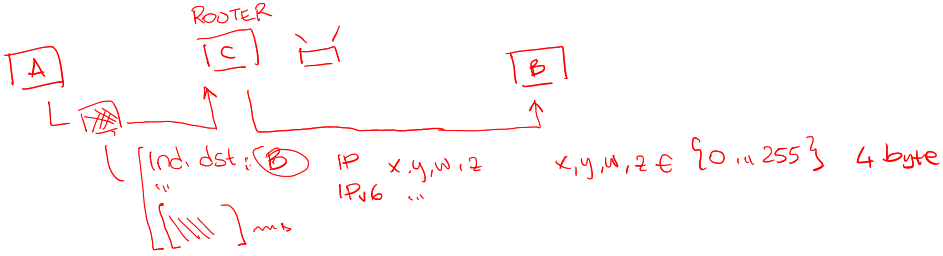
Laboratorio di Comunicazione mediante Calcolatore – A.A. 2020/2021
02 - Protocolli di rete

Leonardo Robol <leonardo.robol@unipi.it>
Sergio Steffè <steffe@dm.unipi.it>

5 ottobre 2020

Dati e protocolli

- La trasmissione di dati in Internet avviene tramite scambio di pacchetti di dati ("protocolli di trasmissione").
- Al di sopra di questo livello, le applicazioni devono accordarsi su un linguaggio per interpretare i dati scambiati ("protocolli delle applicazioni").



Il DNS

Un esempio di protocollo è il DNS (Domain Name Resolution).

- Se vogliamo visitare un sito (ad esempio `https://www.google.com`) dobbiamo inviare una richiesta ad un server, che risponda al nome `www.google.com`.
- Il protocollo IP, che usiamo per collegarci a Internet, non conosce nomi, ma solo numeri come `192.168.0.1`.
- I server DNS servono ad effettuare in modo trasparente questa traduzione
- Chiaramente, il loro indirizzo va conosciuto in modo numerico!

Indirizzo DNS:
- Provider (Tiscali, TIM, ...)
- Google 8.8.8.8 8.8.4.4
- ...

Altre informazioni sui server DNS

Un server DNS non fornisce solamente traduzioni da nome ad indirizzo, ma anche altre informazioni, ad esempio:

- Records `A`, `AAAA`: indirizzi IPv4 o IPv6.
- Records `MX`: chi gestisce la posta per questo dominio?
- Records `SOA`, `NS`: informazioni sui nameserver del dominio.
- Records `TXT`: ancora altre informazioni varie.

Sotto Linux possiamo usare il comando `host` (oppure anche `dig`, `nslookup`) per interrogare un server DNS.

Email e protocolli

Lo scambio di email viene gestito in maniera simile, con diversi protocolli:

- SMTP: "Simple Mail Transfer Protocol", gestisce la spedizione delle e-mail. Questo è il vero protagonista dello smistamento delle e-mail.
- IMAP, POP3, sono invece protocolli che permettono agli utenti di consultare la propria casella e-mail.
- Una mail, nella forma più semplice, è sostanzialmente un file di testo con un'intestazione.
- L'intestazione (header) contiene i metadati del messaggio: data e ora, oggetto, mittente, destinatario, ecc.

Come viene spedita una email?

Supponiamo che Alice voglia scrivere a Bob.

Come viene spedita una email?

Supponiamo che Alice voglia scrivere a Bob. bob@bob.com

1. Alice digita il messaggio sul suo calcolatore.
2. Tramite un software, Alice affida il messaggio ad un server SMTP (il relay), che lo prende in carico.
3. Tramite uno o più passaggi, il server consegna il messaggio al server in carico di gestire le mail di Bob (ricordate `host -t MX bob.com`?
4. Bob accede alla sua casella (con il suo software preferito, o una webmail), e legge il messaggio di Alice.

Come viene spedita una email?

Supponiamo che Alice voglia scrivere a Bob.

1. Alice digita il messaggio sul suo calcolatore.
2. Tramite un software, Alice affida il messaggio ad un server SMTP (il relay), che lo prende in carico.
3. Tramite uno o più passaggi, il server consegna il messaggio al server in carico di gestire le mail di Bob (ricordate `host -t MX bob.com`?
4. Bob accede alla sua casella (con il suo software preferito, o una webmail), e legge il messaggio di Alice.

La trasmissione dei messaggi avviene tramite SMTP.

Il protocollo SMTP

Assumiamo di voler scrivere a `leonardo.robol@unipi.it`.

```
leonardo@georg:~$ host -t MX unipi.it
unipi.it mail is handled by 50 emailsecurity.unipi.it.
```

Ora abbiamo determinato chi gestisce la posta per `@unipi.it`. Colleghiamoci con `telnet`.

Il protocollo SMTP (continua)

```
leonardo@georg:~$ telnet emailsecurity.unipi.it 25
Trying 131.114.142.96...
Connected to emailsecurity.unipi.it.
Escape character is '^]'.
220 esra2.unipi.it ESMTP SonicWall (9.1.3.3987)
```

porta

Il protocollo SMTP (continua)

```
leonardo@georg:~$ telnet emailsecurity.unipi.it 25
Trying 131.114.142.96...
Connected to emailsecurity.unipi.it.
Escape character is '^]'.
220 esra2.unipi.it ESMTP SonicWall (9.1.3.3987)

helo georg
250 esra2.unipi.it
```

Il protocollo SMTP (continua)

```
leonardo@georg:~$ telnet emailsecurity.unipi.it 25
Trying 131.114.142.96...
Connected to emailsecurity.unipi.it.
Escape character is '^]'.
220 esra2.unipi.it ESMTP SonicWall (9.1.3.3987)

helo georg
250 esra2.unipi.it

mail from: <leonardo@georg.cs.dm.unipi.it>
250 2.1.0 MAIL ok
```

Il protocollo SMTP (continua)

```
leonardo@georg:~$ telnet emailsecurity.unipi.it 25
Trying 131.114.142.96...
Connected to emailsecurity.unipi.it.
Escape character is '^]'.
220 esra2.unipi.it ESMTP SonicWall (9.1.3.3987)

helo georg
250 esra2.unipi.it

mail from: <leonardo@georg.cs.dm.unipi.it>
250 2.1.0 MAIL ok

rcpt to: <leonardo.robol@unipi.it>
250 2.1.5 <leonardo.robol@unipi.it> ok
```

Il protocollo SMTP (continua)

```
data
354 3.0.0 End Data with <CR><LF>.<CR><LF>
Subject: Email di prova
[ contenuto del messaggio qui ]
.
250 2.6.0 Message Accepted
```

Il protocollo SMTP (continua)

```
data
354 3.0.0 End Data with <CR><LF>.<CR><LF>
Subject: Email di prova
[ contenuto del messaggio qui ]
.
250 2.6.0 Message Accepted

\quit
221 2.0.0 esra2.unipi.it says goodbye; [...]
Connection closed by foreign host.
```

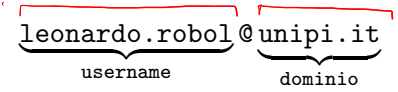
Il protocollo SMTP (continua)

```
data
354 3.0.0 End Data with <CR><LF>.<CR><LF>
Subject: Email di prova
[ contenuto del messaggio qui ]
.
250 2.6.0 Message Accepted

quit
221 2.0.0 esra2.unipi.it says goodbye; [...]
Connection closed by foreign host.
```

- Per una serie di ragioni, questo messaggio sarà sicuramente finito nello SPAM.
- Il protollo SMTP è leggermente più complesso al giorno d'oggi: firme crittografiche permettono di controllare (in parte) che la catena di trasmissione sia corretta – noi abbiamo ignorato tutto questo.

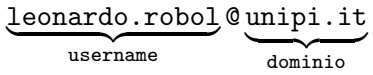
Struttura di un indirizzo email



username è la parte dell'indirizzo che (solitamente) descrive l'utente locale.

dominio invece determina dove dev'essere recapitata l'email (ancora, utilizzando `host -t MX unipi.it`).

Struttura di un indirizzo email



username è la parte dell'indirizzo che (solitamente) descrive l'utente locale.

dominio invece determina dove dev'essere recapitata l'email (ancora, utilizzando `host -t MX unipi.it`).

Esistono molte varianti di questa sintassi, in particolare per lo username – sono perlopiù in disuso. Una che può tornare utile è:

`leonardo.robol+keyword@unipi.it`

- Alcuni caratteri non sono ammessi.
- Ultimamente, è stato esteso il set di caratteri utilizzabile anche per i domini (UTF8).

Struttura di un messagio

Un'email è sostanzialmente un file di testo, con questa struttura:

```
Subject: Messaggio
From: Leonardo Robol <leonardo.robol@unipi.it>
To: Leonardo Robol <leonardo.robol@unipi.it>

Contenuto del messaggio qui
```

- La prima parte si chiama header, e contiene linee del tipo `Chiave: valore`.
- La seconda è il corpo del messaggio (body).
- Sono separate da una linea vuota.

Dissezione di un header

Consideriamo questa e-mail che mi sono auto-mandato:

Dissezione di un header

Consideriamo questa e-mail che mi sono auto-mandato:

Dal programma di posta è possibile aprire il sorgente della e-mail, ed ispezionarne il contenuto.

Dissezione di un header

```
Return-Path: <leonardo.robol@unipi.it>

Received: from mx3.unipi.it (mx3.unipi.it [131.114.21.49])
by mbox5.unipi.it (Postfix) with ESMTP id 8F11DE02B8
for <a019485@mbox5.unipi.it>;
Fri,  9 Nov 2018 08:01:59 +0100 (CET)

Received: from localhost (localhost [127.0.0.1])
by mx3.unipi.it (Postfix) with ESMTP id 95B11C02E3
for <leonardo.robol@unipi.it>;
Fri,  9 Nov 2018 08:01:59 +0100 (CET)

[...] (altri 5 passaggi intermedi)

Received: from georg (dhcp05.cs.dm.unipi.it [131.114.10.165])
(Authenticated User)
by smtp.unipi.it (Postfix) with ESMTPSA id B4A8B40DA1
for <leonardo.robol@unipi.it>;
Fri,  9 Nov 2018 08:01:58 +0100 (CET)
```

Dissezione di un header (parte 2)

```
[...]

Date: Fri, 09 Nov 2018 08:02:19 +0100
From: Leonardo Robol <leonardo.robol@unipi.it>
Subject: Messaggio di prova
To: LEONARDO ROBOL <leonardo.robol@unipi.it>
Message-Id: <1541746939.31612.2@smtp.unipi.it>
X-Mailer: geary/0.12.4
```

Dissezione di un header (parte 2)

```
[...]

Date: Fri, 09 Nov 2018 08:02:19 +0100
From: Leonardo Robol <leonardo.robol@unipi.it>
Subject: Messaggio di prova
To: LEONARDO ROBOL <leonardo.robol@unipi.it>
Message-Id: <1541746939.31612.2@smtp.unipi.it>
X-Mailer: geary/0.12.4


Questo e' un messaggio di prova.

-- Leonardo.
```

MIME

- In realtà, raramente i messaggi hanno un corpo così semplice e leggibile;
- Molti software utilizzando un formato più complesso per avere un documento con più parti nel testo (ad esempio, testo formattato, poi uno o più allegati, ...).
- A volte il contenuto viene codificato in modo particolare (base64) – rendendolo di fatto illeggibile ad un essere umano.

Server di posta

leonardo@georg → valido solo in locale

- Normalmente, noi affidiamo le nostre e-mail ad un server di posta (SMTP) che gira su qualche server (`smtp.unipi.it`, ad esempio).
- Su sistemi Linux è però (abbastanza) comune avere un server su ogni macchina.
- Questo permette di spedire e-mail utilizzando comandi appositi: `sendmail`, `mail`, ...
- Le macchine dell'Aula 4 hanno un server SMTP ciascuna: questi comandi sono disponibili!
- In realtà, questi server redirigono semplicemente tutte le e-mail ad un altro mail server del dipartimento.

Il comando mail

Con il comando `mail` possiamo scrivere e-mail dalla linea di comando:

```
$ mail -s "Subject" utente@gmail.com
Messaggio di prova
.
Cc:
```

CTRL+D

Il comando mail

Con il comando `mail` possiamo scrivere e-mail dalla linea di comando:

```
$ mail -s "Subject" utente@gmail.com
Messaggio di prova
.
Cc:
```

Mini-esercizio: come possiamo utilizzare questo comando per spedire una stessa e-mail a moltissime persone?

Una mailing list "fatta in casa"

```
$ cat indirizzi.txt
utente1@gmail.com
utente2@libero.it
[...]

$ cat email.txt
From: Me <me@unipi.it>

Ciao,
[...]

$ for i in $(cat indirizzi.txt); do
    mail -s "Saluto" $i < email.txt
  done
```

SPAM

- La spedizione di e-mail è libera: chiunque può mandarmi un messaggio.
- Questo causa un'abbondanza di messaggi di spam nelle nostre caselle, in particolare se il nostro indirizzo viene pubblicato online.
- La maggior parte dei mailserver utilizza dei filtri di vario tipo per cercare di limitare queste dinamiche.
- Due tipo di filtri principali: blocking list e filtri statistici / Bayesiani.

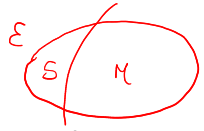
lista spammatori

+ greylist

Blocking list

- Idea molto semplice: mantenere una lista di PC che sono noti per spedire mail di spam (e.g., PC infetti, o server tipicamente utilizzati da spammer).
- La lista viene continuamente aggiornata, se il vostro PC è infetto ci potete facilmente finire per sbaglio!
- Esiste una procedura di removal per chiedere di essere rimossi dalla lista.

Filtri Bayesiani



Consideriamo $\mathcal{E}$ l'insieme di tutte le e-mail, partizionato come

$$\mathcal{E} = \mathcal{S} \cup \mathcal{M}, \qquad \mathcal{S} \cap \mathcal{M} = \emptyset,$$

e dove $\mathcal{S}$ sono le e-mail di spam, e $\mathcal{M}$ quelle "regolari".

- Idea: l'occorrenza di varie parole è diversa in $\mathcal{S}$ ed in $\mathcal{M}$;
- Dato un sample di emails in $\mathcal{M}$ ed $\mathcal{S}$, possiamo studiare le probabilità che una nuova email e stia in $\mathcal{S}$:

$$\mathbb{P}(e \in \mathcal{S} \mid e \in W) = \frac{\mathbb{P}(e \in W \mid e \in \mathcal{S}) \cdot \mathbb{P}(e \in \mathcal{S})}{\mathbb{P}(e \in W)}$$

dove W è un'insieme di e-mail contenente certe parole (e la parte a destra è fatta di cose note o "facilmente stimabili").

Ma da dove viene il termine SPAM?

Ma da dove viene il termine SPAM?

Client di posta

Ci sono svariati client di posta:

- Webmail (GMail, Roundcube (email di ateneo), ...)
- Client desktop (Outlook, Apple Mail, Thunderbird, ...)
- Client per smartphone

Mailing list

- Spesso, si vuole comunicare con un insieme di persone (o discutere di qualcosa).
- Per questo, sono state inventate le mailing-list.
- Quando si spedisce ad una mailing list, tutti ricevono il messaggio; rispondendo alla mailing list si continua la discussione.
- Voi siete già iscritti a varie mailing list: Studenti, Galois, ...

Collegamento ad un computer remoto

- Uno degli utilizzi principali della rete è stato quello di interagire con macchine fisicamente distanti o inacessibili.
- Questo rimane molto importante ancora oggi per chi lavora in ambito scientifico: come possiamo effettuare dei calcoli su un supercomputer come questo?

Telnet

- Verso la fine degli anni '60, i computer si trovavano solo nelle università. Tipicamente erano in stanze inaccessibili.
- Si poteva interagire con dei "terminali", come questo:

- Viene sviluppato il programma `telnet`, che permette di emulare una "telescrivente" da un altro computer connesso in rete.

Telnet

- Negli anni '70, la rete era un lusso per pochi.
- Di conseguenza, la sicurezza del `telnet` era inesistente. Come una telescrivente, tutto quello che era scritto o stampato veniva inviato e ricevuto senza nessun filtro o crittografia.

Telnet

- Negli anni '70, la rete era un lusso per pochi.
- Di conseguenza, la sicurezza del `telnet` era inesistente. Come una telescrivente, tutto quello che era scritto o stampato veniva inviato e ricevuto senza nessun filtro o crittografia.

Il programma server è concettualmente talmente semplice che possiamo emularlo in poche righe sul terminale, usando tool come `netcat`.

Provate su due terminali diversi:

VT100

```
$ nc -l -p 10023
$ telnet localhost 10023
```

SSH

Al giorno d'oggi, `telnet` non viene più utilizzato.

- Chiunque potrebbe ascoltare la "conversazione", rubando ad esempio la nostra password.
- Non c'è nessun modo di garantire che il computer a cui ci stiamo collegando sia "autentico", qualcuno potrebbe aver dirottato la connessione.

SSH

Al giorno d'oggi, `telnet` non viene più utilizzato.

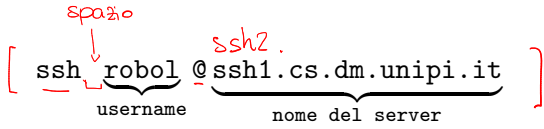
- Chiunque potrebbe ascoltare la "conversazione", rubando ad esempio la nostra password.
- Non c'è nessun modo di garantire che il computer a cui ci stiamo collegando sia "autentico", qualcuno potrebbe aver dirottato la connessione.

Questi problemi e limitazioni sono risolti dal programma che lo ha sostituito, ovvero `ssh`. Vediamo un esempio.

```
$ ssh leonardo@georg
The authenticity of host 'georg (x.y.z.w)' can't be established.
ECDSA key fingerprint is SHA256:xcn6kDnHQzfKjijUuhpgeGyYN5naeAD7r24SX9IwCCI.
Are you sure you want to continue connecting (yes/no/[fingerprint])? yes
Warning: Permanently added 'georg' (ECDSA) to the list of known hosts.
leonardo@georg's password:

leonardo@georg:~$
```

Struttura del comando SSH



- Se lo username è lo stesso sui due PC, può essere omesso.
- Questo comando ci permette di aprire una sessione non grafica sul computer remoto.
- Sarà lo step preliminare di tutti gli esercizi in laboratorio.

Se utilizzate Linux o MAC OS X, avete già il comando ssh a disposizione. Su Windows, è possibile utilizzare il software Putty, che potete scaricare dagli appunti del corso.