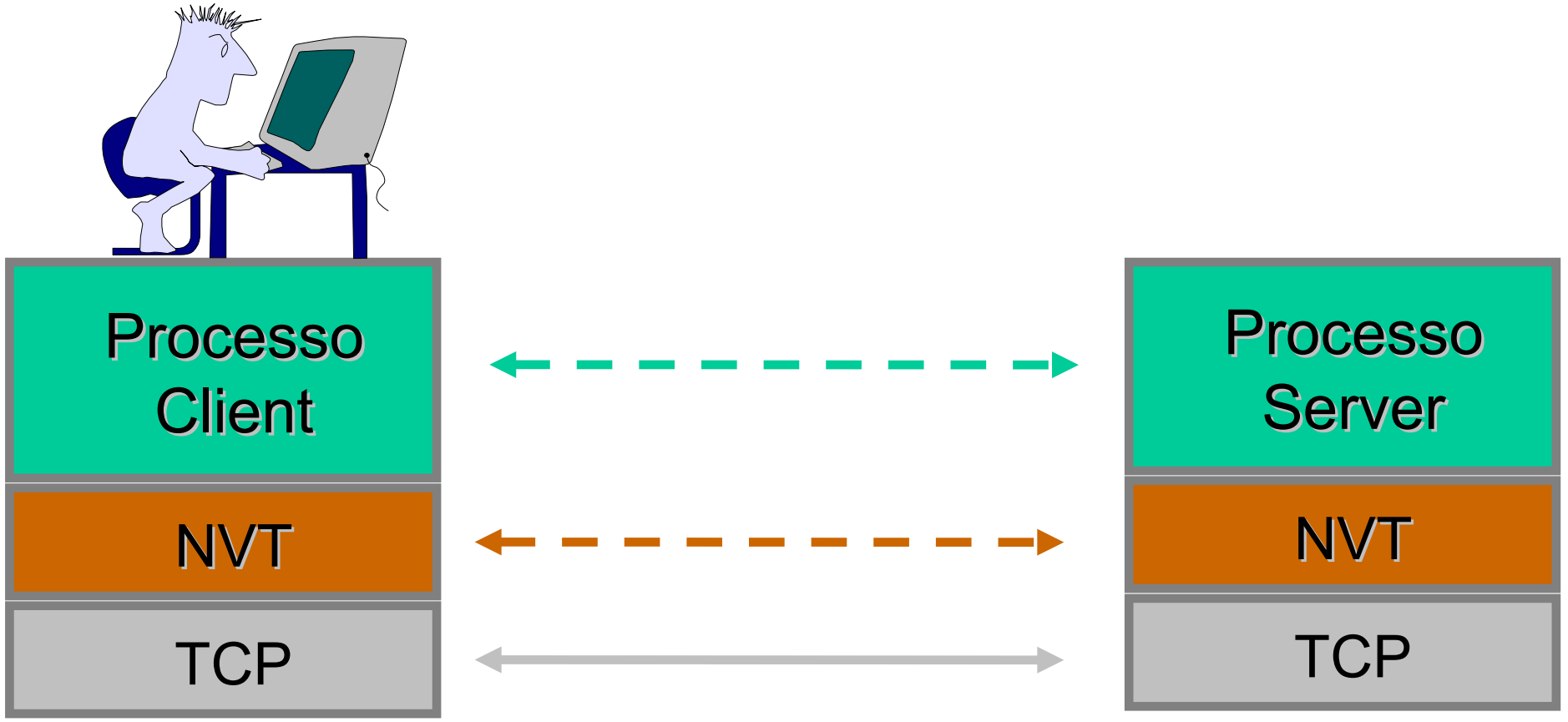
- **Per poter funzionare, il server deve accettare una connessione TCP dal client e poi ritrasmettere i dati dalla connessione TCP al sistema operativo locale**
- **Una volta stabilita la connessione TCP,**
  - il client accetta le sequenze dei tasti dalla tastiera dell'utente e le invia al server
  - il server accetta le sequenze dei tasti dalla tastiera dell'utente e le passa al sistema operativo come se fossero state digitate su un terminale direttamente collegato alla macchina remota
  - il telnet trasferisce anche l'output dalla macchina remota allo schermo dell'utente, ovvero consente che i caratteri rinviati dal server al client vengano visualizzati sullo schermo dell'utente (effetto "echo")

Problema: Eterogeneità

- **Per far sì che TELNET possa funzionare su piattaforme differenti, deve gestire i dettagli di vari sistemi operativi e computer eterogenei**
- **Esempi:**
  - Alcuni sistemi richiedono che le righe di testo terminino con il carattere ASCII CR (Carriage Return, Ritorno Carrello), altri con il carattere LF (Line Feed, Avanzamento Linea), altri ancora con la sequenza dei due caratteri CR-LF
  - Inoltre, molti sistemi interattivi forniscono all'utente un modo diverso per interrompere un programma in esecuzione mediante un tasto o combinazione di tasti (es., ^D, ^C)

Soluzione: Network Virtual Terminal

- Rappresentazione “virtuale” di un terminale
- Dotato di un linguaggio standard di comunicazione per le funzioni di controllo terminale



Network Virtual Terminal

- **Il software del client traduce le sequenze dei tasti e dei comandi dal terminale dell'utente nel formato NVT e le invia al server**
- **Il software del server traduce i dati e i comandi in ingresso dal formato NVT nel formato richiesto dal sistema remoto**

Network Virtual Terminal

- **Per restituire i dati, il server remoto traduce dal formato della macchina remota a NVT e il client locale traduce da NVT al formato della macchina locale**
  - NVT usa la rappresentazione standard US-ASCII a 7 bit per i dati e riserva i byte in cui il bit più significativo è impostato a 1 per le sequenze dei comandi
  - L'insieme di caratteri US-ASCII include 95 caratteri stampabili (ad esempio, lettere, cifre e segni di punteggiatura) e 33 codici di controllo

Network Virtual Terminal

- **Il NVT di TELNET gestisce le funzioni di controllo definendo il modo in cui vengono passate dal client al server**
- **Poiché la maggior parte delle tastiere non fornisce tasti extra per i comandi, si usa un carattere ASCII per una funzione di controllo in modo che quando l'utente preme il tasto, il sistema operativo esegue l'operazione appropriata invece di accettare il carattere come input**

Network Virtual Terminal

- **I progettisti di NVT hanno scelto di tenere i comandi separati dal normale insieme dei caratteri ASCII per due motivi:**
  - maggiore flessibilità; può trasferire tutte le sequenze di caratteri ASCII tra il client e il server, e tutte le funzioni di controllo
  - separando i segnali dai normali dati, NTV consente al client di specificarli in modo non ambiguo: non c'è dubbio se un carattere di input deve essere trattato come dato o come funzione di controllo

Comandi di controllo

- **TELNET fornisce un insieme di comandi di controllo tipicamente supportate dai server**
- **Interrupt Process (IP)**
  - Sospende/interrompe processo
- **Abort Output (AO)**
  - Processo può terminare, senza inviare ulteriori dati verso il terminale dell'utente

Comandi di controllo

- **TELNET fornisce un insieme di comandi di controllo tipicamente supportate dai server**
- **Are You There (AYT)**
  - Controlla se il sistema è ancora attivo e risponde
- **Erase Character (EC)**
  - Elimina l'ultimo carattere inviato
- **Erase Line (EL)**
  - Elimina tutto l'input della linea corrente

Network Virtual Terminal

- **Per passare i comandi di controllo attraverso la connessione TCP, TELNET le codifica usando una sequenza di ESC, che usa un byte riservato per indicare che segue un byte di codice di controllo**
- **In TELNET, il byte riservato che inizia una sequenza di ESC è noto come il byte IAC (Interpret As Command)**
- **Per esempio, per richiedere che il server interrompa il programma in esecuzione, il client deve inviare la sequenza IAC IP di 2 byte (ovvero 255 seguito da 244)**

Opzioni negoziali

- **Tutti i terminali supportano un insieme minimo di funzionalità, tuttavia qualche terminale ha funzionalità maggiori dell'insieme minimo**
- **I due host possono negoziare un insieme di opzioni mutamente accettabili. Es.**
  - Line mode vs. character mode
  - Echo modes
  - Codifica set di caratteri (EBCDIC vs. ASCII)

Opzioni negoziali

- **Il protocollo per richiedere e negoziare funzionalità opzionali include una serie di regole ben definite**
- **L'insieme di opzioni non fa parte del protocollo TELNET, per cui nuove funzionalità di terminale possono essere aggiunte senza modificare il protocollo TELNET**
- **La serie di opzioni TELNET è vasta: alcune riguardano le funzioni nei modi più importanti, mentre altre trattano dettagli minori**

esempio

- **Il protocollo originale è stato progettato per un ambiente semi-duplex in cui era necessario comunicare all'altro lato di "andare avanti" prima che venissero inviati altri dati.**
  - Un'opzione controlla se TELNET funziona in modalità semi- o full-duplex
  - Un'altra consente al server su una macchina remota di stabilire il tipo di terminale dell'utente, che è importante per l'esecuzione remota di software che crea le sequenze di posizionamento del cursore (ad esempio, un editor a schermo intero in esecuzione su una macchina remota)

Situazione odierna

- **Non tutti i server accettano connessioni telnet, anzi attualmente si cerca di impedire tali connessioni, perché:**
  - sono “connessioni in chiaro” (come FTP)
  - consentono un controllo da remoto dell’host
- **Si raccomanda l’utilizzo di connessioni cifrate: SSH**
  - SSH Secure Shell Client
  - OpenSSH

Modulo 3:
Posta elettronica

Posta elettronica

- **E’ una delle “killer” application di Internet**

- **Sebbene l’idea di comunicare via computer addirittura anticipa Internet**
  - Anni ’60 e scambio informazioni tra utenti dello stesso computer
  - Poi, comunicazione tra computer diversi
  - Una delle prime applicazioni in ARPANET (citata già nel 1969)

Prima email

- Prima e-mail inviata tra due computer (BBN- TENEXA e BBN-TENEXB, della Digital con sistema operativo TENEX), affiancati ma collegati via ARPAnet

- Lo stile moderno vuole **email** invece di **e-mail**

Storia del simbolo @

- **Sicuramente di origine commerciale:**
  - unità di misura (amphora – mercante veneziano)
  - al tasso di
  - at the price of come “prezzo per ciascuno”
  - …
- **Due ipotesi:**
  - deriva da “à”
  - deriva da “at”
- **Raymond Tomlinson (1971) fu il primo a usare @ con il senso moderno**
- **Utente@DominioComputer**

@ : non solo “chiocciola”

- **La hall of shame dell'abuso linguistico nei confronti di un simbolo indifeso:**
  - proboscide d'elefante
  - orecchio d'elefante
  - coda di scimmia
  - zampa di gatto
  - coda di gatto
  - strudel
  - filetto d'aringa
  - kanelbulle (dolce svedese)
  - ...

Indirizzi email

- **Local_username@Domain**
- **Formato di indirizzi email → RFC 2822, RFC 3696**
  - Max 64 caratteri nella parte Username
    - La componente locale dovrebbe essere trattata come “case sensitive”, anche se molti mail server, per garantire maggiore compatibilità, scoraggiano o escludono questa possibilità ed escludono anche molti dei caratteri non alfabetici inclusi come possibili negli RFC
  - Max 255 caratteri nella parte Domain
    - Solo caratteri, cifre, -, .
    - Questa parte non è, né è trattata come “case sensitive”

Indirizzi email

- **Gli indirizzi email non devono essere considerati affidabili perché i protocolli e gli agenti tipicamente non richiedono meccanismi di autenticazione**

Gestione posta elettronica

- **La posta elettronica è un sistema alquanto complesso che include diversi componenti**
- **Elementi fondamentali**
  - Mail Transfer Agent: il mail server per la gestione e il trasferimento della posta
  - SMTP: il protocollo per il trasferimento della posta
  - DNS: resource record MX
  - Mail User Agent: il processo per la gestione della posta lato utente
- **Elementi aggiuntivi**
  - Mail Delivery Agent (MDA)
  - Mail Submission Agent (MSA)

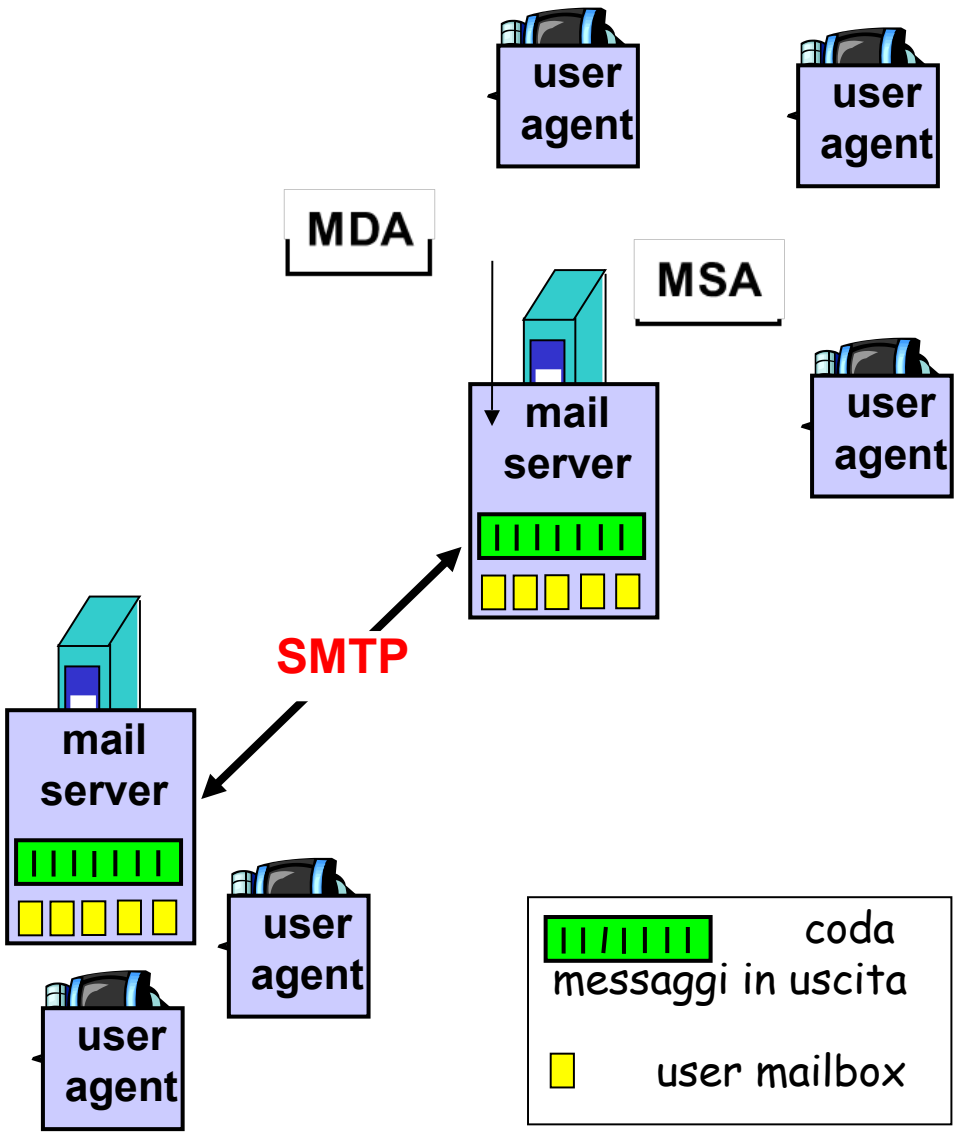
Posta elettronica

Componenti principali:

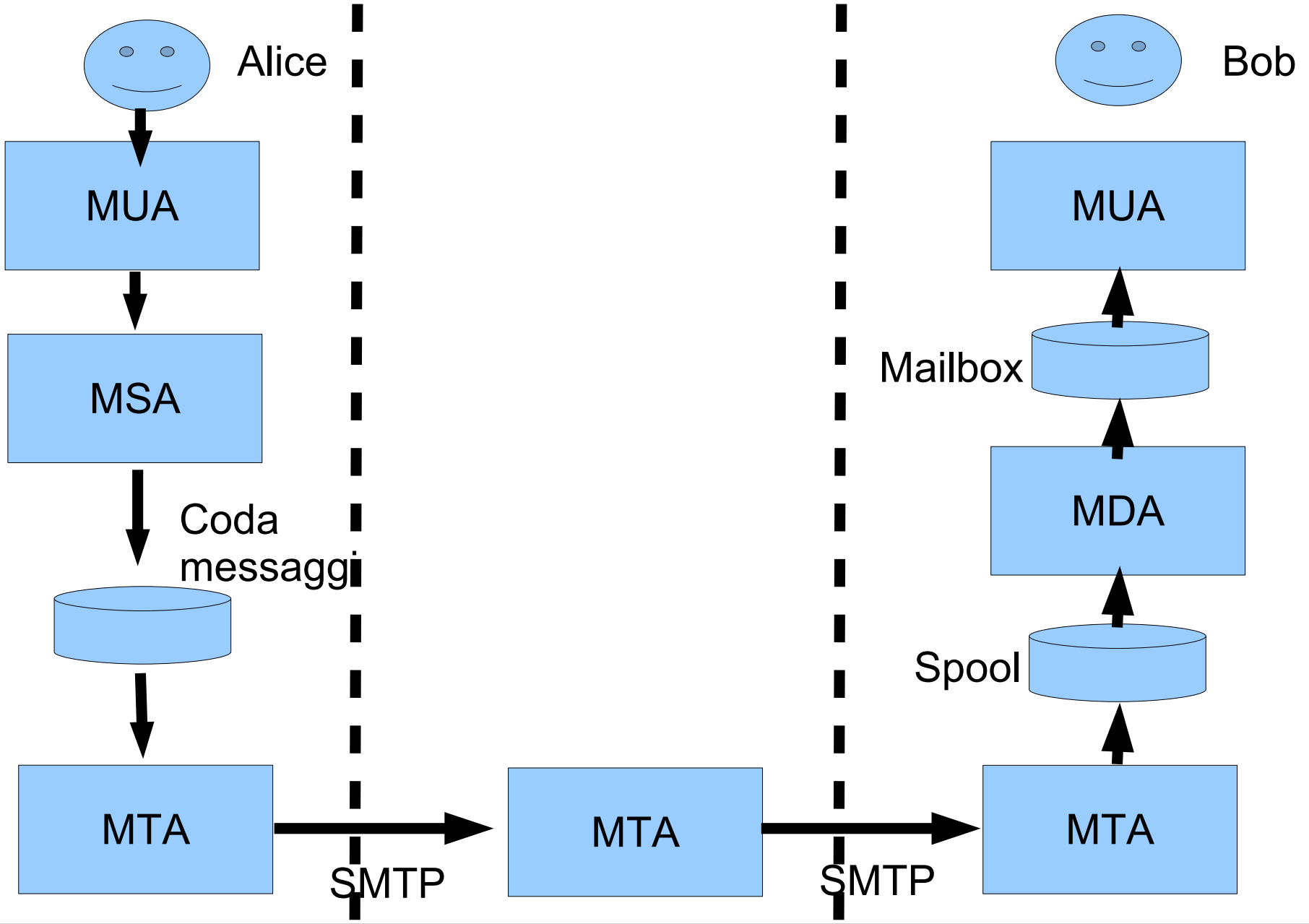
- **Mail user agent (MUA)**
- **Mail server (MTA)**
- **Simple Mail Transfer Protocol (SMTP)**

Componenti aggiuntivi:

- **Mail submission agent (MSA)**
- **Mail delivery agent (MDA)**



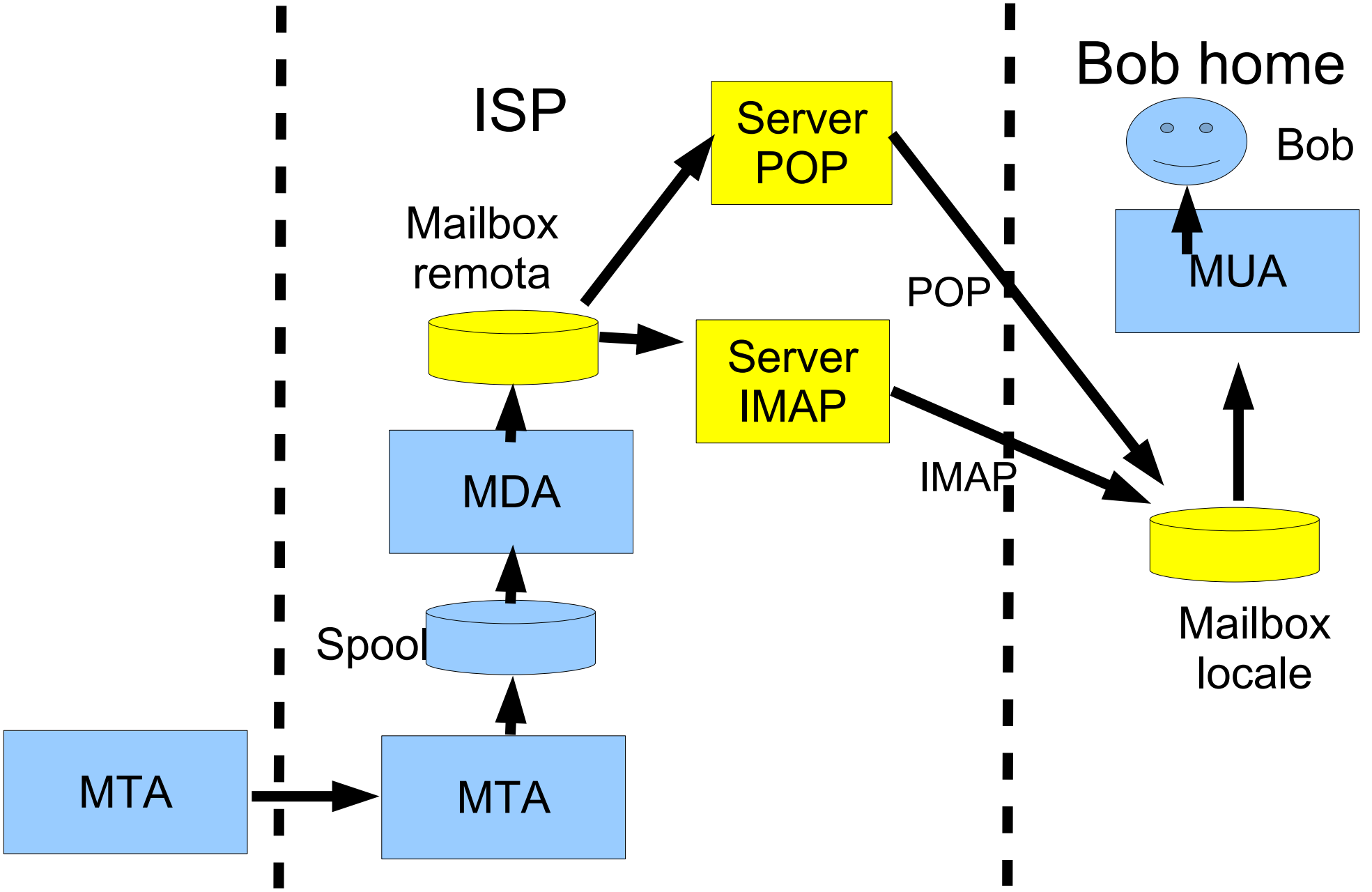
Un sistema di posta (scenario “tradizionale”)



Spiegazione dei componenti

- **MUA= Mail User Agent**
  - Programma usato per comporre e spedire posta
  - E.g., Mozilla Thunderbird, Outlook, Eudora, Evolution...
  - Può essere anche un'applicazione Web (Webmail)
- **MSA= Message Submit Agent**
  - Serve per inoltrare il messaggio al sistema di posta
  - E.g., Sendmail, molti MUA comprendono anche un MSA
- **MTA= Mail Transfer Agent**
  - Gestisce l'invio della posta mediante SMTP
  - E.g., sendmail, exim, qmail
- **MDA=Mail Delivery Agent**
  - Prende i dati dal MTA e li conserva in una mailbox
  - E.g., sendmail, exim, qmail

Un sistema di posta (scenario “con ISP”)

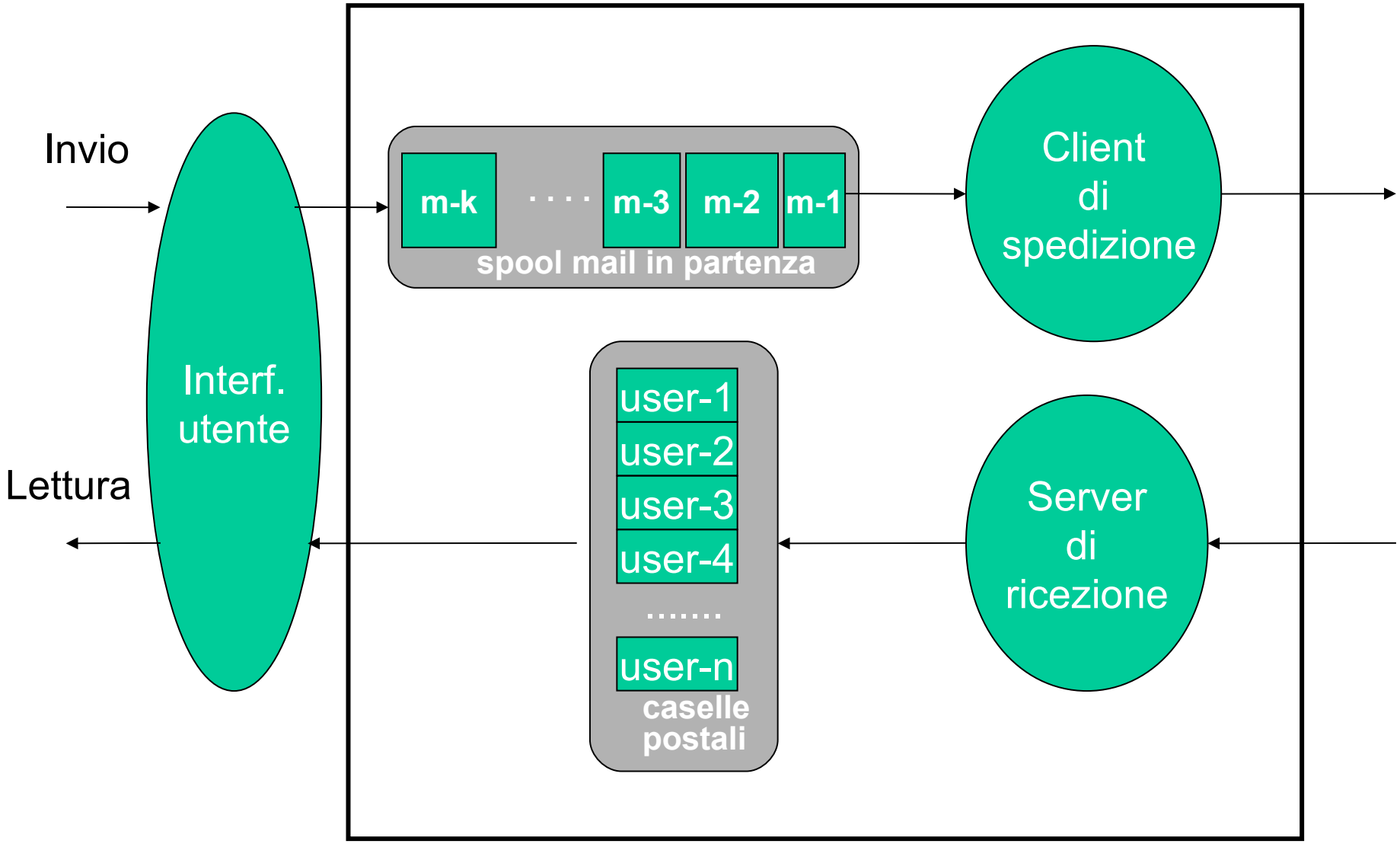


Cambiamenti rispetto a prima

- **Tra il MDA e l'MUA di Bob c'e' uno nuovo strato software**
- **La mailbox viene resa fruibile in remoto mediante appositi server (protocolli POP e IMAP)**
- **Bob usa i server per copiare i contenuti della mailbox remota in locale**

Modulo 3a:
Mail server

Architettura del Mail Transfer Agent



Mail server

- **Il Mail server è più correttamente il Mail Transfer Agent (MTA)**
- **Componenti:**
  - User mailbox contenente i messaggi in ingresso (ancora da leggere) dell'utente
  - Coda di messaggi di posta in uscita (da inviare)
  - Gestione protocollo SMTP tra mail server per inviare messaggi di posta

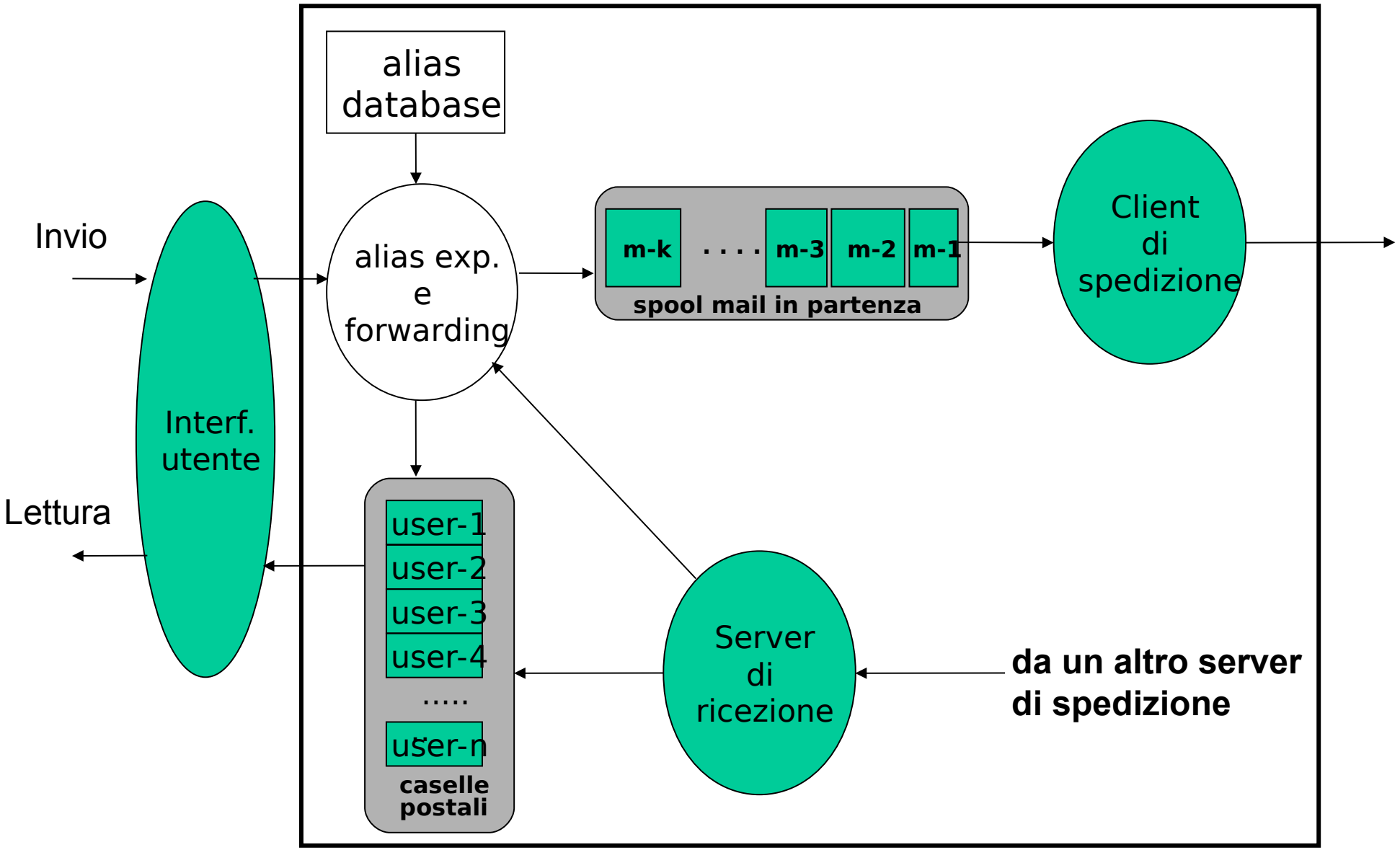
Funzioni mail server

- **Memorizzazione messaggi in arrivo nelle user mailbox**
- **Memorizzazione messaggi in uscita:**
  - Temporaneamente nella coda in uscita
  - Per un periodo di tempo più lungo (prefissato) nel caso di server di destinazione non raggiungibile o non in grado di ricevere messaggi
- **Invio/ricezione messaggi mediante il protocollo SMTP**
- **MTA è sia client sia server di altri mail server**

Destinazioni e mailbox

- **La destinazione è una macchina il cui nome è caratterizzato come “mail-exchange object”**
- **Il nome della mailbox, normalmente corrisponde ad uno username, ma può corrispondere ad un “alias”**
- **Una mail può anche arrivare da un utente interno (quindi non arriva tramite una connessione dall’esterno)**
- **Una mail, sia dall’esterno sia dall’interno, può essere inoltrata ad un’altra destinazione (mail forwarding)**

Architettura di mail con user database



Mail Transfer Agent (MTA)

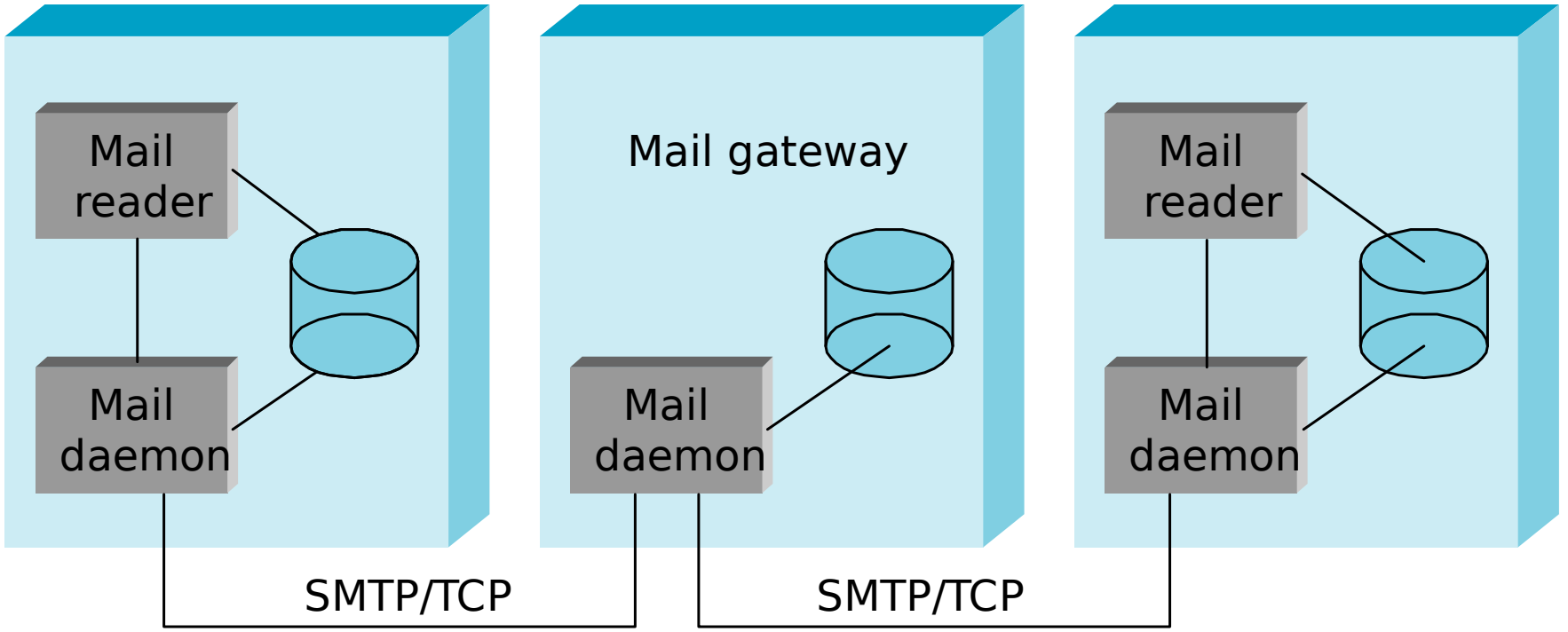
- **E’ il processo principale che trasferisce messaggi da un computer all’altro**
- **E’ il processo che agisce dietro le quinte, in quanto l’utente interagisce con MUA**
- **Può ricevere messaggi da:**
  - Un altro MTA (modalità relaying)
  - Un Mail Submission Agent (MSA) che, a sua volta, ha ricevuto posta da un MUA
  - Direttamente da un MUA, agendo quindi da MSA

Due modalità di trasferimento

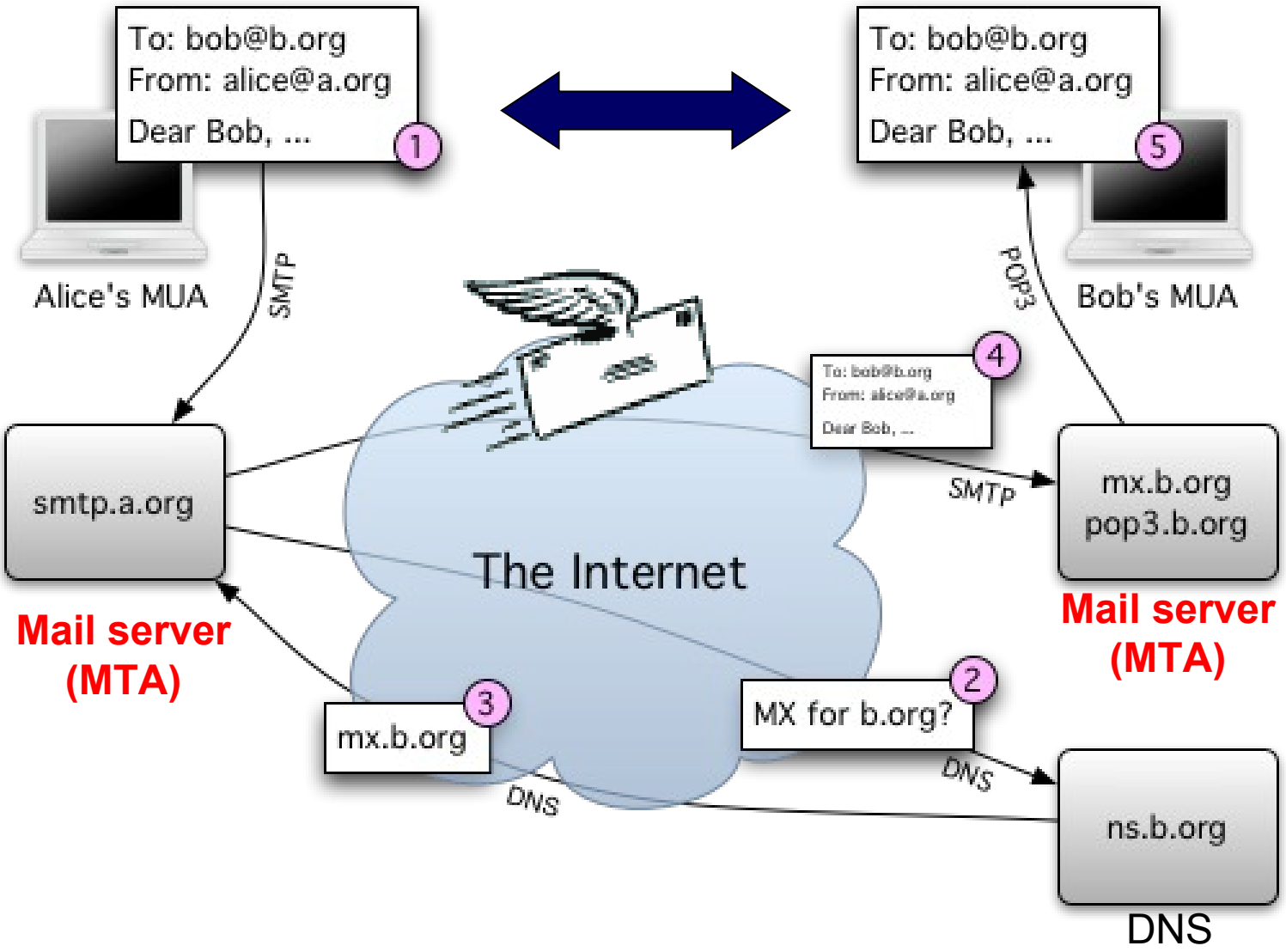
- **Trasferimento diretto: dall'MTA sending server all'MTA receiving server**

- **Trasferimento relay: trasferimento dall'MTA sending server ad un server intermedio (mail gateway, mail bridge, mail relay)**

Trasferimento indiretto (relay)



Le fasi di trasmissione



Interazione con il DNS

- **La prima informazione da conoscere è il mail server che gestisce il dominio di posta del destinatario**
- **Il DNS gestisce un tipo di resource record MX ovvero Mail eXchanger record che specifica come deve essere inoltrato un messaggio. Due informazioni:**
  - elenco di mail server (hostname, indirizzo IP) che possono ricevere l'email per quel dominio
  - priorità relativa

Interazione con il DNS

- **L'MTA mittente cerca di stabilire una connessione SMTP/TCP con il server con priorità più alta e continua fin quando non c'è un server dell'elenco che risponde**
- **Il messaggio viene inoltrato al primo server che risponde**
- **Si parla di un generico Mail Exchange Server perché potrebbe essere l'MTA di destinazione oppure un gateway server intermediario (relay transfer)**

Mail Transfer Agent multipli

- **SMTP server**
  - Gestisce il trasferimento dei messaggi inviati dall'interno del dominio
  - Interagisce direttamente con i client di posta
  - Deve essere molto veloce
  - Gestisce code relativamente piccole

Mail Transfer Agent multipli

- **MX server (Mail eXchanger)**
  - Gestisce la ricezione dei messaggi che provengono dall'esterno del dominio
  - Gestisce code di messaggi in arrivo molto lunghe
  - Richiede molta potenza computazionale
- **MX server (Mail eXchanger)**
  - Gestisce la ricezione dei messaggi che provengono dall'interno del dominio
  - Gestisce code piccole, spesso è un semplice forwarding

Principali applicativi per Mail server

- **Sendmail**
  - Il più antico, diretto discendente di Delivermail (nato per BSD), solo per sistemi *nix. Licenza Open Source
  - Considerato poco sicuro (il primo famoso Internet worm Morris sfuttava una vulnerabilità di questo software per propagarsi)
- **Qmail**
  - Licenza free-to-use (controversa). Per sistemi *nix e Mac OS X

Principali applicativi per Mail server

- **Postfix**
  - Licenza IBM Public License. Per sistemi *nix e Mac OS X
  - Alternativa più sicura degli anni ’90 a Sendmail. Molta attenzione alla sicurezza e affidabilità del codice
  - Presente in tutte le distribuzioni Linux, spesso come MTA di default
- **Exim**
  - Molto semplice da configurare grazie a script di supporto
- **Microsoft Exchange**
  - Solo per sistemi MS

Modulo 3b:
Altri agent del mail server

Altri mail agent del mail server

- **In alcuni casi, il Mail server è costituito da diversi mail agent oltre al Mail Transfer Agent (MTA) sempre presente:**
  - Mail Submission Agent (MSA) – in alcuni casi (posta da utenti interni)
  - Mail Delivery Agent (MDA)

- **Questa organizzazione multi-livello sta diventando la norma nei sistemi di gestione mail delle organizzazioni più complesse**

Consegna mail

- **La comunicazione tra il client di spedizione e il server di ricezione che avviene su protocollo di trasporto TCP garantisce che quando il messaggio è rimosso dallo spool del mittente, è stato recapitato con successo al ricevitore (non ci sono passi intermedi a livelli di mail exchange)**
- **Quando si utilizzano, invece, sistemi di mail che prevedono server intermediari (mail gateway, mail bridges, mail relay), si è garantiti solo fino al primo recapito, dopodiché si perdono le tracce ...**

Necessità di elaborazioni accessorie

- **Logging**
- **Spam detection**
- **Virus detection**
- **Controllo dell'autorizzazione a insiemi di mittenti/destinatari**
  - Il mail delivery può essere assistito da un database tipo LDAP

Mail Delivery Agent (MDA)

- **La consegna di un messaggio ad una mailbox utente oggi avviene spesso mediante un Mail Delivery Agent (MDA)**
- **Invocato da un MTA, è un filtro che elabora le mail arrivate ad un server, effettuando le seguenti operazioni:**
  - Filtering
  - Ordinamento
  - Inserimento in cartelle sulla base di keyword, soggetto, mittente, testo
  - Invio di auto-reply sulla base di eventi stabiliti dall'utente

Mail Delivery Agent (MDA)

- **Su sistemi Unix, Sendmail invoca Procmail**
- **A sua volta, un MDA come Procmail può invocare programmi esterni come antispam e antivirus che possono prendere diversi provvedimenti: avviso (mittente/destinatario), quarantena, eliminazione**

Esempio MTA-MDA

- **Su sistemi Unix, l'MTA Sendmail invoca Procmail che è un MDA**
- **A sua volta, un MDA come Procmail può invocare programmi esterni come antivirus (es., ClamAV) e antispam (es., SpamAssassin) che possono prendere diversi provvedimenti non mutuamente esclusivi:**
  - avvisare il mittente e/o il destinatario
  - mettere il messaggio in quarantena
  - eliminare il messaggio
  - …

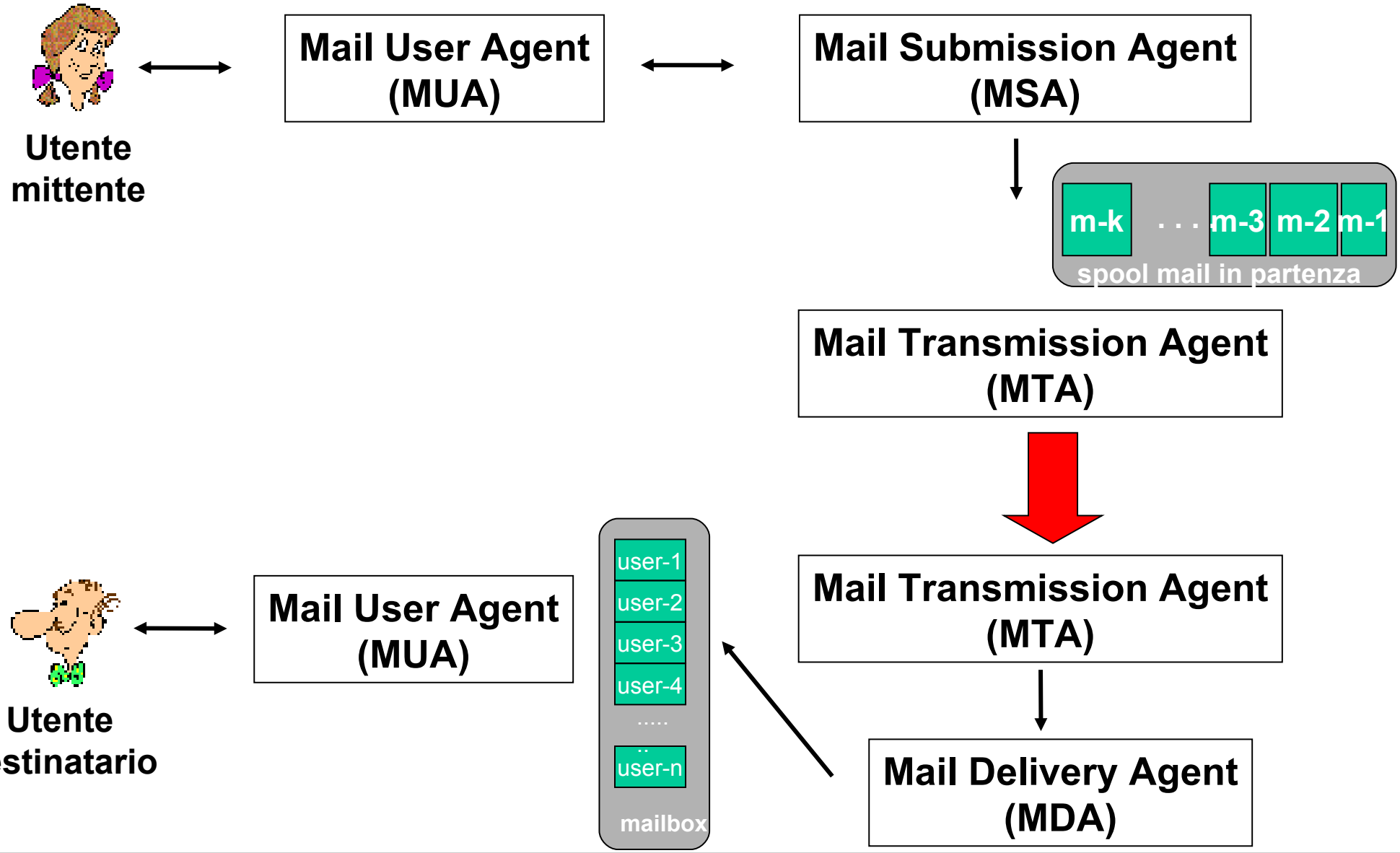
Mail Submission Agent

- **E’ il processo che riceve un messaggio da un MUA e contatta un MTA per inviarlo**
- **Molti MTA agiscono anche da MSA, anche perché queste funzionalità sono sempre state storicamente integrate**

Mail Submission Agent

- **Benefici della separazione:**
  - Interagendo direttamente con un MUA, può correggere o segnalare all'utente mittente alcuni errori evidenti
  - MTA e MSA possono utilizzare diverse politiche di filtro e antispam per l'accettazione dei messaggi
  - Alcuni MSA richiedono autenticazione e quindi è possibile introdurre qualche meccanismo di trust sul mittente
  - Si evita la connessione diretta sulla porta 25 da parte di client "telnet" (si vedrà dopo)
  - Si facilita la gestione della posta per utenti mobili (che possono usare il proprio server di sottomissione anche da reti altrui)

Ricapitolando (un caso articolato)



Modulo 3c: Protocollo SMTP

Simple Mail Transfer Protocol

- **SMTP [RFC 821] [RFC 822] – Agosto 1982**
- **Protocollo tra mail server (ovvero tra Mail Transfer Agent)**
- **Paradigma client/server, in cui:**
  - client: MTA del mittente
  - server: MTA del destinatario
- **Client e server sono in esecuzione su ogni mail server**
- **Usa il protocollo di trasporto TCP per il trasferimento affidabile dei messaggi tra client e server**

Simple Mail Transfer Protocol

- **Il dialogo sender-receiver avviene sulla porta 25 ed è costituito da “frasi” in formato testuale comprensibili**

- **Interazione comando/risposta**
  - Comandi: in formato ASCII-7 bit
  - Risposta: status code e frase di commento

Dialogo sender-receiver (lato sender)

Nome	Formato	Descrizione
HELO	HELO<sp><domain><crlf>	identificazione del sender
MAIL	MAIL<sp>FROM:<rev.-path><crlf>	identifica il mittente
RCPT	RCPT<sp>TO:<forward-path<crlf>	identifica il destinatario
DATA	DATA<crlf>	inizia la trasmissione
RSET	RSET<crlf>	abortisce la transazione
NOOP	NOOP<crlf>	no operation
QUIT	QUIT<crlf>	chiude la connesione TCP
SEND	SEND<sp>FROM :<rev.-path><crlf>	manda la mail al terminale
SOML	SOML<sp>FROM :<rev.-path><crlf>	manda la mail al terminale se possibile se no alla mailbox
SAML	SAML <sp>FROM :<rev.-path><crlf>	manda la mail al terminale e alla mailbox
VRFY	VRFY<sp><string><crlf>	verifica un user name
EXPN	EXPN <sp><string><crlf>	riporta l'appartenenza ad una mailing list
HELP	HELP<sp>[<string>]<crlf>	invia documentazione di sistema
TURN	TURN<crlf>	scambia i ruoli sender e receiver

Dialogo sender-receiver (lato receiver)

Codice	Descrizione (risposta)
211	system status/system help reply
214	*messaggio di help* (pagina di manuale, per una persona)
220	<domain> service ready
221	<domain> service closing transmission channel
250	Requested mail action ok, completed
251	User not local; will forward to <forward-path>
354	Start mail input; en with<crlf>crlf>
421	<domain>Service not avilable, losing transmission channel
450	Requested mail action not taken; mailbox unavailable (eg.: busy)
451	Requested action not taken: insufficient system storage
500	Syntax error, command unrecognized
501	Syntax error in parameter or arguments
502	Comand not implemented
503	Bad sequence of commands
504	Command parameter not implemented
550	Requested mail action not taken: mailbox unavailable (not found)
551	User not local; please try <forward path>
552	Requested action aborted: eceeded storage allocation
553	Requested action aborted: mailbox name not allowed (eg.: syntax error)
554	Transaction failed

Simple Mail Transfer Protocol

Attivazione della connessione TCP

```
S: 220 mail.ucla.edu
C: HELO mail.unimo.it
S: 250  Hello mail.unimo.it, pleased to meet you
C: MAIL FROM: <alice@mail.unimo.it>
S: 250 alice@ mail.unimo.it ... Sender ok
C: RCPT TO: <bob@mail.ucla.edu>
S: 250 bob@ mail.ucla.edu ... Recipient ok
C: DATA
S: 354 Enter mail, end with "." on a line by itself
C:  Do you like computer science books?
C:   How about journals?
C: .
S: 250 Message accepted for delivery
C: QUIT
S: 221 mail.ucla.edu closing connection
```

CRLF.CRLF per indicare la fine del messaggio

Chiusura della connessione TCP

Provare SMTP in “altro modo”

- **telnet servername 25**
  - (servername è il nome del mail server remoto)
- **Attendi la risposta 220 dal server**
- **Inserisci opportunamente i comandi HELO, MAIL FROM, RCPT TO, DATA, QUIT**
- **→ E’ possibile inviare una mail senza usare uno user agent !**
- **E quindi: anche con nome “mittente” fittizio**
- **→ Attenzione: spoofing di email è facile!!**

Modalità di trasferimento

- **Tre fasi del trasferimento**
  - Handshaking (SMTP greetings diverso da TCP handshaking)
  - Trasferimento messaggi
  - Chiusura

- **SMTP usa connessioni TCP persistenti per trasferire più messaggi (se esistono) in una sola volta dallo stesso MTA sender allo stesso MTA receiver**

Formato del messaggio SMTP

- **Il messaggio di mail consiste di due parti:**
  - un header che contiene dei campi codificati
  - il body del messaggio che deve essere un testo in ASCII a 7 bit (recenti evoluzioni consentono anche e-mail in formati più elaborati)
- **Il server SMTP usa CRLF.CRLF per indicare la fine del messaggio**
- **Alcune stringhe di caratteri non sono permesse nel messaggio (es., CRLF.CRLF). Quindi il messaggio deve essere codificato (di solito in base 64 oppure quoted printable)**

Formato del messaggio SMTP

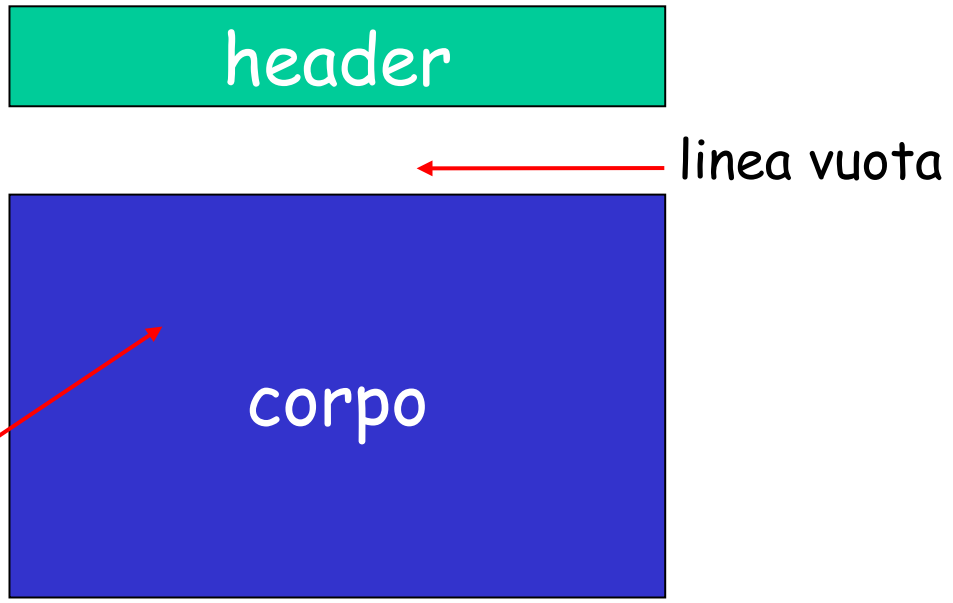
RFC 822 definisce lo standard per il formato del messaggio:

- **Linee di header, es.:**
  - `To:`
  - `From:`
  - `Subject:`

  Diverse dai comandi SMTP!

- **Corpo**
  - **il "messaggio", soltanto in caratteri ASCII**

- **Linea contenente solo '.'**



Esempio

Subject: Avviso
Date: 12/12/2000, 18:25
From: <direttore@dii.unimo.it>
To: <riccardo@weblab.ing..unimo.it>
Reply-to: segretaria@dii.unimo.it

Caro Professore,
volevo comunicarle che.....

- Alcuni campi nell'header sono obbligatori, altri sono opzionali
- Il client di trasmississione interpreta l'header in modo da gestire il dialogo con il server corrispondente
- Quindi trasmette il messaggio una riga alla volta

SMTP vs. HTTP

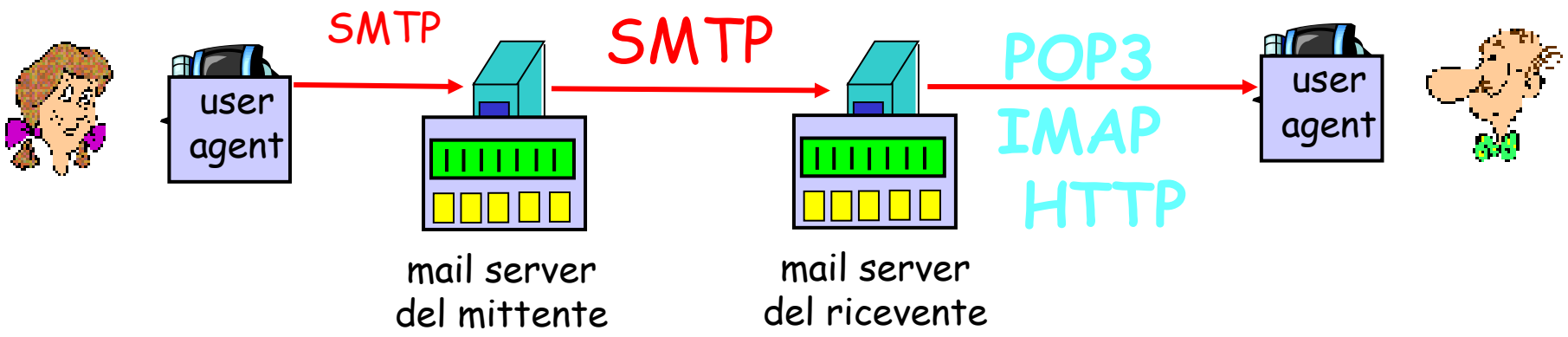
- **HTTP: protocollo pull (client “prende” da server)**
- **SMTP: protocollo push (client “dà” al server)**
- **Entrambi presentano un’interazione di tipo comando/risposta, ed usano codici di stato**
- **HTTP/1.0: oggetti multipli (embedded objects) di una risorsa Web inviati in connessioni TCP separate**
- **HTTP/1.1: oggetti multipli (embedded objects) di una risorsa Web inviati nella stessa connessione TCP**
- **SMTP: parti multiple di un messaggio inviate nella stessa connessione TCP**

Modulo 3d:
Mail User Agent

Mail user agent

- **E' l'applicazione utente "della posta" che comprende varie funzionalità:**
  - interfaccia utente
  - strumenti per composizione, editing, invio
  - strumenti per lettura messaggi di posta
  - strumenti per gestione di attachment
  - strumenti per organizzazione della posta inviata e ricevuta
  - Esempi di applicativi:
    - Outlook (MS)
    - Mozilla Thunderbird / Evolution
    - pine
    - …

Protocolli di accesso alla posta



- **SMTP: consegna/memorizzazione al mail server del destinatario**
- **Protocollo di accesso alla posta: consegna dal mail server**
  - ***POP*: Post Office Protocol [RFC 1939]**
    - **autorizzazione (user agent ↔ mail server) e download**
  - ***IMAP*: Internet Mail Access Protocol [RFC 1730]**
    - **più caratteristiche (maggiore complessità)**
    - **possibilità di manipolazione dei messaggi memorizzati sul server**
  - ***HTTP*: accesso alla posta tramite Web**

Protocollo POP3

- **Fasi di una sessione POP3**
- **Fase di instaurazione della connessione**
  - user agent apre una connessione TCP con mail server (porta 110)
- **Fase di autorizzazione**
  - user agent invia al mail server la propria login e password
- **Fase di transazione**
  - user agent recupera i messaggi
  - user agent può indicare alcuni messaggi affinché siano cancellati (modalità download-and-delete oppure download-and-keep)

Protocollo POP3

- **Fase di aggiornamento**
  - dopo il comando quit eseguito dal client, vengono cancellati dalla mailbox i messaggi eventualmente indicati come tali dall'utente

Protocollo POP3

Fase di autorizzazione

- comandi del client:
  - **`user:` dichiara username**
  - **`pass:` password**
- il server risponde
  - **`+OK`**
  - **`-ERR`**

Fase di transazione, client:

- **`list:`** elenca i numeri dei messaggi
- **`top:`** prendi l'head del messaggio
- **`retr:`** scarica i messaggi tramite il numero
- **`dele:`** cancella
- **`quit:`** esci

```
S: +OK POP3 server ready
C: user alice
S: +OK
C: pass pippo
S: +OK user successfully logged on
```

```
C: list
S: 1 498
S: 2 912
S: .
C: retr 1
S: <message 1 contents>
S: .
C: dele 1
C: retr 2
S: <message 1 contents>
S: .
C: dele 2
C: quit
S: +OK POP3 server signing off
```

Protocollo POP3

- **Modalità di transazione:**
  - Fetch all messages
  - Fetch messages selected by filter
  - Fetch messages selected by user

Protocollo IMAP

- **IMAP (attuale versione 4rev1 – RFC3501)**
- **Più funzionalità e maggiore complessità rispetto al protocollo POP3**
- **Il server IMAP deve essere in grado di gestire una gerarchia di mailbox per ogni utente**

Protocollo IMAP

- **Permette all'utente di modificare la propria mailbox come se fosse locale**
  - Es., creazione di cartelle (folder) remote nella mailbox
  - Es., ricerca (search) di particolari messaggi nelle cartelle remote
- **Permette all'utente di ottenere alcune parti del messaggio**
  - Es., specificare quali attachment "scaricare"
  - Caratteristica utile per connessioni a banda stretta (ad es., dispositivi mobili)

Fasi di una sessione IMAP

- **Fase di instaurazione della connessione**
  - Il Mail User Agent apre una connessione TCP con il mail server (MSA o MTA a seconda del sistema di posta)
- **Fase di autorizzazione**
  - Il Mail User Agent invia al mail server la propria login e password
- **Fase di transazione**
  - comprende comandi client, dati dal server, risultati del server

Accesso alla posta tramite Web

- **Servizi di posta elettronica tramite tecnologie Web (browser e protocollo HTTP)**
- **Il Web browser ha il ruolo di Mail User Agent**
  - comunicazione con mail server tramite protocollo HTTP

Accesso alla posta tramite Web

- **Permette all'utente di modificare la propria mailbox come se fosse locale (analogamente a IMAP)**
  - es., definizione di cartelle (folder) remote nella mailbox
  - es., ricerca di determinati messaggi nelle cartelle remote
- **Pro/Contro**
  - maggiore flessibilità a scapito di minore velocità/efficienza

Esempio di Web client: SquirrelMail

- Package scritto in PHP
- Supporto ai protocolli IMAP e SMTP
- Restituisce tutte le pagine in HTML puro (4.0)
- Supporta tutte le funzionalità avanzate per la gestione delle cartelle:
  - **create**
  - **delete**
  - **rename**
  - **subscribe/unsubscribe**

Modulo 4: MIME

Messaggi non testuali

- **SMTP tratta correttamente soltanto caratteri ASCII a 7 bit**
- **PROBLEMA: Come trasferire un messaggio che contiene altri caratteri (ASCII 8 bit) o addirittura dati in binario?**
- **Metodi gestiti esplicitamente dall'utente:**
  - uuencode
  - binex
  - .....
- **Standard de facto attuale:**
- **Multipurpose Internet Mail Extension (MIME) [RFC 2045, 2056]**

MIME

- **MIME si interpone tra l'interfaccia utente ed il mailer, automatizzando in pratica la procedura di conversione che viene effettuata dai client e dai server quando si spediscono mail in formato SMTP/MIME**
- **Introduce un ulteriore livello di incapsulamento**
- **Poiché l'header dello standard permette solo l'invio di messaggi di caratteri ASCII**
- **→ Si estende l'header del messaggio con 5 campi specifici del MIME**

Estensione header del MIME

- **MIME-version: deve avere il valore 1.0 per indicare la conformità alle specifiche delle RFC**
- **Content-Type: descrive i dati nel “body” in modo tale che l’agente ricevente possa scegliere l’applicazione adatta per decodificare i dati**
- **Content-Transfer-Encoding: tipo di trasformazione usata per il body in modo da renderlo trasmissibile con caratteri ASCII a 7 bit**
- **Contend-ID: usato per identificare le entità del MIME in contesti multipli**
- **Content-Description: una descrizione testuale dell’oggetto codificato (commento)**

MIME Content type

Tipo	Sotto-tipo	descrizione
Text	Plain	testo semplice
Multipart	Mixed	parti indipendenti ma l'ordine si mantiene
	Parallel	parti indipendenti ma l'ordine non si mantiene
	Alternative	versioni alternative della stessa parte
	Digest	simile a mixed ma il tipo/sottotipo di default è message/rfc822
Message	rfc822	il "body" è un messaggio conforme a RFC822
	Partial	frammento di un body più grande, spezzato
	External-body	contiene un puntatore ad un oggetto che esiste altrove
Image	jpeg	formato JPEG codificato JFIF
	gif	formato GIF
Video	mpeg	formato video MPEG
Audio	Basic	canale singolo 8 bit ISDN mu-low codificante un campione a 8KHz
Application	PostScript	file in postscript
	octect-stream	dato binario in byte (8 bit completi)

MIME message type

- **message/rfc822: definisce ricorsivamente il contenuto come un altro messaggio completo (incapsulato nel body di una parte)**
- **message/partial: viene usato per gestire la frammentazione di un messaggio in più messaggi**
  - id: valore comune a tutti frammenti
  - number: numero d'ordine della sequenza
  - total: numero totali di parti
- **message/external-body: usato per accedere a documenti esterni**

MIME message type

- **FTP: seguono i parametri per un comando ftp**
- **TFTP: come sopra**
- **Anon-FTP: ftp con login anonimo**
- **local-file: segue un path-name locale sul ricevitore**
- **AFS: file accessibile via Andrew File System**
- **mail-server: body accessibile inviando una mail a un dato sito**

Tipi/sottotipi MIME principali

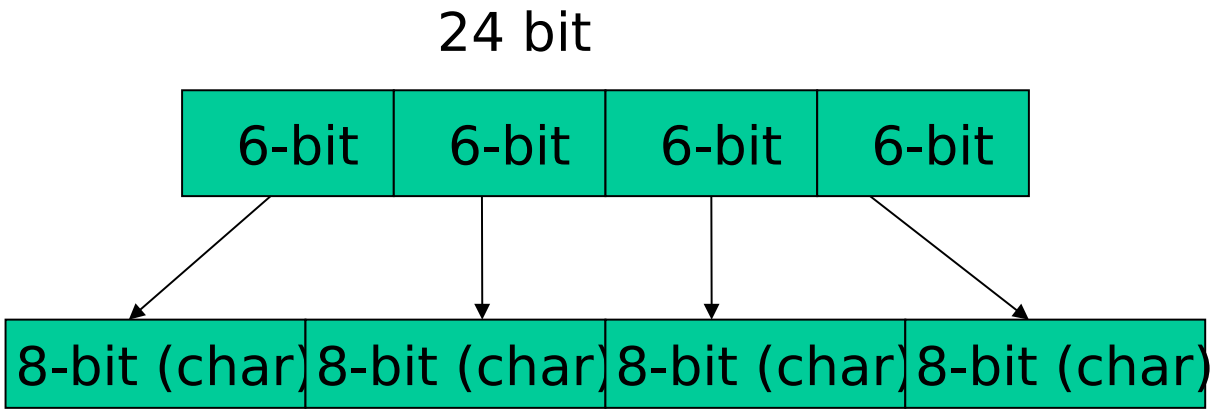
- **Testo**
  - esempio di sottotipi: plain, html
- **Immagini**
  - esempio di sottotipi: jpeg, gif
- **Audio**
  - esempio di sottotipi: basic (8-bit mu-law encoded), 32kadpcm (32 kbps coding)
- **Video**
  - esempio di sottotipi: mpeg, quicktime
- **Applicazioni**
  - Altri dati che devono essere processati prima di essere fruibili all'utente
  - esempio di sottotipi: msword, octet-stream

Proprietà importanti

- **Lo standard MIME è nato per essere espandibile**
- **Le sue definizioni includono metodi per definire nuovi tipi di contenuto e altri attributi MIME**
- **Per questa sua flessibilità viene utilizzato anche da altri protocolli non di posta, es. HTTP**

MIME Content transfer encoding

Codifica	Descrizione
7bit	codifica a 7 bit, linee corte con ASCII
8bit	codifica a 8 bit, linee corte ma anche caratteri a 8 bit
binary	file binario, le linee possono essere anche lunghe
quoted-printable	codifica in modo speciale solo i caratteri non ASCII
base64	codifica in radice 64
x-token	forma di codifica non standard



65 caratteri:
64 di codifica
+
1 di padding

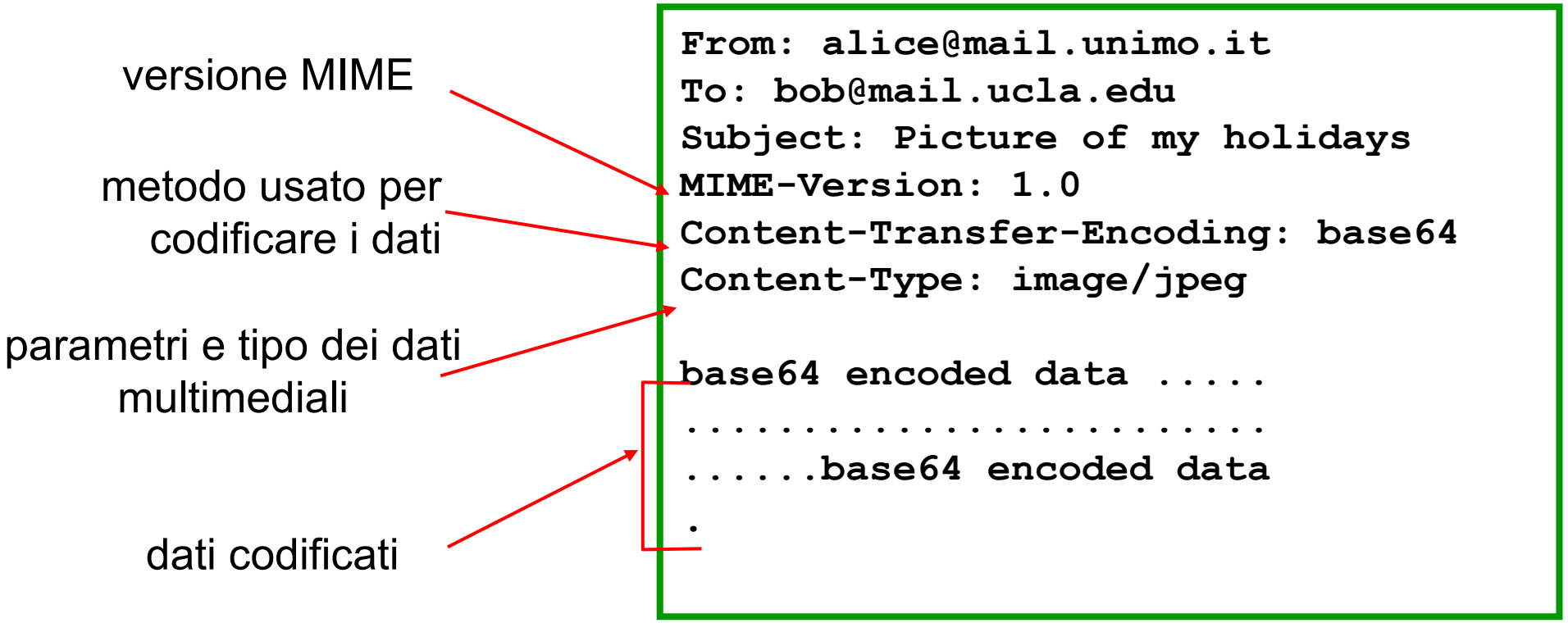
Esempio

Subject: Avviso
Date: 12/12/2000, 18:25
From: <direttore@dii.unimo.it>
To: <mc@dii.unimo.it>
MIME-version: 1.0
Content-type: multipart/mixed; boundary: boundary11

preambolo esplicativo (ignorato da MIME, ma usato da interfacce utente)
--boundary11
testo implicitamente codificato in ASCII.....
--boundary11
Content-type: text/plain; charset=us-ascii
testo esplicitamente codificato in ASCII come testo semplice
--boundary11
epilogo (anche questo ignorato dal MIME)

Formato del messaggio MIME

Ulteriori linee nell'header del messaggio SMTP dichiarano il tipo di contenuto MIME



Esempio MIME

Multipart type: per indicare la presenza di oggetti multipli

```
From: alice@mail.unimo.it
To: bob@mail.ucla.edu
Subject: Picture of my holidays
MIME-Version: 1.0
Content-Type: multipart/mixed; boundary=98766789

--98766789
Content-Transfer-Encoding: quoted-printable
Content-Type: text/plain

Dear Bob,
Please find a picture of my holidays.
--98766789
Content-Transfer-Encoding: base64
Content-Type: image/jpeg

base64 encoded data .....
.........................
......base64 encoded data
--98766789--
```

Modulo 5: Limiti del sistema posta elettronica

(In)sicurezza nell'e-mail

- **Problemi di confidenzialità**
- **Trasferimento in chiaro**
- **Storage di messagi su nodi potenzialmente non sicuri**
- **Problemi di integrità**
- **Il contenuto delle mail può essere alterato**
- **Insicurezza sull'origine**
- **I campi “From” non sono affidabili**

(In)sicurezza nell'e-mail

- **Posso sostituirmi al MUA e creare mail artefatte!**
- **Manca “non-repudiation”**
- **Sistema non affidabile**
- **Chi mi assicura che un messaggio è arrivato?**
- **Mancanza di attenzione ai problemi di sicurezza nel progetto di base (non sono previsti abusi)**

Conseguenze

- **Scarsa confidenzialità**
  → rivelazione di informazioni sensibili
- **Insicurezza sull'origine + problemi di integrità**
  → diffusione di codice malizioso
  - Molti virus si diffondono via email
- **Abusi non previsti + insicurezza sull'origine**
  → possibilità di mail bombing e spam

Inoltre è sempre possibile usare vulnerabilità locali dei singoli server per guadagnare accesso ai sistemi...

SPAM

Origini da una scenetta del Monty Python Flying Circus (la carne in scatola Spam)

→ Azione di diffondere in modalità broadcast (cioè, a tutti i possibili utenti di posta elettronica) messaggi pubblicitari via e-mail

In generale, si considera SPAM qualsiasi e-mail non richiesta e non desiderata

Consuma tempo, spazio su disco e banda

E' sempre fastidiosa, spesso offensiva, e talvolta contenente hoax o scam

Costa milioni di dollari ai grandi provider

Cosa può essere considerato Spam?

Se ricevo notizie di una conferenza che mi interessa è spam?

Volume e frequenza dei messaggi: quando diventa spam?

…

Perché lo SPAM?

Basta fare un po' di conti …

- Inviare e-mail spam a circa 100 milioni di mailbox
- Se anche solo il 10% legge la mail e clicca sul link ➔ si raggiungono 10 milioni di persone
- Se 1% delle persone che va sul sito, sottoscrive per esempio all'offerta di prova per 3 giorni ➔ (100,000 persone) x ($0.50) = $50,000
- Se l'1% di questi, si iscrive per 1 anno ➔ (1,000 persone) x ($144/anno) = $144,000

La potenza del social engineering

Alcune tipologie di spam

- **Vendita di prodotti**

Viagra, Psicofarmaci, Medicine assortite

Software (di solito piratato e spesso pieno di trojan, virus e altri *malware*)

- **Hoax (voci infondate e allarmanti con preghiera di diffusione)**

Catene di S. Antonio,

Petizioni, allarmi sanitari

Allarmi per virus

Un possibile sito cui fare riferimento: http://attivissimo.net

Alcune tipologie di spam

- **Scam ovvero usare lo spam per condurre truffe o peggio.**

se lo spam è reato tanto vale essere davvero cattivi…

- **Esempi di Scam**

Offerte di lavoro part time come consulente finanziario (riciclaggio di denaro)

Consigli di investimento

Nigerian scam (anche noto come 419 – premio ignobel per la letteratura)

Verifiche credenziali di accesso (furto di identità)

- **Spesso lo scam viene “personalizzato” per sembrare piu' attendibile.**

Offuscamento nello Spam

- **Lo spam viene spesso combattuto con software si analisi detti filtri antispam**

- **I filtri (tra le altre cose) analizzan il contenuto della mail per vedere se “sembra spam”**

Se una mail contiene le parole “Viagra” “Cialis” molto probabilmente è spam

- **Per limitare l'efficacia delle contromisure lo spam si è evoluto: mesaggi di spam offuscati**

Tecniche di offuscamento comuni

- **Sostituzione di caratteri in una parola:**

V1agra, Vìagra

- **Inserimento di parole “neutre” nel messaggio per confondere i filtri**

- **Uso di finti tag HTML**

GU<ABC>ARA<DEF>NTEED L<GHI>O<LMN>WEST RA<OPQ>TES ON TH<RST>E PL<UVZ>ANET
→*GUARANTEED LOWEST RATES ON THE PLANET*

- **Uso di immagini per il messaggio invece che testo**

Come migliorare la sicurezza?

- **Estensione dei protocolli e degli standard per la gestione delle posta: molto è disponibile ma pochi lo usano**
- **Estensioni del protocollo SMTP**
  - autenticazione degli utenti
  - cifratura dei dati
  - conferma di ricezione
- **Supporto per la crittografia nei messagi di posta (PGP e GPG)**
  - Firma digitale (messaggi non ripudiabili)
    - non tutti MUA supportano questa feature, OE in particolare
  - Cifratura (messagi non leggibili anche quando sono archiviati)

Evoluzione di SMTP: ESMTP

- **ESMTP: Extended SMTP**
- **Attualmente lo standard SMTP è RFC 2821, che riunisce RFC 821 (SMTP) e RFC 1869 (SMTP Service Extension: EHLO, ...)**
- **Altre estensioni sono in RFC appositi:**

DSN: Delivery status notification, RFC 1891

AUTH: Authenticated SMTP, RFC 2554

STARTTLS: Transport layer security, RFC 3207

...