Seconda esercitazione

Sommario

- Configurazione della rete con DHCP
  - Funzionamento
  - Configurazione lato server
  - Configurazione lato client

Sommario

- Test di connettività
  - ping
  - traceroute
- Test del DNS
  - nslookup
  - host
- Osservare i protocolli
  - tcpdump (accenni)

Configurazione della rete con DHCP

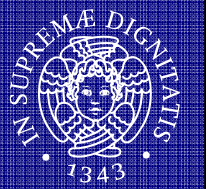
Configurazione manuale - Problemi

- Richiede molto tempo
- Possibilità di errori
- Riconfigurazione faticosa
- Spreco indirizzi

Configurazione della rete con DHCP

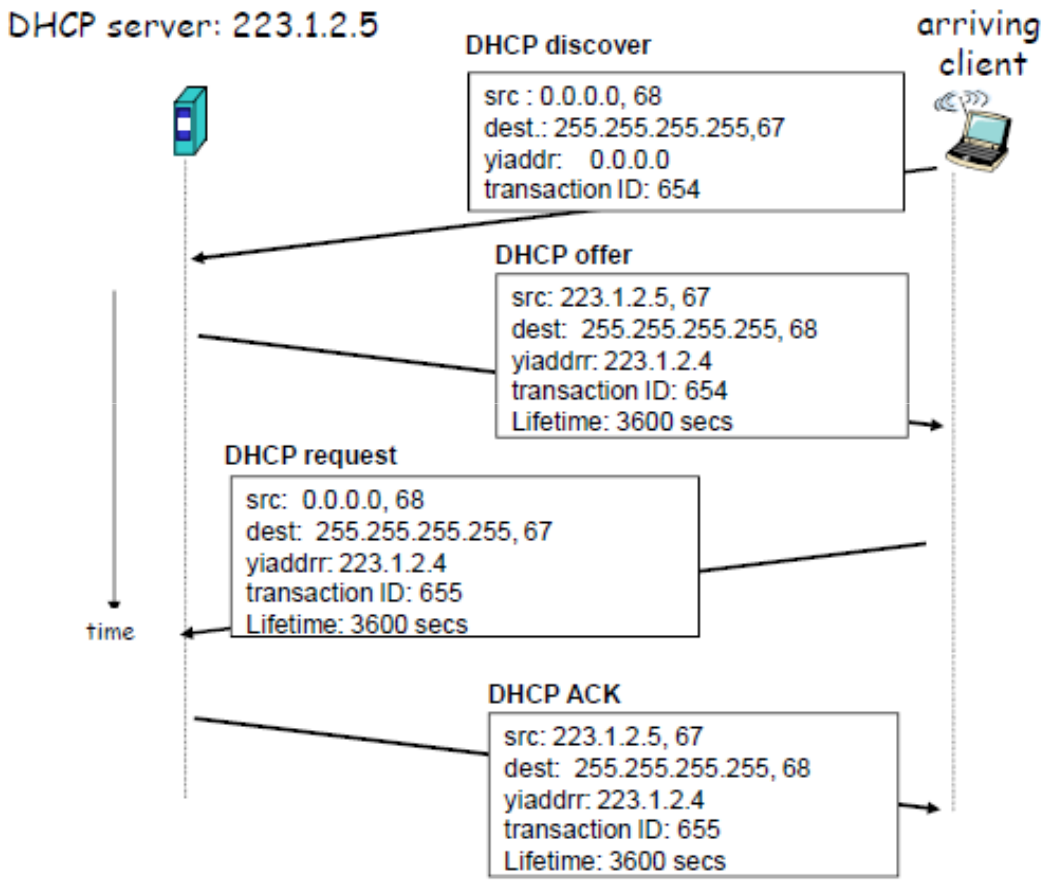
- **DHCP (Dynamic Host Configuration Protocol):** permette la configurazione dinamica dei parametri del TCP/IP.
- Descrive i passi attraverso i quali un sistema può ottenere le informazioni necessarie per comunicare attraverso quella rete.
- Il DHCP server ha il compito di fornire ad ogni client che ne fa richiesta una serie di parametri per la configurazione della rete:
  - ⇨ indirizzo ip
  - ⇨ indirizzo del gateway
  - ⇨ indirizzo del DNS server
  - ⇨ ...

Configurazione della rete con DHCP



- Tutte queste informazioni sono valide solo per un certo periodo di tempo (configurato dall'amministratore del server DHCP).
  - In questo modo, gli indirizzi IP bloccati da client che non sono più connessi alla rete possono essere riutilizzati automaticamente.

DHCP: funzionamento



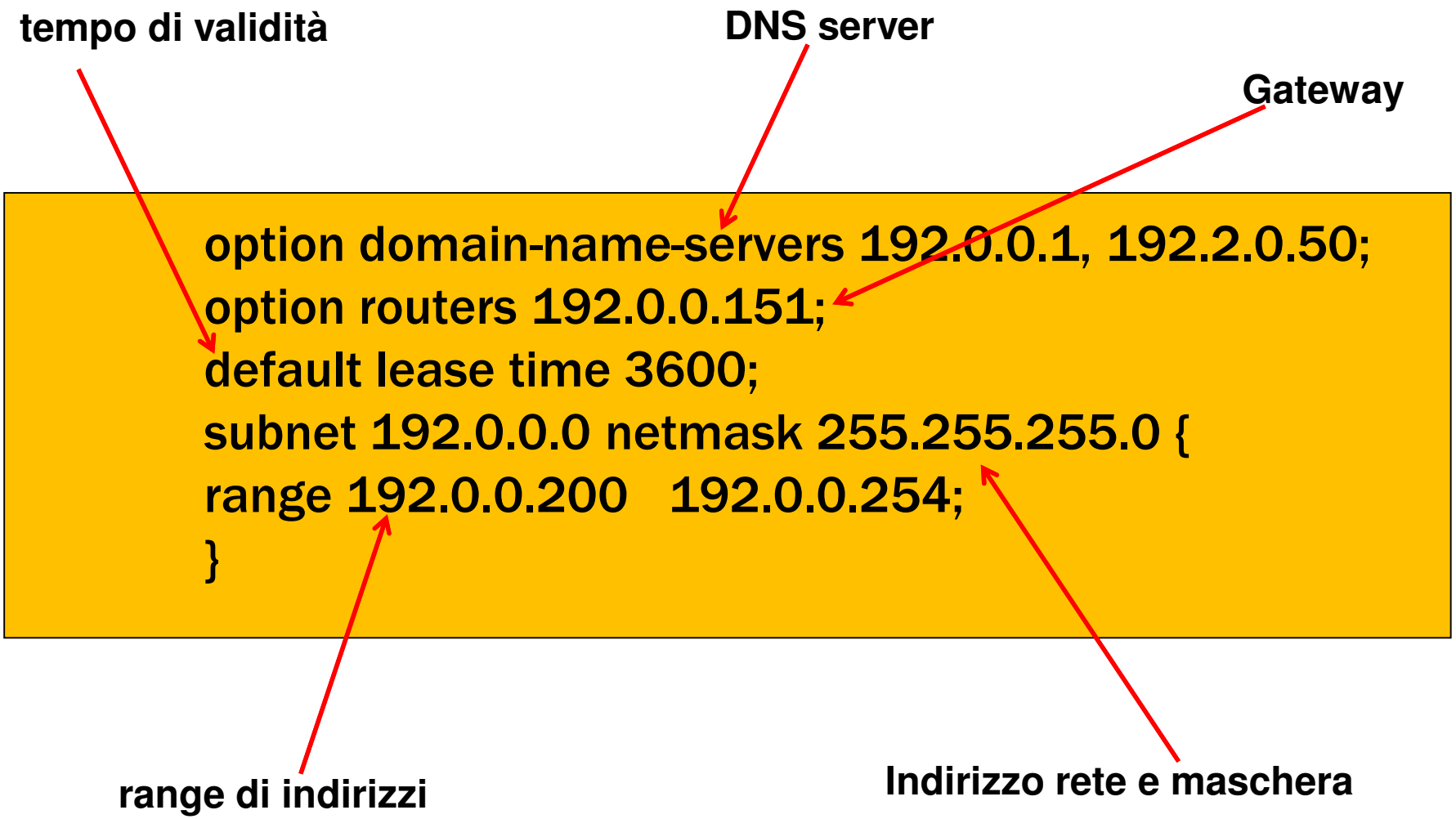
DHCP: server

- **Per installare il dhcp server:**
  - **comando:**
    - ⇨ `apt-get install dhcp3-server`
- **File di configurazione del server:**
  - ⇨ `/etc/dhcp/dhcpd.conf`
- **Avvio del server:**
  - ⇨ `/etc/init.d/isc-dhcp-server start`

DHCP: esempio dhcpd.conf

```
option domain-name-servers 192.0.0.1, 192.2.0.50;
option routers 192.0.0.151;
default lease time 3600;
subnet 192.0.0.0 netmask 255.255.255.0 {
range 192.0.0.200   192.0.0.254;
}
```

DHCP: esempio dhcpd.conf



DHCP: modifiche dhcpd.conf

- **Dopo aver modificato il file** `dhcpd.conf` **bisogna riavviare il server con il comando:**
  - ⇨ `/etc/init.d/isc-dhcp-server restart`

DHCP client: /etc/network/interfaces

- **Esempio:**

```
auto lo eth0
iface lo inet loopback
iface eth0 inet dhcp
```

- **Con questa configurazione all'avvio del sistema operativo verrà utilizzato il DHCP per configurare eth0.**

Comando dhclient

`dhclient`

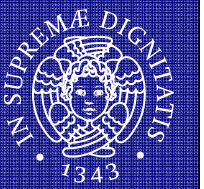
- **Deve essere specificato il nome dell'interfaccia da configurare**
  - **Esempio:**
    - ⇨ `dhclient eth0`
- **Se non passate il nome dell'interfaccia prova a configurare tutte le interfacce.**
- `-p <numero porta>`
  - **Per usare una porta diversa da quella di default.**

Test di connettività

ICMP

- *Internet Control Message Protocol* (Protocollo per i Messaggi di Controllo in Internet).
- Il protocollo IP fornisce un meccanismo di trasferimento dei pacchetti dal mittente al destinatario secondo un approccio *best-effort.*
- Questo vuol dire che l'IP non è in grado di garantire la consegna dei pacchetti al destinatario.
- ICMP segnala solamente errori e malfunzionamenti, ma non esegue alcuna correzione

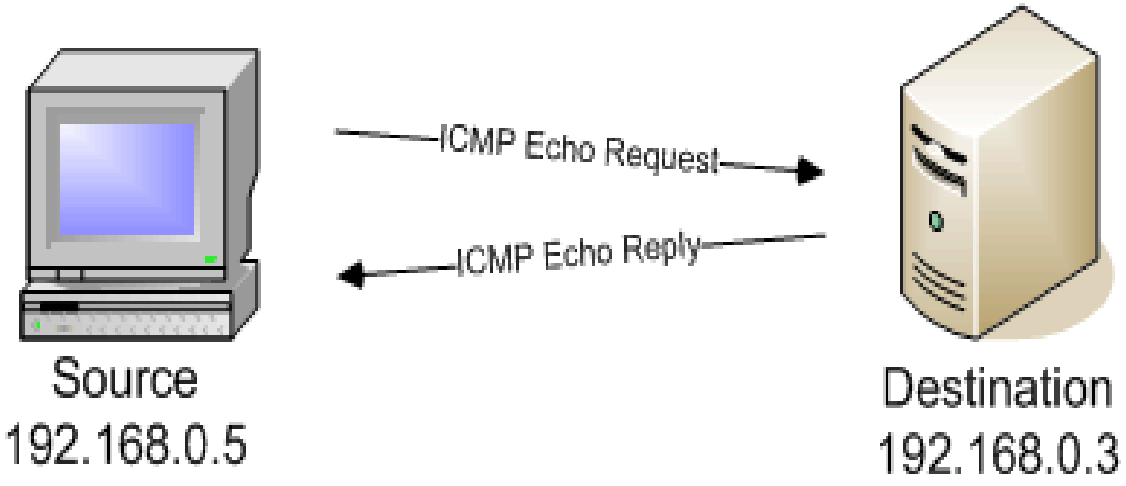
ping



- Serve per testare la connessione di rete tra l'host sul quale viene eseguito ed un host remoto.

- Invia una successione di pacchetti ICMP `ECHO REQUEST` all'host remoto.

ping


ICMP Echo Request
ICMP Echo Reply
Source
192.168.0.5
Destination
192.168.0.3
A
B

Esempio

- Supponiamo che dalla **stazione A** si voglia controllare l'integrità della connessione fino alla **stazione B**.
- Si esegue il comando `ping`, passandogli come argomento l'indirizzo della **stazione B**.
- Il programma manda una serie di messaggi ICMP `ECHO_REQUEST` (generalmente uno al secondo) dalla **stazione A** verso la **stazione B**.
- Quando la **stazione B** riceve un pacchetto `ECHO_REQUEST`, risponde con un nuovo datagramma `ECHO_REPLY`, che viene mandato indietro alla **stazione A**.
- `ping` userà le informazioni così collezionate (esistenza dei pacchetti di ritorno, tempo intercorso per ogni pacchetto, etc.) per calcolare dei valori statistici sulla bontà della connessione e presentarli all'utente.

ping: output

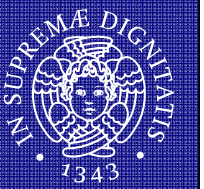
```
ping www.google.it
PING www.google.it (64.233.183.103) 56(84) bytes of data.
64 bytes from www.google.it (64.233.183.103): icmp_seq=1
   ttl=232 time=91.6 ms
64 bytes from www.google.it (64.233.183.103): icmp_seq=2
   ttl=232 time=94.6 ms

--- www.google.it ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time
   4001ms
rtt min/avg/max/mdev = 69.754/87.351/101.647/11.455 ms
```

- **dimensione del pacchetto** ECHO_REPLAY
- **indirizzo IP di DEST**
- **numero di sequenza della risposta**
- **“time-to-live”** (TTL)
- **“round-trip time”** (RTT)
- **risultati statistici**

ping: possibili errori

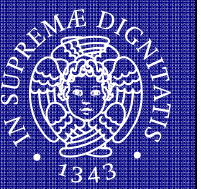
- **Network unreachable :**
  - l'host locale non ha una route valida per l'host remoto
- **No answer (100% packet loss) :**
  - l'host destinatario del ping non risponde ai messaggi.
- **Unknown host:**
  - il name service non è stato in grado di tradurre il nome in un indirizzo

`ping`: opzioni



- `-c count`
  - Stop dopo l'invio (e ricezione) di count pacchetti `ECHO_RESPONSE`
- `-i wait`
  - Aspetta wait secondi tra gli invii dei pacchetti.
  - Il default è di aspettare per un secondo tra ciascun pacchetto.
- `-n`
  - Solo output numerico. Non verrà fatto nessun tentativo di cercare nomi simbolici per gli indirizzi dell'host.
- `-q`
  - Output silenzioso. Non è visualizzato nulla tranne le linee di sommario all'avvio e quando termina.
- `-s dimensione pacchetto`
  - Specifica il numero di byte di dati da inviare.
  - Il default è 56, che si traduce in 64 byte di dati ICMP quando combinato con gli 8 byte dei dati di intestazione di ICMP.

traceroute

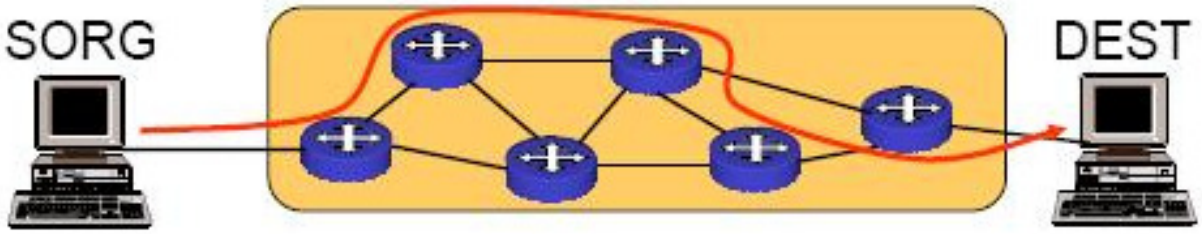


- Permette di conoscere e testare il percorso seguito dai pacchetti inviati da una sorgente e diretti verso una destinazione.
- Utilizza dei pacchetti UDP e le informazioni fornite dall'ICMP.

```
traceroute <destinazione>
```

- Ogni pacchetto ha lo scopo di attivare un messaggio di errore da qualche parte lungo il percorso.
- `traceroute` inizia la trasmissione di pacchetti (utilizzando il protocollo UDP) con un TTL molto basso.
- Incrementando lentamente il valore del TTL, `traceroute` riesce a conoscere gli indirizzi dei nodi attraversati, purché tutto funzioni come previsto (cioè che i vari nodi generino correttamente i messaggi ICMP di errore).
- `traceroute` genera un elenco di host a partire da primo nodo attraversato, fino all'ultimo che rappresenta la destinazione richiesta.
- Se in alcuni punti non si ottiene risposta, i nodi ipotizzati vengono segnalati con degli asterischi.

traceroute: esempio



- SORG **invia a** DEST **una serie di pacchetti con un** TIME-TO-LIVE (TTL) **progressivo da 1 a 30 (opzione –m per cambiarlo)**
- **Ciascun nodo intermedio decrementa** TTL
- **Il nodo che decrementa** TTL **a** 0 **invia a** SORG **un pacchetto ICMP di tipo** TIME_EXCEEDED
- SORG **costruisce una lista dei nodi attraversati fino a** DEST

traceroute: output


```
traceroute to www.google.it (173.194.35.151), 30 hops max, 60 byte packets
 1  10.115.255.254 (10.115.255.254)  0.195 ms  0.212 ms  0.241 ms
 2  131.114.58.1 (131.114.58.1)  2.080 ms  2.074 ms  2.131 ms
 3  131.114.186.33 (131.114.186.33)  0.825 ms  1.240 ms  1.252 ms
 4  131.114.192.205 (131.114.192.205)  1.194 ms  1.185 ms  1.167 ms
 5  ru-unipi-rt-pi1.pi1.garr.net (193.206.136.13)  1.126 ms  1.171 ms  1.213 ms
 6  rt-pi1-rt-to1.to1.garr.net (193.206.134.73)  5.495 ms  5.025 ms  5.037 ms
 7  rt-to1-rt-mi2.mi2.garr.net (193.206.134.41)  7.886 ms  7.845 ms  8.101 ms
 8  193.206.129.134 (193.206.129.134)  8.210 ms  8.203 ms  8.221 ms
 9  216.239.47.128 (216.239.47.128)  8.230 ms  8.261 ms  8.290 ms
10  216.239.48.122 (216.239.48.122)  18.151 ms  17.703 ms  17.839 ms
11  209.85.250.35 (209.85.250.35)  18.085 ms  18.290 ms  18.255 ms
12  muc03s01-in-f23.1e100.net (173.194.35.151)  17.795 ms  17.733 ms  17.729 ms
```

traceroute: domande

- Perché DEST manda un messaggio ICMP di errore?
- Come fa SORG a capire che un messaggio di errore arriva da DEST e quindi che deve terminare?

Test del DNS

nslookup

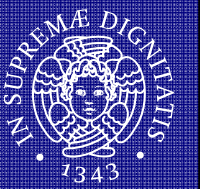
- `Name Server Lookup` **è uno strumento presente in tutti i sistemi operativi che utilizzano il protocollo TCP/IP**
- **Consente di effettuare delle query ad un server DNS per la risoluzione di indirizzi IP o hostname**
- **Si usa per poter ottenere da un dominio il relativo indirizzo IP o nome host e viceversa.**
- **Si può utilizzare in due modi: interattivo e non interattivo.**

`nslookup` interattivo

- Questa modalità permette di effettuare più query e visualizza i singoli risultati.
- Viene abilitato in modo automatico quando:
  - il comando non è seguito da argomenti
  - se il primo argomento è un trattino (-) seguito dal secondo argomento che corrisponde all'host name o all'ip del name server.
- `exit`: permette di uscire dalla shell interattiva

```
$ nslookup
> www.ing.unipi.it
Server:         131.114.21.15
Address:        131.114.21.15#53
www.ing.unipi.it  canonical name =www.web.ing.unipi.it.
Name:   www.web.ing.unipi.it
Address: 131.114.28.27
>
```

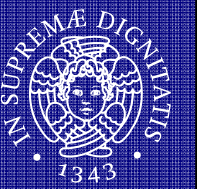
`nslookup` non interattivo



- **Permette di effettuare una sola query e ovviamente visualizza il risultato della singola query.**

```
$ nslookup www.ing.unipi.it
Server:         131.114.21.15
Address:        131.114.21.15#53
www.ing.unipi.it        canonical name
  =www.web.ing.unipi.it.
Name:   www.web.ing.unipi.it
Address: 131.114.28.27
$
```

host

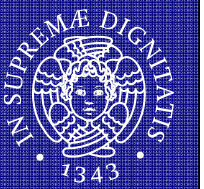


- `host` **è considerato il sostituto ufficiale di** `nslookup`**.**
- **Viene utilizzato per risolvere i nomi in indirizzi numerici e viceversa.**
- **Se lanciato senza parametri ci presenta una lista delle opzioni possibili e una breve spiegazione della loro funzione.**

```
host [opzioni] host [server]
```

Osservare i protocolli

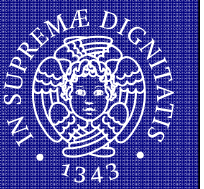
tcpdump(accenni)



- Strumento per l'analisi del traffico che avviene nella rete fisica a cui si è collegati.
- È uno strumento di sniffing particolarmente flessibile
- Si setta in modalità promiscua cioè non vede solo i pacchetti diretti a lui ma tutto il traffico
- NON visualizza il contenuto dei pacchetti ma solo le loro intestazioni (protocollo, IP sorgente, destinazione, porte ecc.) per cui si presta bene alla diagnostica di problemi di networking
- Sintassi

```
tcpdump [opzioni] [espressioni]
```

tcpdump: esempi




```
tcpdump port 80
```

- **Visualizza solo i pacchetti che hanno come sorgente o destinazione la** `porta 80`

```
tcpdump host 192.168.0.150
```

- **Visualizza solo i pacchetti che hanno come IP sorgente o destinazione** `192.168.0.150`**.**

```
tcpdump host 10.0.0.150 and not port 22
```

- **Visualizza solo i pacchetti relativi all'host** `10.0.0.150` **che non usino la porta** `ssh` **(**`and not port 22`**).**