

Protocols i Funcionament de la Capa Aplicació

Maria dels Àngels Cerveró Abelló

26 d'octubre de 2015

1 Relació entre el Model TCP/IP i el Model OSI

La Capa d'Aplicació del Model TCP/IP correspon a les capes Aplicació, Presentació i Sessió del Model OSI. No obstant això, com que els protocols TCP/IP es van crear abans que apareguessin les primeres interfícies gràfiques i els continguts multimèdia, alguns d'ells no implementen la funcionalitat de les capes Presentació i Sessió.

Capa d'Aplicació

Té tres components bàsics: les aplicacions, els serveis i els protocols.

Les aplicacions són la **interfície** (els programes) que permet que els usuaris es puguin comunicar a través de la xarxa. Aquestes aplicacions s'encarreguen de **formatejar** (codificar), **transmetre** i **interpretar** els missatges.

Els serveis són tots aquells programes que s'encarreguen de **donar resposta** a les peticions que els usuaris fan a través de les aplicacions. És a dir, les aplicacions es comuniquen amb els serveis per tal d'obtenir la informació que demana l'usuari.

Els protocols d'aquesta capa permeten l'**intercanvi de dades** entre els programes dels dispositius d'origen i de destí.

Capa de Presentació

Funcionalitats:

1. Codificació i decodificació de les dades de la Capa d'Aplicació
2. Compresió i descompresió de les dades
3. Encriptació i desencriptació de les dades

Per exemple, la Capa de Presentació és l'encarregada de codificar i decodificar amb el códeg d'MPEG (per a vídeos) o amb els formats JPEG i PNG (per a imatges).

Capa de Sessió

Funcionalitats:

1. Crear, mantenir i finalitzar el diàleg entre les aplicacions d'origen i de destí
2. Reiniciar sessions que hagin estat tallades

La majoria d'aplicacions actuals ja incorporen totes aquestes funcionalitats. Per exemple, el client de Gmail a través del navegador ens permet

- entrar amb usuari i contrasenya (Sessió);
- veure arxius adjunts (Presentació);
- tenir una interfície per escriure i enviar correus (Aplicació)

A partir d'ara, sempre que parlem de Capa d'Aplicació ens referirem a la 4a capa del Model TCP/IP.

2 Sotware de la Capa d'Aplicació

La Capa d'Aplicació pot accedir a la xarxa a través de 2 tipus de programes:

1. Les aplicacions: són els programes que permeten que els usuaris puguin comunicar-se a través de la xarxa. Implementen els protocols necessaris de la Capa d'Aplicació i es poden comunicar directament amb la xarxa. Per exemple: VLC, navegador, client de correu, emule, etc.
2. Els serveis: són els programes de la Capa d'Aplicació que permeten que altres programes puguin accedir als recursos de la xarxa. Per exemple, el servei d'impressió en xarxa.

3 Protocols de la Capa d'Aplicació

Per tal de poder-nos comunicar, cal que els dispositius d'origen i de destí tinguin implementats els mateixos protocols.

Els protocols de la Capa d'Aplicació són un conjunt de normes que permeten que les aplicacions es comuniquin entre si. Les seves funcions són:

- Definir els processos en cadascun dels dispositius finals
- Definir el tipus de missatges
- Definir la sintaxi dels missatges
- Definir el significat dels camps d'informació
- Definir com s'envien els missatges i la resposta que s'espera

- Definir la interacció amb la capa següent

Alguns dels protocols i de serveis de la Capa d'Aplicació que estudiarem són els següents

- DNS: ens permet associar les URL a IP
- HTTP: transfereix arxius HTML
- SMTP i POP: transfereix arxius i adjunts (correu)
- Telnet: ofereix un terminal per accedir de manera remota a servidors i dispositius
- FTP: permet transferir arxius
- DHCP: defineix el *gateway*, la màscara de xarxa i la IP d'un dispositiu

4 Tipus de Xarxes

Des del punt de vista de la Capa d'Aplicació, podem definir dos tipus de xarxes

- Les xarxes Client/Servidor
- Les xarxes Peer-to-Peer (P2P)

Xarxa Client/Servidor

La xarxa Client/Servidor està formada per un dispositiu que demana informació, el **client**, i un dispositiu que respon a aquesta petició, el **servidor**.

Client: té instal·lada l'aplicació necessària per permetre que l'usuari es connecti al servidor adequat segons la petició que vulgui fer.

Servidor: té instal·lat un servei, anomenat *daemon*, que està esperant o escoltant les peticions dels clients. Quan aquest *daemon* rep una petició, es produeix un intercanvi de dades amb el client, depenent del protocol que s'hagi d'executar.

Alguns serveis poden demanar autenticació al client per saber si aquest té permís per fer-ne ús.

Els servidors acostumen a emmagatzemar informació que pot ser compartida per múltiples clients (per exemple, pàgines web, bases de dades, arxius, etc).

Depenent de la petició que faci el client, podem diferenciar entre:

- **Pujada de dades:** transferència de dades des del client al servidor
- **Descàrrega de dades:** transferència de dades des del servidor al client

Connexions simultànies

Una única aplicació client pot utilitzar diferents serveis. Per exemple, el navegador et permet visualitzar pàgines web (servei web) i també et permet imprimir-les (servei d'impressió en xarxa).

Una única sol·licitud o petició de dades pot implicar l'intercanvi de múltiples missatges entre el client i el servidor.

Un mateix servidor pot estar atenent a múltiples clients alhora.

El tractament de totes aquestes peticions ha de ser simultani i individual.

Xarxa Peer-to-Peer (P2P)

En una xarxa Peer-to-Peer tots els dispositius connectats poden compartir recursos entre ells. És a dir, cada dispositiu final, o *peer*, pot actuar tant de servidor com de client.

En les xarxes Peer-to-Peer el concepte de servidor dedicat, tal i com s'entén en les xarxes Client/Servidor, no existeix. Els recursos estan descentralitzats.

Un exemple de xarxa Peer-to-Peer pot ser una xarxa domèstica o dos ordinadors connectats a Internet que comparteixen arxius a través d'un software determinat.

5 Aplicacions Peer-to-Peer

Una aplicació Peer-to-Peer (P2P) permet que un dispositiu actui com a client i com a servidor dins d'un mateix procés de comunicació. D'aquesta manera, cada *host* té una interfície d'usuari (client) i executa un servei (servidor) dins de la mateixa aplicació.

Hi ha aplicacions P2P híbrides que treballen de la següent manera:

1. Els dispositius es connecten a un servidor centralitzat que té un índex que indica com accedir als recursos. És a dir, l'índex indica quins *hosts* tenen quins recursos i/o informació.
2. Un cop obtingut l'índex, cada dispositiu es connecta al dispositiu corresponent segons la informació o el recurs que vulgui.

Per exemple, l'emule i les aplicacions que s'utilitzen per descarregar els torrent serien aplicacions híbrides.

Les aplicacions P2P es poden utilitzar en xarxes P2P, en xarxes Client/Servidor i a Internet.

6 Protocols en detall

Servei i Protocol DNS

Domain Name Service (DNS): és el servei que s'encarrega de traduir els noms de domini (les URL) en IP. Associat a aquest servei, tenim el protocol DNS, que utilitza els ports TCP/UDP 53. Tant el servei com el protocol DNS treballen en xarxes Client/Servidor.

Permet canviar les IP dels servidors de manera transparent (l'usuari no se n'assabenta).

El servei DNS utilitza un conjunt de servidors distribuïts per tal de poder fer la traducció entre el domini a la IP corresponent.

Com ja s'ha dit, el servei DNS és un servei Client/Servidor. No obstant això, en aquest cas, el client no té una aplicació amb interfície gràfica, sinó que executa un servei que, cada cop que connectem amb un domini, envia una petició al servidor de DNS per obtenir la IP corresponent. El client DNS s'anomena *resolució DNS*.

Quan configurem un dispositiu, hem de definir un conjunt de servidors de DNS (normalment això ho fa l'ISP, el proveïdor de serveis d'internet) per tal que el client DNS els pugui enviar les peticions de resolució.

Per veure quins servidors DNS tenim configurats

```
$ cat /etc/resolv.conf
```

Per fer una resolució manual

```
$ nslookup DOMINI
```

El *daemon* del servidor DNS s'anomena *named*.

Cada servidor DNS té un conjunt de *registres de recursos* de tal manera que cada registre conté:

1. Nom del domini
2. IP
3. Tipus de registre: A (dispositiu final), NS (servidor de noms), CNAME (nom per diferenciar diversos serveis amb la mateixa IP però diferents registres)

Quan un client fa una petició de resolució

1. El servidor DNS mira si té el registre adequat. Si és així, retorna la informació al client.
2. Si no té el registre, ell mateix fa la sol·licitud a altres servidors DNS.
3. Quan un servidor DNS troba el registre correcte envia la informació al 1r servidor DNS.
4. El primer servidor DNS guarda aquest registre a la catxé i respon al client.

Si el client torna a buscar el mateix nom, el servidor l'agafa de la catxé i així disminueix el trànsit de paquets DNS.

El format d'un missatge DNS mostra la següent estructura:

- Capçalera
- Pregunta: el domini que busquem
- Resposta: registres que donen resposta a la pregunta
- Autoritat: registres que apunten a una autoritat
- Addicional: registres amb informació extra

Podem veure aquest format executant la comanda

```
$ dig google.com
```

El servei DNS utilitza una jerarquia en forma d'arbre per tal de poder resoldre els noms. Així doncs, si tenim les adreces *www.escoladeltreball.org* i *correu.escoladeltreball.org*: *org* és el Domini de Nivell Superior; *escoladeltreball* és el Subdomini; *www* i *correu* són la Màquina.

Els servidors DNS també estan jerarquitats, de tal manera que cada domini o subdomini té una o més zones d'autoritat. Al capdamunt d'aquesta jerarquia hi ha els servidors arrel, els quals tenen la informació per resoldre els dominis de 1r i 2n nivell.

Servei WWW i Protocol HTTP

L'HTTP utilitza el port TCP 80 i és un protocol Client/Servidor.

L'aplicació client per fer ús del servei WWW és el navegador.

Quan s'escriu una URL al navegador, aquest estableix una connexió amb el servidor web corresponent. Ambdós dispositius utilitzen el protocol HTTP per comunicar-se.

Els navegadors són capaços d'interpretar molts formats de dades, com per exemple, el text pla o l'HTML. No obstant això, per altres formats necessiten augmentar les seves capacitats mitjançant plugins.

Donada una URL, per exemple *http://www.cisco.com/web-server.html*, el navegador interpreta que *http* és el protocol que ha d'utilitzar per comunicar-se amb el servidor; *www.cisco.com* és el servidor i *web-server.html* és el recurs del servidor que es vol obtenir. Aleshores, els passos que segueix són els següents:

1. Consulta el DNS per conèixer la IP del servidor indicat a la URL
2. Utilitzant el protocol HTTP, envia una sol·licitud GET al servidor per tal d'obtenir el fitxer web *web-server.html*
3. El servidor envia el codi html
4. El navegador interpreta i presenta la pàgina web

El protocol HTTP és un protocol de petició/resposta. Té diferents tipus de peticions

- GET: el client envia la petició GET per tal d'obtenir les pàgines des del servidor. El servidor respon amb un HTTP 1.1 200 OK i l'arxiu (o un missatge d'error)
- POST i PUT: serveixen perquè el client pugui carregar dades al servidor (per exemple, dades d'un formulari, un arxiu...).

Amb el protocol HTTP les dades viatgen en clar. Per tal que viatgin segures es necessita el protocol HTTPS, el qual permet encriptar les dades i demanar autenticació. L'HTTPS afegeix normes addicionals per passar de la capa aplicació a la de transport.

Servei de Correu i Protocols SMTP i POP

Els protocols SMTP i POP són protocols Client/Servidor.

L'aplicació client per al servei de correu és el client de correu electrònic, el qual permet **enviar missatges**, mitjançant SMTP, i **rebre missatges**, mitjançant POP.

El servidor de correu necessita 2 processos

- MTA (Agent de Transferència de Correu): envia els correus
- MDA (Agent d'Entrega de Correu): entrega els correus

Un MTA pot rebre un correu des del client (MUA) o des d'un altre MTA. Si el correu que rep és per algun usuari que té la bústia en aquell mateix servidor, l'MTA envia el correu a l'MDA. Si no és així, l'MTA reenvia el correu a l'MTA del servidor corresponent.

EL protocol POP permet enviar el correu des de l'MDA al client (MUA).

El protocol SMTP permet enviar el correu del client a l'MTA o entre els processos MTA de diversos servidors. Les seves funcions són, entre altres:

- Iniciar sessió
- Transacció de correu
- Reenviament
- Verificació de les bústies

Algunes comandes del protocol

- HELO/EHLO: identifica el client
- MAIL FROM: emissor
- RCPT TO: receptor
- DATA: missatge

Protocol FTP

L'FTP és el Protocol de Transferència d'Arxius.

El client FTP llença peticions per carregar i descarregar arxius a un servidor que té instal·lat el servei FTP i el daemon FTP (FTPD) actiu.

Tota petició FTP requereix de 2 connexions entre el client i el servidor.

1. Una connexió a través del port TCP 21 per tal de realitzar el control del trànsit i permetre l'enviament de comandes des del client i respostes del servidor
2. Una connexió a través del port TCP 20 per fer la transferència real de l'arxiu. Aquesta connexió s'obre i es tanca per cada arxiu

Protocol DHCP

El DHCP és el Protocol de Configuració Dinàmica de Hosts.

Aquest protocol permet que els dispositius d'una xarxa obtinguin de manera automàtica la configuració d'aquesta xarxa. És a dir, n'obtenen la IP, la Màscara de Xarxa, el Gateway i els DNS. Dit d'una altra manera, el servidor DHCP permet configurar, de manera automàtica, la xarxa d'un dispositiu.

Quan un dispositiu demana una IP, el servidor DHCP n'hi assigna una dins d'un rang que té preconfigurat (pool). Cada cop que el dispositiu es desconnecta de la xarxa, la IP *torna* a la pool del DHCP per tal que pugui ser utilitzada per algú altre.

Normalment, es combina el direccionament estàtic amb el DHCP. Per exemple, el DHCP s'utilitza pels dispositius generals. En canvi, els gateways, els switchos, els servidors i les impressores tenen direccionament estàtic.

Per trobar el DHCP:

```
$ grep dhcp-server-identifier /var/lib/dhcp/dhclient.leases
```

Si no s'utilitza DHCP, la configuració de la xarxa s'ha de fer manualment en cada dispositiu.

Quan un client DHCP es connecta a la xarxa, envia una petició per *descobrir* el servidor DHCP de la xarxa. Quan el servidor rep la petició, respon amb una oferta DHCP que conté la configuració de la xarxa (IP, MX, GW i DNS). Si hi ha més d'un servidor DHCP, el client pot rebre moltes configuracions de xarxa. Per això, respon enviant en broadcast el servidor que ha escollit i la configuració que se li ha assignat. SI tot ha anat bé, el servidor seleccionat retorna un missatge de **ACK DHCP**. Però si, per alguna raó, la configuració de xarxa inicial ja no està disponible, el servidor envia un **NAK DHCP**. Aleshores, el procés ha de començar des del principi amb el **DHCP DISCOVER**. Si la configuració està a punt d'expirar, el client la pot renovar amb un *DHCP REQUEST*.

DHCP assegura que no hi ha IPs repetides.

Protocol SMB

SMB és el protocol de Bloc de Missatges del Servidor. És un protocol Client/Servidor, com tots els que hem vist fins ara, i serveix per compartir arxius. Va ser desenvolupat per IBM per tal que els dispositius poguessin descobrir l'estructura dels recursos de xarxa compartits (directoris, arxius, impressores...)

És un protocol de petició/resposta que només necessita una connexió entre el client i el servidor a llarg plaç. Aquesta connexió permet que el client pugui accedir als recursos del servidor com si fossin locals.

SMB és un protocol inherent dels dispositius de Microsoft. No obstant això, els dispositius Apple també l'accepten i els dispositius UNIX/Linux en tenen una versió anomenada SAMBA.

SMB permet

1. Iniciar, autenticar i tancar sessions

2. Controlar l'accés als recursos
3. Permetre l'enviament de missatges entre dispositius

Protocol Gnutella

El protocol Gnutella permet que les aplicacions P2P puguin compartir arxius.

Té 5 tipus de missatges

- ping: per descobrir els peers
- pong: resposta al ping
- consulta: per trobar un arxiu
- query hit: resposta a la consulta
- push: sol·licitud de descàrrega

Protocol Telnet

Telnet és el protocol que permet accedir de manera remota a altres dispositius. És un dels protocols TCP/IP més antics i només permet accés remot a través del terminal de text.

La connexió Telnet s'anomena Sessió o Connexió de Terminal Virtual (VTY).

És un protocol Client/Servidor

Telnet no permet encriptació i els missatges viatgen en clar. SI es necessita seguretat, aleshores es fa servir el protocol SSH (Secure Shell).