

Altri protocolli applicativi

1 Domain Name System

Per accedere a un servizio tramite Internet, è necessario conoscere l'indirizzo IP del server, ma gli indirizzi IP non sono facili da ricordare per gli utenti. Allora, si usano invece degli **indirizzi simbolici**, detti anche **logici**, come ad esempio `www.uninsubria.it`.

Il **Domain Name System (DNS)** ha la funzione principale di **risolvere** (tradurre) gli indirizzi simbolici nei corrispondenti indirizzi IP (e viceversa). Le altre sue funzioni, invece, sono:

- *host aliasing*: definizione di *alias* (nomi alternativi) per uno stesso host;
- *mail server aliasing*: definizione di alias per un server di posta;
- *load distribution*: distribuzione del carico tra più server.

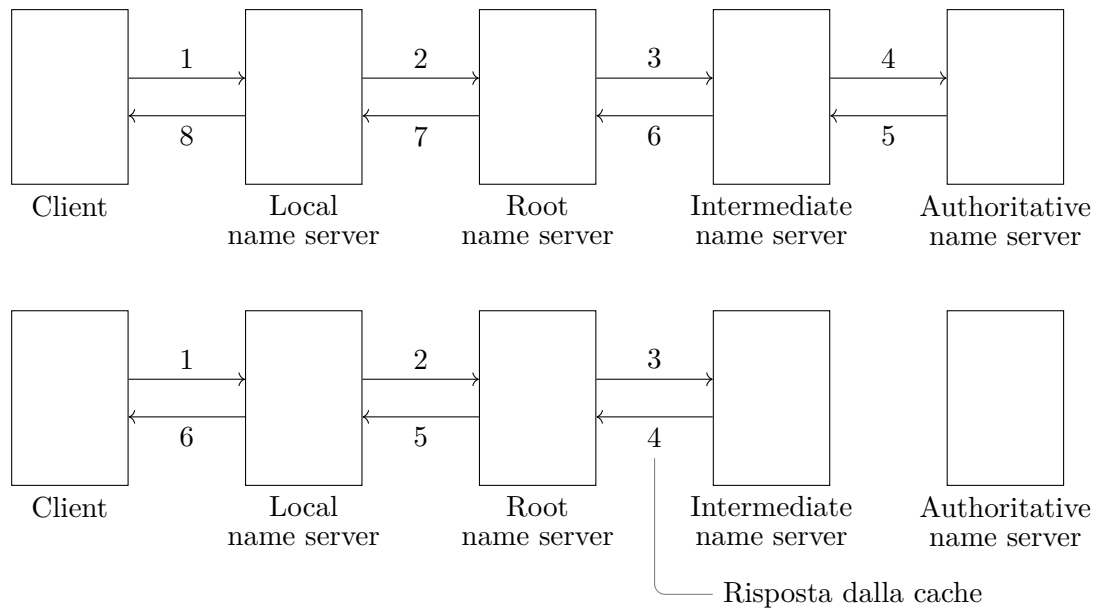
Il DNS è un database distribuito, composto da una gerarchia di **name server** (server dei nomi) situati in tutto il mondo. Lo scambio delle informazioni tra i client e i name server avviene tramite un apposito protocollo applicativo, chiamato semplicemente DNS. Esso usa il protocollo di trasposto UDP: un indirizzo simbolico inserito da un utente deve essere risolto il più velocemente possibile, quindi è meglio evitare l'overhead del TCP.

1.1 Funzionamento

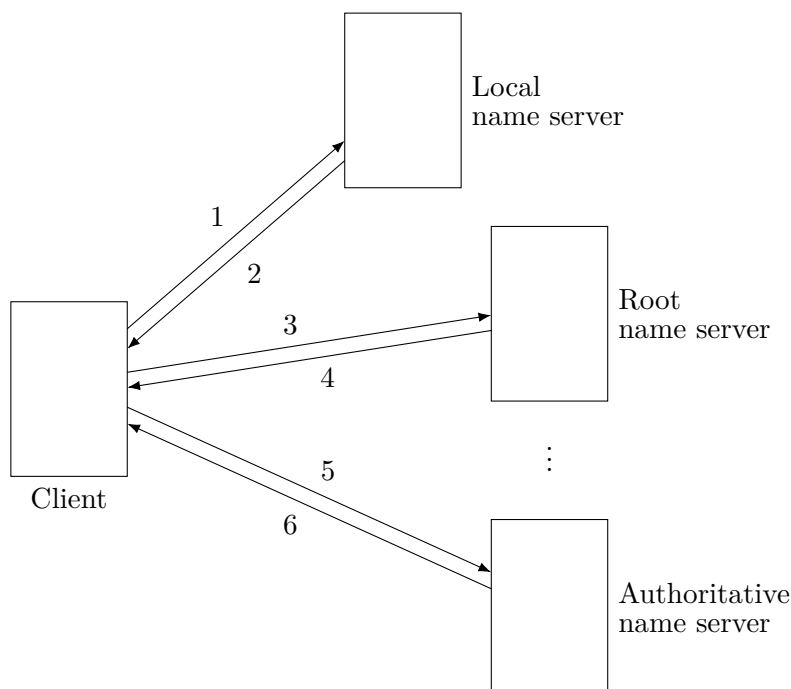
Per eseguire la risoluzione di un indirizzo simbolico, un client deve contattare uno o più name server, fino a raggiungerne uno che conosca l'indirizzo IP corrispondente. Il primo name server che il client interroga è quello del proprio ISP: ogni ISP ha un name server di default, usato da tutti i clienti dell'ISP che necessitano di una risoluzione di indirizzo. Se questo non conosce l'indirizzo IP corrispondente, allora la richiesta deve essere inoltrata a un altro server, e ciò può avvenire in modo ricorsivo o iterativo.

Nella **risoluzione ricorsiva**, il client comunica solo con il *local name server* (il server dell'ISP). Se questo non ha le informazioni necessarie a risolvere l'indirizzo, inoltra la richiesta al *root name server* (il livello più alto nella gerarchia). Questo può a sua volta restituire l'indirizzo IP risolto, oppure inoltrare la richiesta a un *intermediate name server*. Quest'ultimo, infine, inoltra se necessario la richiesta a un *authoritative name server* (server dei nomi assoluto), che conosce sicuramente l'indirizzo IP richiesto. Il risultato della risoluzione viene poi rimandato indietro nella "catena" di server, fino al

client, e intanto ciascun server nella catena lo può salvare in una propria *cache*, in modo da poter fornire la risposta a interrogazioni successive senza bisogno di contattare ancora l'authoritative name server.



Nella **risoluzione iterativa**, invece, il client contatta in sequenza i server della gerarchia (local, root, intermediate e authoritative): ciascun server o fornisce direttamente l'indirizzo risolto, oppure indica il prossimo name server da contattare. In questa modalità, le risposte arrivano direttamente al client, senza passare da server intermedi, ma ciò significa che i server non possono implementare meccanismi di caching.



Tutti questi aspetti sono trasparenti dal punto di vista delle applicazioni, per le quali il DNS è solo una *black box* che traduce gli indirizzi simbolici in indirizzi IP (e viceversa).

1.2 Messaggi DNS

Un messaggio DNS contiene vari campi, alcuni dei quali sono un identificativo, delle etichette che indicano

- se si tratta di un'interrogazione (query) o di una risposta (reply),
- se la risposta è assoluta, cioè prodotta da un authoritative name server,
- se l'interrogazione è ricorsiva, cioè inoltrata da un altro name server,
- se il server supporta la ricorsione,

e un numero variabile di risoluzioni, ciascuna rappresentata da un **resource record (RR)**.

Ogni resource record è a sua volta composto da diversi campi, tra cui i più importanti sono **type**, **name** e **value**. Il campo type indica il tipo di informazione contenuta nel record, determinando così il significato di name e value:

- se type = A, allora name è un *hostname* (indirizzo simbolico) e value è l'*indirizzo IP* corrispondente;

- se `type = NS`, allora `name` è un *dominio* e `value` è l'*hostname di un authoritative name server* per quel dominio;
- se `type = CNAME`, allora `name` è un *alias* di un host e `value` è il suo *hostname canonico*;
- se `type = MX`, allora `name` è un *alias* e `value` è l'*hostname di un server di posta* che ha tale alias.

I resource record di tipo A sono quelli corrispondenti alla funzione principale del DNS, la risoluzione degli indirizzi simbolici in indirizzi IP.

2 Posta elettronica

Nella rete Internet, la posta elettronica è basata su quattro componenti:

- gli **user agent**, gli applicativi usati dagli utenti per comporre, inviare e leggere messaggi;
- i **mail server**, che contengono i messaggi in ingresso e uscita e gestiscono le **mailbox** (caselle postali) degli utenti;
- **SMTP**, il protocollo usato dai mail server per il trasferimento dei messaggi;
- **POP3** e **IMAP**, i protocolli per l'accesso alle mailbox.

Quando un utente vuole inviare un messaggio email, lo compone utilizzando il proprio user agent, e lo invia al proprio server di posta, che lo inserisce in una coda (gestita solitamente secondo la politica *FIFO, First In First Out*:¹ il primo messaggio inserito nella coda sarà il primo a essere processato), e successivamente lo inoltra, tramite il protocollo SMTP, al mail server del destinatario,² Su quest'ultimo, il messaggio viene salvato nella mailbox del destinatario, alla quale l'utente destinatario può accedere con il proprio user agent, tramite il protocollo POP3 o IMAP.

2.1 SMTP

SMTP, Simple Mail Transfer Protocol, è il protocollo usato dal mail server del mittente, che assume il ruolo di client SMTP, per inoltrare un messaggio al mail server del destinatario, che ricopre il ruolo di server SMTP. A livello di trasporto, SMTP usa il protocollo TCP, perché è necessaria l'affidabilità e non importano i ritardi. Il numero di porta well-known assegnato a SMTP è il 25.

¹In generale, esistono anche altre politiche di gestione delle code, come ad esempio *LIFO, Last In First Out*, secondo la quale l'ultimo elemento inserito nella coda è il primo a essere estratto.

²L'indirizzo IP di questo server viene ottenuto risolvendo tramite DNS l'indirizzo email del destinatario.

Il client SMTP interagisce con il server inviando una serie di comandi, e il server risponde con dei codici numerici che indicano se tali comandi siano stati eseguiti correttamente o meno.

Ogni messaggio di posta elettronica è formato da degli **header** e da un **body**. I principali header sono **From** (l'indirizzo del mittente), **To** (l'indirizzo del destinatario) e **Subject** (l'oggetto del messaggio), mentre il body è il testo vero e proprio del messaggio.

Per permettere il trasferimento di contenuti multimediali, o in generale di tutto ciò che non sia testo ASCII, sono state definite delle estensioni chiamate **MIME, Multipurpose Internet Mail Extensions**, basate su alcuni header aggiuntivi che caratterizzano i contenuti dei messaggi:

- **Content-Type** indica il tipo di contenuto del messaggio;
- **Content-Transfer-Encoding** specifica la codifica usata per rappresentare il contenuto del messaggio.

2.2 POP3

POP3, Post Office Protocol (versione 3), è un protocollo per l'accesso a una mailbox, che usa TCP sulla porta well-known 110. Esso opera in tre fasi:

1. *autenticazione* del client, con nome utente e password;
2. *transazione*: il client scarica i messaggi ed eventualmente li contrassegna per la cancellazione;
3. *aggiornamento della mailbox*: quando si chiude la sessione, vengono eliminati i messaggi contrassegnati.

Il client POP3 interagisce con il server mediante vari comandi, che servono per effettuare l'autenticazione, elencare, scaricare e contrassegnare i messaggi, terminare la sessione, ecc.

2.3 IMAP

Un altro protocollo per l'accesso a una mailbox è **IMAP, Internet Message Access Protocol**, anch'esso basato su TCP.

IMAP consente una gestione dei messaggi basata sulle cartelle, permettendo all'utente di creare cartelle sul server e spostare i messaggi da una cartella a un'altra. Inoltre, è possibile eseguire la ricerca di messaggi, e recuperare parti di messaggi.

3 Real-time Transport Protocol

Il **Real-time Transport Protocol (RTP)** è un protocollo usato per il trasporto di contenuti multimediali nelle applicazioni real-time. Per evitare i ritardi introdotti dal TCP, esso usa il protocollo di trasporto UDP.

Al fine di supportare flussi multimediali eterogenei, RTP consente di definire, per ogni classe di applicazioni,

- un *profilo*, che descrive i contenuti dell'header RTP;
- uno o più *formati*, che indicano come interpretare i dati contenuti in un pacchetto RTP.

Alcuni dei principali campi presenti nell'header RTP sono i seguenti:

- *payload type*: indica il tipo di dato multimediale contenuto nel pacchetto;
- *timestamp*: serve per la sincronizzazione, che è importante nelle applicazioni real-time;
- *sequence number*: consente l'ordinamento dei pacchetti;
- *source ID*: un identificativo della sorgente del pacchetto.

Si osservi che, siccome usa UDP e non TCP, RTP deve implementare alcuni degli stessi servizi che sarebbero altrimenti offerti da TCP, come ad esempio l'ordinamento dei pacchetti e la sincronizzazione tra mittente e destinatario.

Insieme all'RTP si usa l'**RTP Control Protocol (RTCP)**, per lo scambio di informazioni di controllo tra client e server. In particolare, il client genera e invia al server dei report contenenti:

- l'SSRC dello stream RTP (un identificativo usato per la sincronizzazione di mittente e destinatario);
- la frazione di pacchetti persi;
- l'ultimo numero di sequenza;
- il jitter degli arrivi dei pacchetti.

Analogamente, il server genera dei report contenenti:

- l'SSRC dello stream RTP;
- il timestamp e l'ora effettiva;
- il numero di pacchetti inviati;
- il numero di byte inviati.

4 Distribuzione dei contenuti

Il tempo di accesso ai contenuti disponibili sul Web è principalmente dovuto a:

- link a bassa velocità, che comportano elevati tempi di trasmissione;
- link congestionati, con elevati tempi di accodamento;
- server Web sovraccarichi.

Una soluzione per ridurre il tempo di accesso è allora replicare gli stessi contenuti su una molteplicità di server, e scegliere ogni volta il server più idoneo per un determinato client.

Un modo per replicare i contenuti su un server più vicino al client è realizzare una **web cache** tramite un **proxy server**,³ che può essere situato anche direttamente nella LAN dell'utente. Tutte le richieste HTTP vengono fatte passare attraverso il proxy server, che inizialmente inoltra le richieste ai server di origine, ma intanto salva localmente i contenuti restituiti nelle risposte HTTP. Quando poi il proxy riceve richieste successive per gli stessi contenuti, è in grado di rispondere immediatamente con le sue copie locali, senza bisogno di contattare ancora i server di origine.

Una replicazione dei contenuti su scala più ampia può essere ottenuta tramite le **Content Distribution Network (CDN)**. Una CDN offre i propri servizi ai **content provider** (le organizzazioni che vogliono rendere disponibili dei contenuti agli utenti di Internet), fornendo dei proxy server su cui vengono replicati i contenuti, e mettendo a disposizione meccanismi per la selezione del server CDN più opportuno.

4.1 Peer-to-peer

Un modello alternativo di distribuzione dei contenuti è il **peer-to-peer (P2P)**, nel quale gli host connessi alla rete — chiamati **peer**, cioè “pari” — possono scaricare oggetti l'uno dall'altro. In questo caso, i ruoli di client e server sono dinamici: uno stesso peer a volte fornisce un servizio e altre volte lo richiede.

Quando un peer deve usufruire di un servizio, ha bisogno di selezionare un altro peer a cui richiederlo. Ciò può avvenire in modi diversi:

- con una directory centralizzata;
- con una directory decentralizzata;
- mediante l'invio di interrogazioni sulla rete (query flooding).

³I proxy server possono essere usati non solo per il web, ma anche per altre applicazioni. Ad esempio, in una LAN è solitamente presente un proxy DNS server, che memorizza localmente le risoluzioni di indirizzi più recenti e/o frequenti.