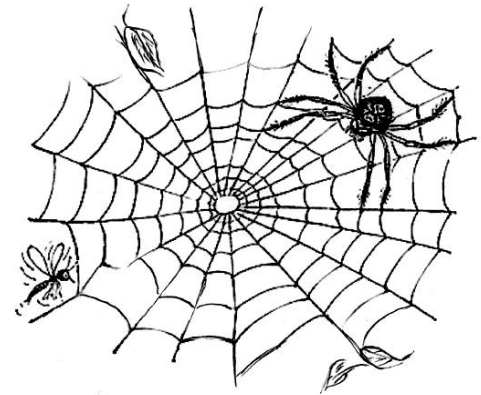


Modulo 8

Protocolos de Aplicación

Inter
Net
Working



Índice

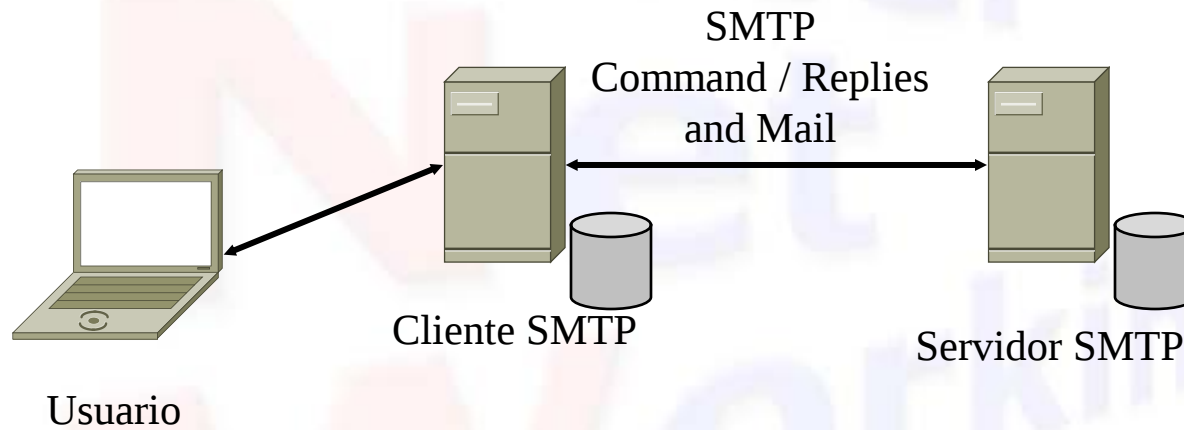
1. SMTP/POP3
2. DNS
3. HTTP
4. WAP
5. Protocolos de Señalización IP: SIP, MGCP
6. RTP

SMTP/POP3

SMTP

- SIMPLE MAIL TRANSFER PROTOCOL
- RFC 821
- Protocolo para el envío de e-mail
- SMTP establece un canal de transmisión bidireccional.
- El servido SMTP puede ser el destino final o uno intermedio.
- El cliente genera comandos SMTP al servidor que son respondidos para inyectarle los correos

Funcionamiento



Ejemplo de Conexión

- MAIL FROM: <jsandova@ing.uchile.cl>
- 250 OK
- RCPT TO: <el64e@ing.uchile.cl>
- 250 OK
- RCPT TO: <superman@ing.uchile.cl>
- 550 No such user here
- RCPT TO: <el65l@ing.uchile.cl>
- 250 OK
- DATA
- 354 Start mail input; end with .
- Blah blah blah...
- ...etc. etc. etc.
- .
- 250 OK

Forwarding

- A veces la información de destino es incorrecta pero el servidor “sabe” cual es el destino correcto.

Inter
Net
Working

Ejemplo de Forward

```
MAIL FROM:<jsandova@ieee.org>
```

```
251 User not local; will forward to <jsandova@ing.uchile.cl>
```

En este caso el servidor es responsable de redirigir el mensaje al destino correcto

o

```
MAIL FROM:<jsandova@ieee.org>
```

```
251 User not local; please try <jsandova@ing.uchile.cl>
```

En este caso el server rechaza el correo y el cliente debe reenviarlo al destino correcto.

VERIFYING AND EXPANDING

- SMTP provee comandos para verificar un nombre de usuario o expandir una lista de mail.
 - Con el comando VRFY se puede enviar el nombre de usuario y recibir el nombre completo incluido su e-mail
 - Con el comando EXPN se puede enviar el nombre de una lista de distribución e correo y recibir el nombre completo de los usuarios y e-mail de los miembros de la lista.

Ejemplo de Verifying

VRFY jsandova

250 Jorge Sandoval <jsandova@ing.uchile.cl>

VRFY jsandova

251 User not local; will forward to <jsandova@internetworking.cl>

VRFY el643

550 String does not match anything.

VRFY el643

551 User not local; please try

VRFY Gourzenkyinplatz

553 User ambiguous.

Ejemplo de Expanding

```
EXPN el64e
250 Jorge Sandoval <jsandova@ing.uchile.cl>
250 Redes de Computadores <el64e@ing.uchile.cl>
250 Laboratorio LABNET <labnet@gorrion.die.uchile.cl>250
```

O

```
EXPN el64e
502 5.7.0 Sorry, we do not allow this operation
```

Algunos Comandos

HELLO <hostname>

MAIL FROM: <sender> [<parameters>]

RCPT TO: <recipient> [<parameters>]

DATA

RSET

VERFY

EXPN

HELP

NOOP

QUIT

TURN

Reply Codes By Function Groups

- 500 Syntax error, command unrecognized
 - [This may include errors such as command line too long]
- 501 Syntax error in parameters or arguments
- 502 Command not implemented
- 503 Bad sequence of commands
- 504 Command parameter not implemented

- 211 System status, or system help reply
- 214 Help message [Information on how to use the receiver or the meaning of a particular non-standard command; this reply is useful only to the human user]

- 220 <domain> Service ready
- 221 <domain> Service closing transmission channel
- 421 <domain> Service not available, closing transmission channel
 - [This may be a reply to any command if the service knows it must shut down]

Reply Codes By Function Groups (Cont)

- 250 Requested mail action okay, completed
- 251 User not local; will forward to <forward-path>
- 450 Requested mail action not taken: mailbox unavailable
 - [E.g., mailbox busy]
- 550 Requested action not taken: mailbox unavailable
 - [E.g., mailbox not found, no access]
- 451 Requested action aborted: error in processing
- 551 User not local; please try <forward-path>
- 452 Requested action not taken: insufficient system storage
- 552 Requested mail action aborted: exceeded storage allocation
- 553 Requested action not taken: mailbox name not allowed
 - [E.g., mailbox syntax incorrect]
- 354 Start mail input; end with <CRLF>.<CRLF>
- 554 Transaction failed

Ejemplo de envío de un e-mail

```
220 BBN-UNIX.ARPA Simple Mail Transfer Service Ready
HELO USC-ISIF.ARPA
250 BBN-UNIX.ARPA
MAIL FROM:<Smith@USC-ISIF.ARPA>
250 OK
RCPT TO:<Jones@BBN-UNIX.ARPA>
250 OK
RCPT TO:<Green@BBN-UNIX.ARPA>
550 No such user here
RCPT TO:<Brown@BBN-UNIX.ARPA>
250 OK
DATA
354 Start mail input; end with <CRLF>.<CRLF>
Blah blah blah...
...etc. etc. etc.
.
250 OK
QUIT
221 BBN-UNIX.ARPA Service closing transmission channel
```

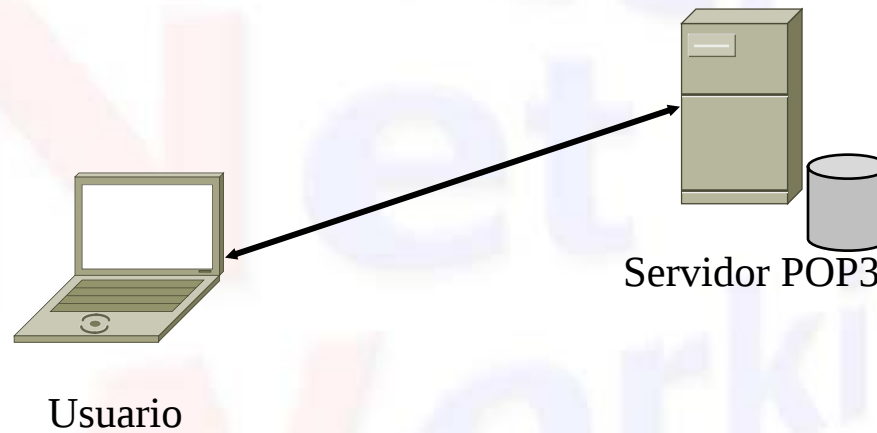
Ejemplo 2 de envío de un e-mail abortado

```
220 MIT-Multics.ARPA Simple Mail Transfer Service Ready
HELO ISI-VAXA.ARPA
250 MIT-Multics.ARPA
MAIL FROM:<Smith@ISI-VAXA.ARPA>
250 OK
RCPT TO:<Jones@MIT-Multics.ARPA>
250 OK
RCPT TO:<Green@MIT-Multics.ARPA>
550 No such user here
RSET
250 OK
QUIT
221 MIT-Multics.ARPA Service closing transmission channel
```

POP3

- Post Office Protocol - Versión 3
- RFC 1939
- Protocolo para lectura de e-mail
- POP3 establece un canal de transmisión bidireccional.
- El cliente genera comandos POP3 al servidor que son respondidos para la lectura los correos

Funcionamiento



Validación

- Ingreso Exitoso

```
+OK Qpopper (version 4.0.3) at arrayan starting.  
USER jsandova  
+OK Password required for jsandova.  
PASS mipassword  
+OK jsandova has 13 visible messages (0 hidden) in  
23450 octets.
```

- Ingreso fallido

```
+OK Qpopper (version 4.0.3) at arrayan starting.  
USER jsandova  
+OK Password required for jsandova  
PASS 1234  
-ERR [AUTH] Password supplied for "jsandova" is  
incorrect.
```

Mensajes en el Buzón

- **Al Ingreso**

```
+OK Qpopper (version 4.0.3) at arrayan starting.  
USER jsandova  
+OK Password required for jsandova.  
PASS mipassword  
+OK jsandova has 13 visible messages (0 hidden) in  
23450 octets.
```

- **Comando STAT**

Indica numero de mensajes y largo total en byte

Ejemplo

STAT

```
+OK 13 23450
```

Información de mensajes en el Buzón

- Comando LIST

- Indica largo total en byte de cada uno de los mensajes
- Ejemplo

LIST

+OK 10 visible messages (20456 octets)

1 2585

2 2074

3 3374

4 2271

5 2225

6 1464

7 1074

8 1283

9 1678

10 2428

Lectura de un Mensaje

- La Lectura se realiza utilizando el comando RETR
- Ejemplo:

```
RETR 1
+OK 538 octets
Date: 16 May 2002 12:10:57 -0400
From: Mail System Internal Data <MAILER-DAEMON@ing.uchile.cl>
Subject: DON'T DELETE THIS MESSAGE -- FOLDER INTERNAL DATA
Message-ID: <1021565457@ing.uchile.cl>
X-IMAP: 0967564211 0000000751
X-UIDL: ='h"!394!!_aJ!!'GV!!
Status: RO
```

This text is part of the internal format of your mail folder, and is not a real message. It is created automatically by the mail system software. If deleted, important folder data will be lost, and it will be re-created with the data reset to initial values.

.

Lectura Parcial de un Mensaje

- La lectura parcial de un mensaje se realiza utilizando el comando RETR
- Ejemplo (leo el encabezado y 5 primeras líneas):

```
TOP 1 5
+OK Message follows
Date: 16 May 2002 12:10:57 -0400
From: Mail System Internal Data <MAILER-DAEMON@ing.uchile.cl>
Subject: DON'T DELETE THIS MESSAGE -- FOLDER INTERNAL DATA
Message-ID: <1021565457@ing.uchile.cl>
X-IMAP: 0967564211 0000000751
X-UIDL: ='h"!394!!_aJ!!'GV!!
Status: RO
```

This text is part of the internal format of your mail folder, and is not a real message. It is created automatically by the mail system software. If deleted, important folder data will be lost, and it will be re-created with the data reset to initial values.

.

Eliminación de un Mensaje

- La Eliminación se realiza utilizando el comando DELE

- Ejemplo:

```
DELE 1
```

```
+OK message 1 deleted
```

Ejemplo de Conexión

```
+OK Qpopper (version 4.0.3) at arrayan starting.  
USER jsandova  
+OK Password required for jsandova.  
PASS mipassword  
+OK jsandova has 13 visible messages (0 hidden) in 23450 octets.  
STAT  
+OK 13 23450  
RETR 1  
+OK 538 octets  
Date: 16 May 2002 12:10:57 -0400  
From: Mail System Internal Data <MAILER-DAEMON@ing.uchile.cl>  
Subject: DON'T DELETE THIS MESSAGE -- FOLDER INTERNAL DATA  
Message-ID: <1021565457@ing.uchile.cl>  
X-IMAP: 0967564211 0000000751  
X-UIDL: ='h"!394!!_aJ!!'GV!!  
Status: RO
```

This text is part of the internal format of your mail folder, and is not a real message. It is created automatically by the mail system software. If deleted, important folder data will be lost, and it will be re-created with the data reset to initial values.

```
.  
DELE 1  
+OK message 1 deleted  
QUIT  
+OK Pop server at arrayan signing off.
```

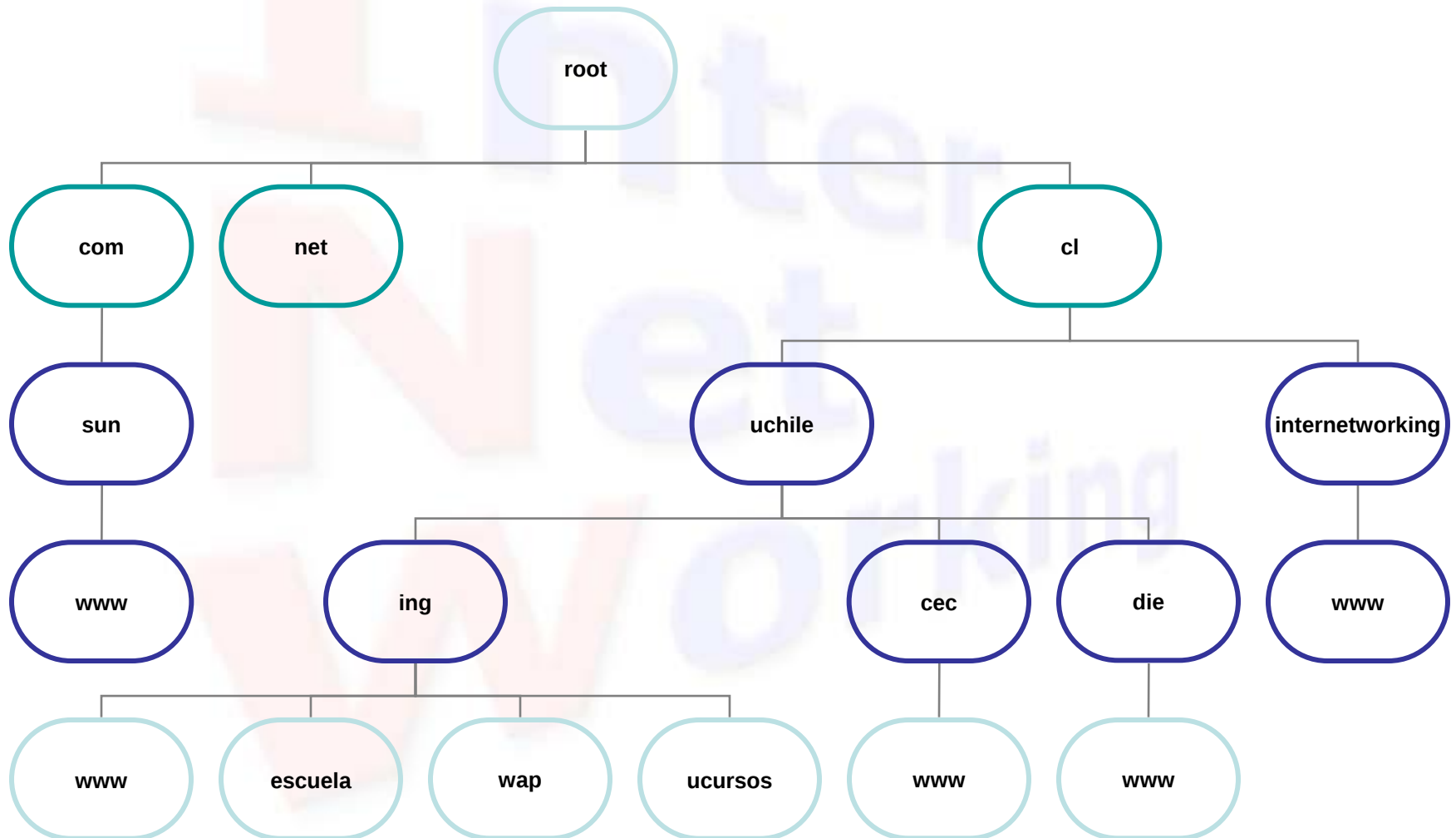
Inter Net Working

DNS

DNS

- Aplicación distribuida que permite asociar un nombre a una dirección IP.
- Posee una estructura jerárquica.
- Delegación de autoridad
- BIND. Berkeley Internet Name Domain
- Conceptos:
 - Servidor Primario. Actualiza
 - Servidor Secundario. Informa
 - Servidor Cache. Informa sin autoridad

Estructura jerárquica



Delegación de Autoridad

- Se delegan a subdominios definiendo un record NS.
- Las preguntas sobre hosts de un subdominio son derivadas al DNS del subdominio

Inter
Net
Working

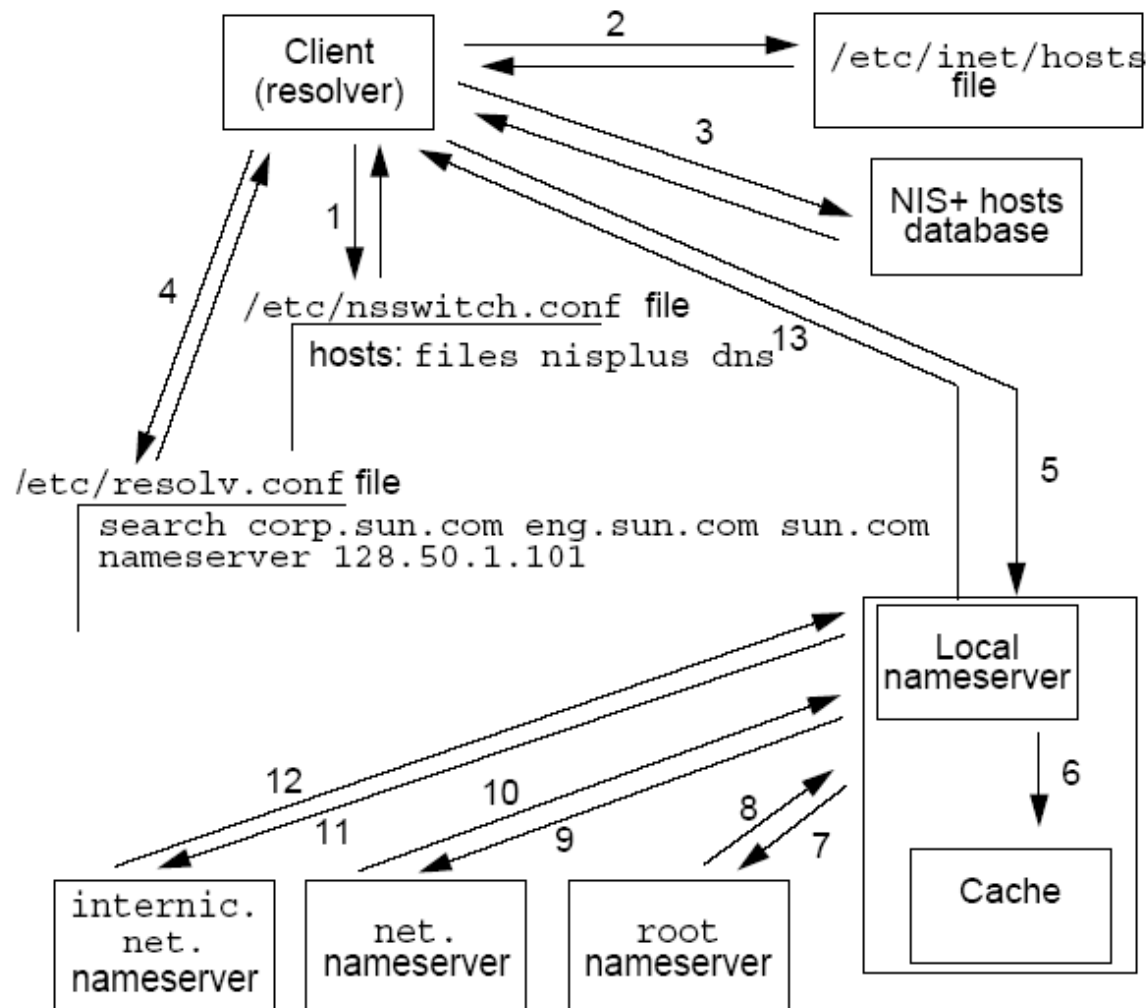
Tipos de Registros

- **A** Address Record. Para definir un host
- **CNAME** Canonical Name. Usado para definir una alias de un host.
- **NS** Name Service. Para definir un DNS
- **MX** Mail Exchange. Para definir un servidor SMTP
- **PTR** para definir un dominio inverso
- **SOA** Start of Authority

SOA Start of Authority

- **Serial Number.** Habitualmente en formato yyyymmddhh para indicar modificación. Se asume una nueva versión si este número se incrementa.
- **Refresh.** Cada cuanto tiempo el secundario refresca.
- **Retry.** Cada cuanto tiempo reintenta si la actualización del secundario fracaza.
- **Expire.** Indica cuanto tiempo puede estar la copia en el secundario sin refrescar.
- **TTL.** Tiempo que permanecerán los registros en los caches de los DNSs.

Ejemplo de resolución de nombres



Inter Net Working

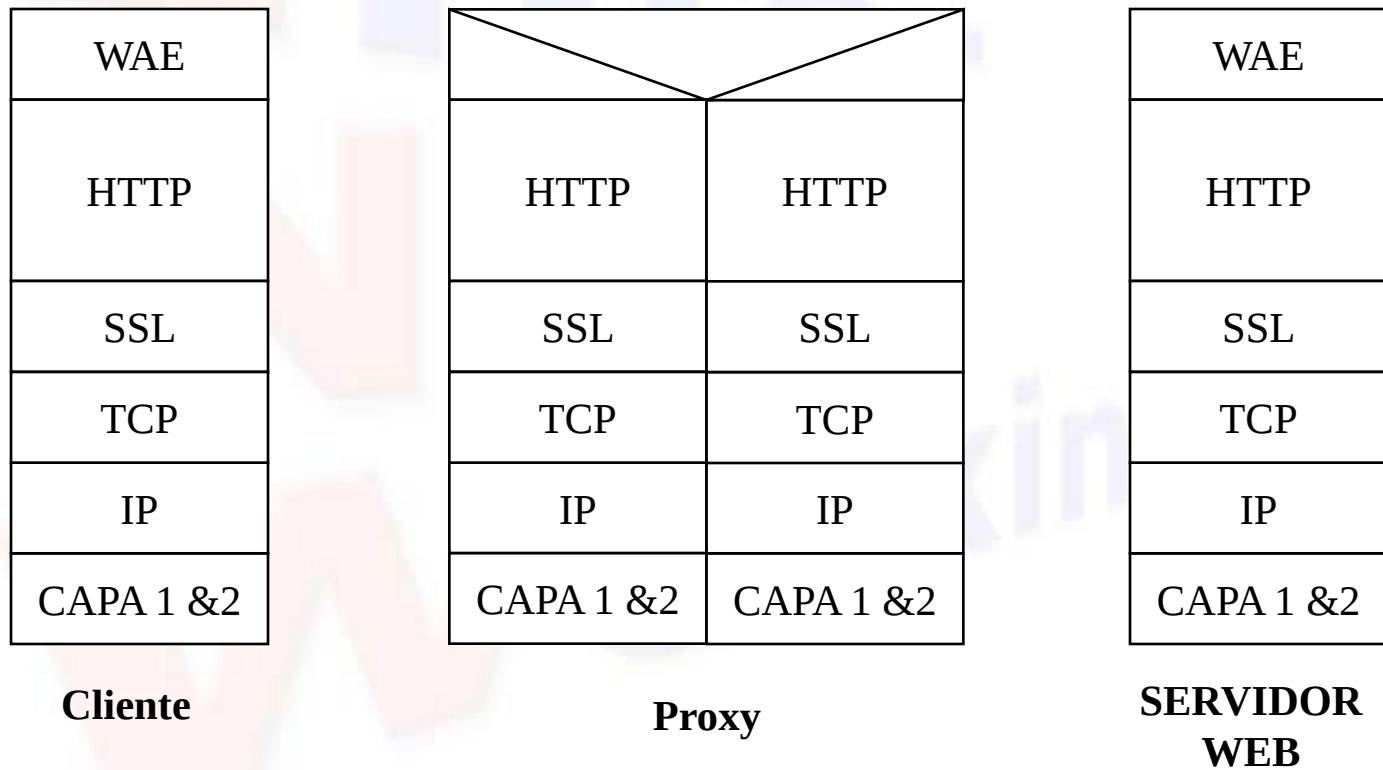
HTTP

HTTP

- HyperText Transfer Protocol
- RFC 2616
- Protocolo pregunta-respuesta

Inter
Net
Working

Stack de protocolos



URL

- Formato URL
 - `http_URL = "http:" "//" host [ ":" port ] [ abs_path [ "?" query ] ]`

Internet
Networking

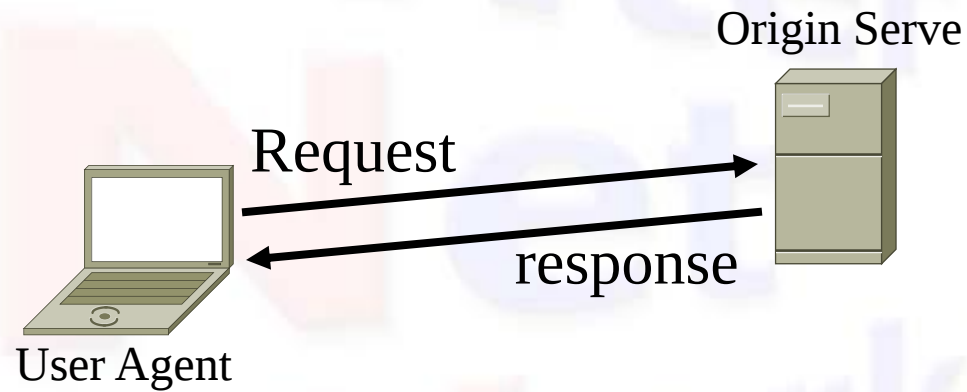
Ejemplo

```
%telnet www.cec.uchile.cl 80
GET http://www.cec.uchile.cl/~jsandova HTTP/1.0
```

```
HTTP/1.1 301 Moved Permanently
Date: Thu, 23 May 2002 15:47:19 GMT
Server: Apache/1.3.11 (Unix)
Location: http://www.cec.uchile.cl/~jsandova/
Connection: close
Content-Type: text/html; charset=iso-8859-1
```

```
<!DOCTYPE HTML PUBLIC "-//IETF//DTD HTML 2.0//EN">
<HTML><HEAD>
<TITLE>301 Moved Permanently</TITLE>
</HEAD><BODY>
<H1>Moved Permanently</H1>
The document has moved <A
  HREF="http://www.cec.uchile.cl/~jsandova/">here</A>.<P>
</BODY></HTML>
```

Ejemplo de Conexión

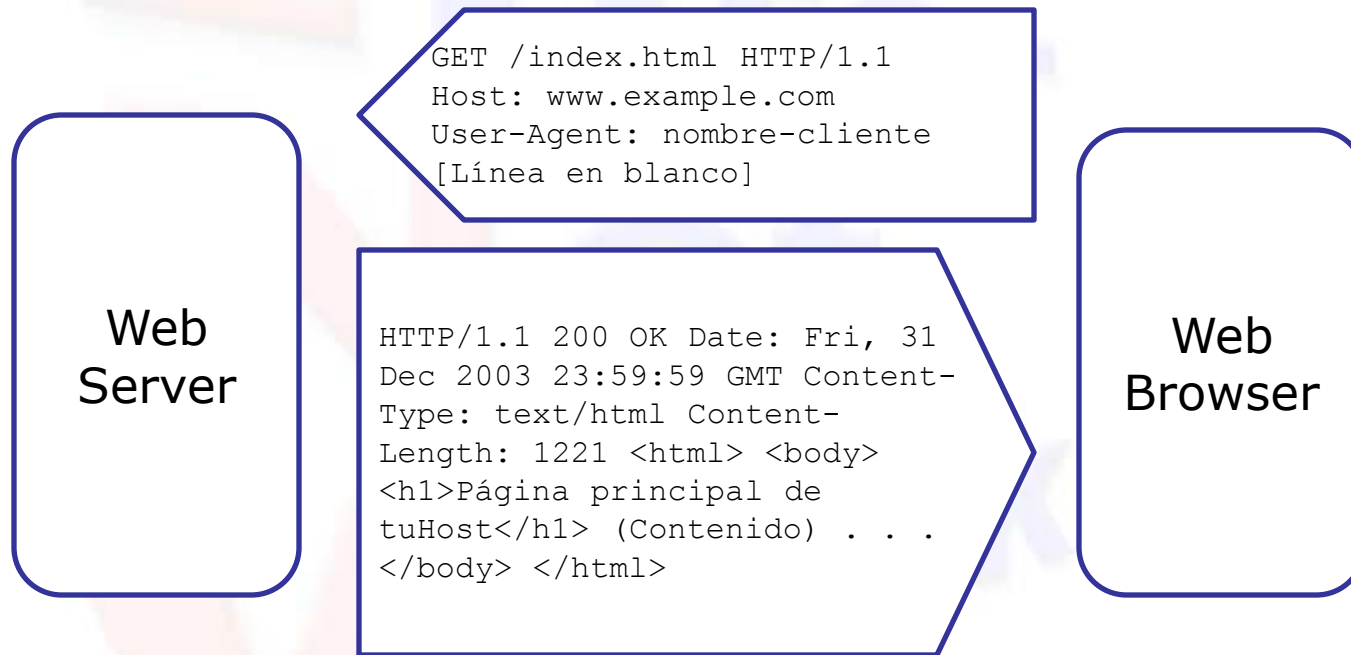


Status Code Definitions

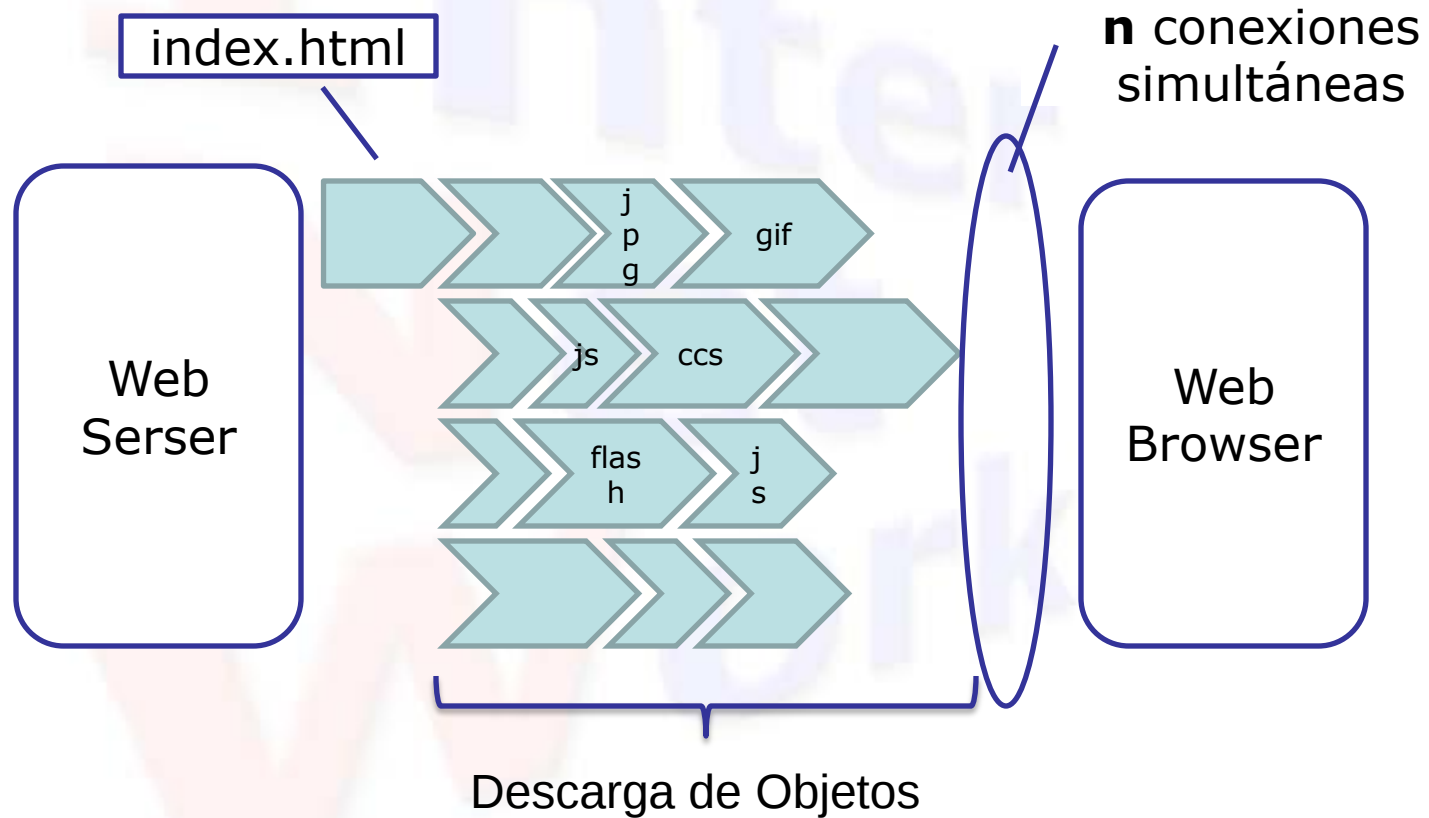
- Informational 1xx
- Successful 2xx
- Redirection 3xx
- Client Error 4xx
- Server Error 5xx

Inter
Net
Working

Ejemplo



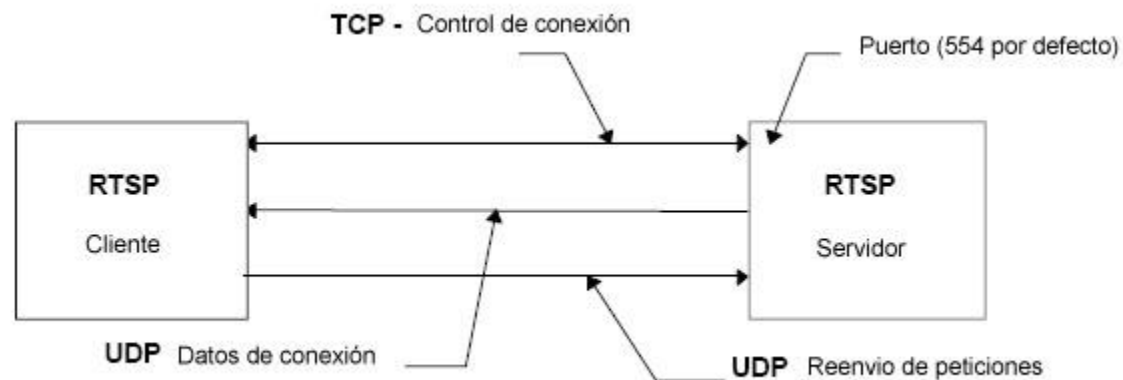
Funcionamiento



RTSP

Real Time Streaming Protocol

- RFC-2326
- Es un protocolo no orientado a la conexión que utiliza:
 - TCP para el control de la conexión y
 - UDP para la transmisión de datos de audio y video



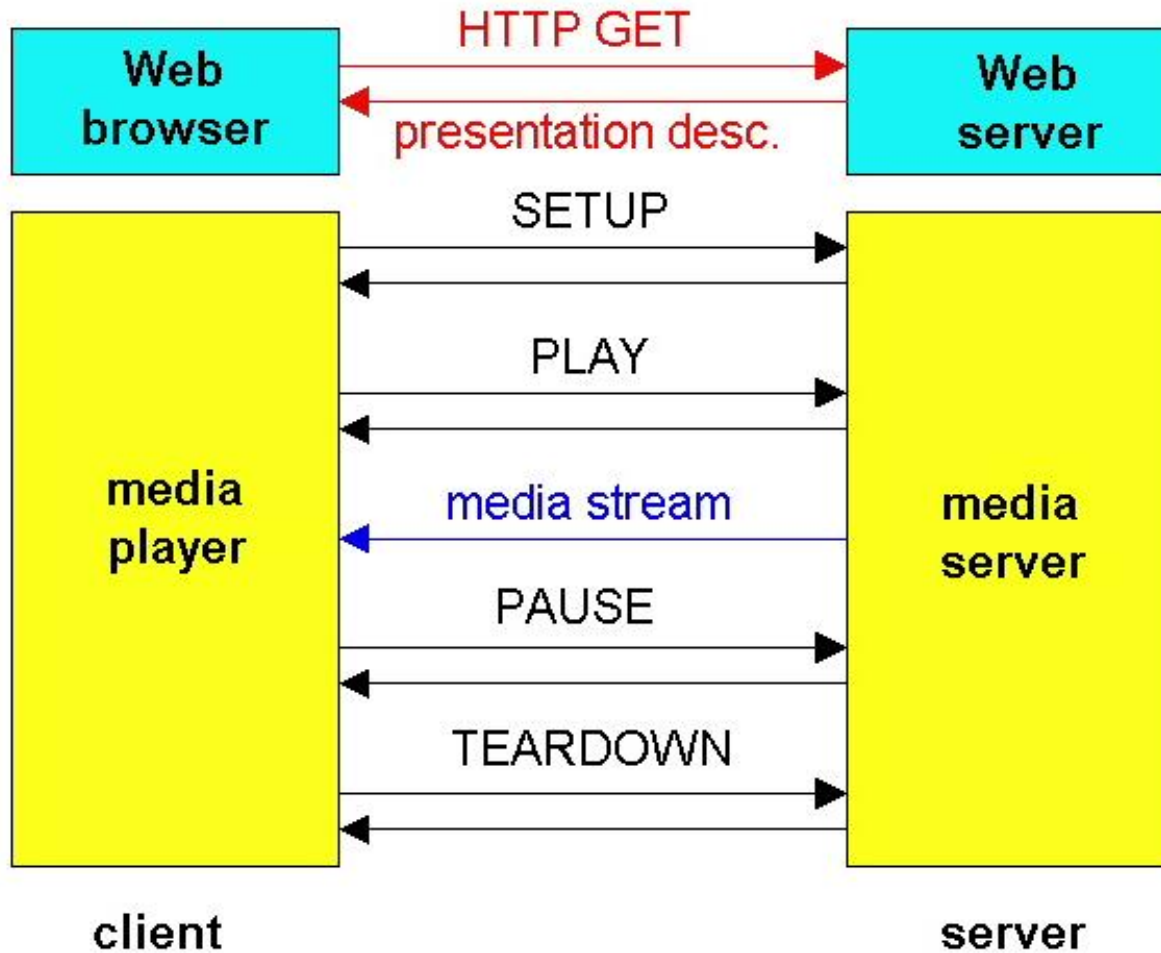
Propiedades

- RTSP tiene las siguientes propiedades:
- Extensible: nuevos métodos y parámetros pueden ser fácilmente añadidos al RTSP
- Seguro: RTSP reutiliza mecanismos de seguridad web ya sea a los protocolos de transporte (TLS) o dentro del mismo protocolo. Todas las formas de autenticación HTTP ya sea básica o basada en resumen son directamente aplicables.
- Independiente del protocolo de transporte: RTSP puede usar indistintamente protocolos de datagrama no fiables (UDP) o datagramas fiables (RDP, no muy extendido) o un protocolo fiable orientado a conexión como el TCP.
- Capacidad multi-servidor: Cada flujo multimedia dentro de una presentación puede residir en servidores diferentes, el cliente automáticamente establece varias sesiones concurrentes de control con los diferentes servidores, la sincronización la lleva a término la capa de transporte.
- Control de dispositivos de grabación: El protocolo puede controlar dispositivos de grabación y reproducción (p.ej cámaras IP RTSP).
- Adecuado para aplicaciones profesionales: RTSP soporta resolución a nivel de frame mediante marcas temporales SMPTE para permitir edición digital.

Stack de protocolos

Application Control Commands + SDP	Audio Data, Video Data, Sender/Receiver Reports
RTSP	RTP/RTCP
TCP	UDP
IP	
Radio Link/Data Link	
Physical Layer	

Operación



El control de la comunicación se realiza con peticiones basadas en http

DESCRIBE

- Este método obtiene una descripción de una presentación o del objeto multimedia apuntado por una URL RTSP situada en un servidor
- Cliente -> Servidor

```
DESCRIBE rtsp://unservidor.com/uncontingut RTSP/1.0  
Accept: application/sdp, application/rtsl, application/ mheg
```

- Servidor -> Cliente

```
RTSP/1.0 200 OK  
Content-Type: application/sdp  
Content-Length: 376  
i=Descripció del contingut;  
m=audio 3456 RTP/AVP 0  
m=video 2232 RTP/AVP 31
```

SETUP

- Especifica como será transportado el flujo de datos

- Cliente -> Servidor

```
SETUP rtsp://example.com/foo/bar/baz.rm RTSP/1.0  
Transport: RTP/AVP;unicast;client_port=4588-4589
```

- Servidor -> Cliente

```
RTSP/1.0 200 OK  
Session: 47112344  
Transport: RTP/AVP;unicast;  
client_port=4588-4589;server_port=6256-6257
```

PLAY

- Solicitud para que el servidor comience a enviar datos de los flujos especificados con SETUP
- Cliente -> Servidor

```
PLAY rtsp://unservidor.com/audio RTSP/1.0  
Session: 12345678
```

PAUSE

- Detiene temporalmente uno o todos los flujos, de manera que puedan ser recuperados con un PLAY posteriormente
- Cliente -> Servidor

```
PAUSE rtsp://unservidor.com/video1 RTSP/1.0  
Session: 12345678
```

- Servidor -> Cliente

```
RTSP/1.0 200 OK
```

TEARDOWN

- Detiene la entrega de datos para la URL indicada liberando los recursos asociados

- Cliente -> Servidor

```
TEARDOWN rtsp://unservidor.com/video1 RTSP/1.0  
Session: 12345678
```

- Servidor -> Cliente

```
RTSP/1.0 200 OK
```

RTSP Exchange Example

C: SETUP rtsp://audio.example.com/twister/audio RTSP/1.0
Transport: rtp/udp; compression; port=3056; mode=PLAY

S: RTSP/1.0 200 1 OK
Session 4231

C: PLAY rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0
Session: 4231
Range: npt=0-

C: PAUSE rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0
Session: 4231
Range: npt=37

C: TEARDOWN rtsp://audio.example.com/twister/audio.en/lofi RTSP/1.0
Session: 4231

S: 200 3 OK

RFCs relacionados

- RFC 2326: Real-Time Streaming Protocol (RTSP)
- RFC 2327: Session Description Protocol (SDP)
- RFC 2543: Session Initiation Protocol (SIP)
- RFC 2974: Session Announcement Protocol (SAP)
- RFC 3108: SDP for ATM Bearer Connections
- RFC 3259: A Message Bus for Local Coordination
- RFC 3264: An Offer/Answer Model with Session Description Protocol (SDP)
- RFC 3266: Support for IPv6 in Session Description Protocol (SDP)
- RFC 3388: Grouping of Media Lines in the Session Description Protocol (SDP)
- RFC 3524: Mapping of Media Streams to Resource Reservation Flows

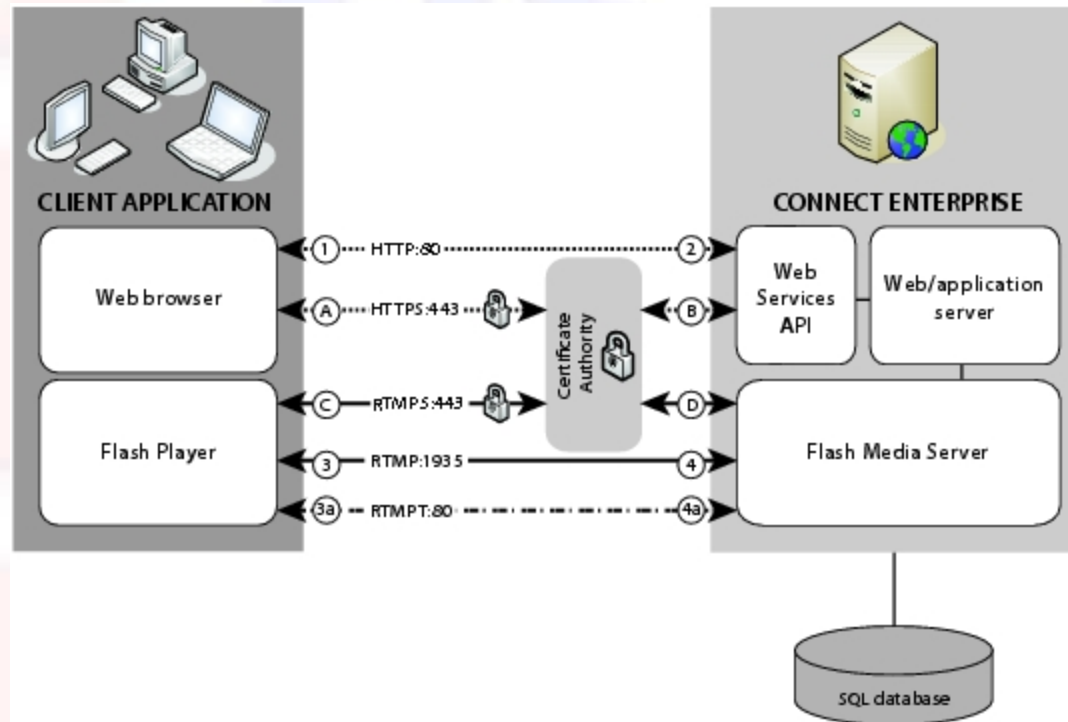
Inter Net Working

RTMP

RTMP

- Real Time Messaging Protocol
- Propietario de Adobe
- Protocolo para comunicación en cliente servidor de video flash
- Usa los siguientes puertos:
 - **RTMP** (default port 1935)
 - **RTMPT** (tunnel via http - default port of 80)
 - **RTMPS** (tunnel via https - default port of 443)

Funcionamiento



Ejemplo de Conexión

Datos Video:

Duración: 3:21

Volumen: 7.677.432 Bytes (~35KB por cada segundo de video)

Datos Descarga:

BAM

Duración: ~192 segundos

Throughput: ~400kbps

Datos Conexión:

BAM

Window: 17.640 Bytes

RTT: ~342 [ms]

Throughput Teórico Máximo: ~ 412 [kbps]

LAN

Duración: ~153 segundos

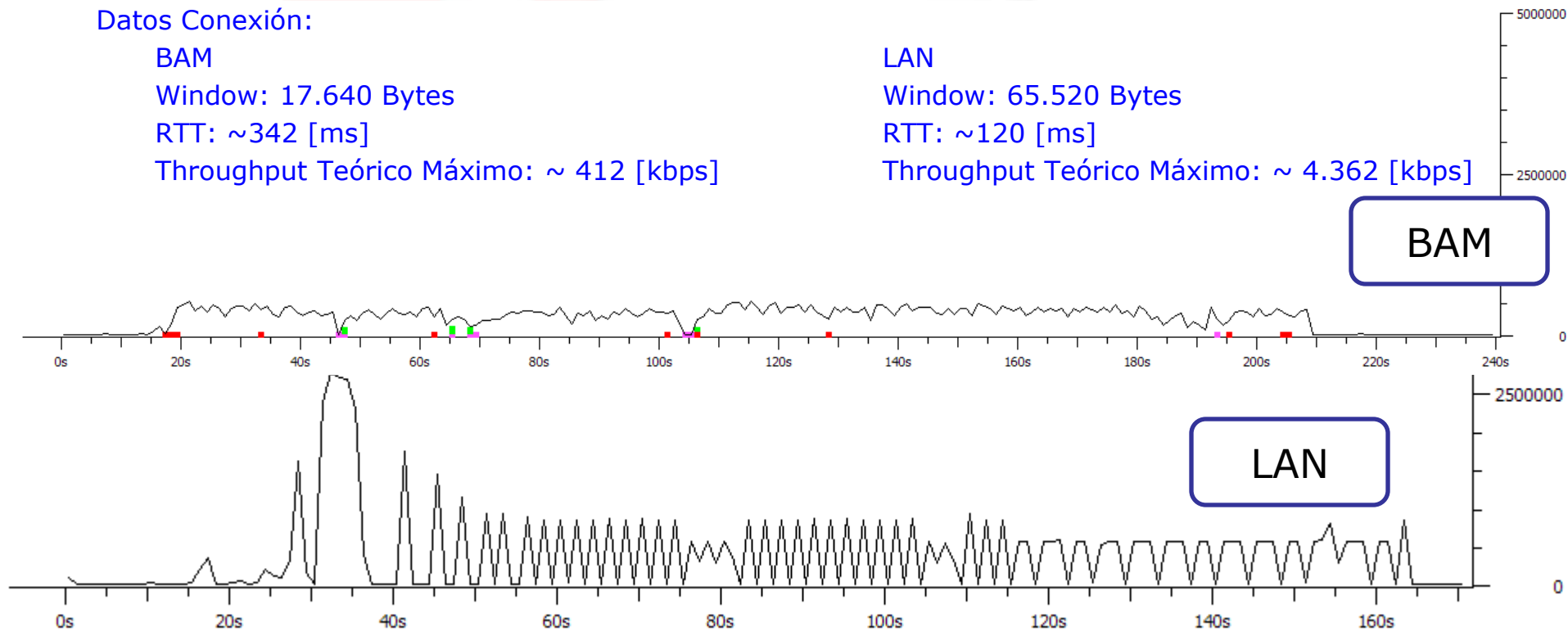
Throughput: ~500kbps

LAN

Window: 65.520 Bytes

RTT: ~120 [ms]

Throughput Teórico Máximo: ~ 4.362 [kbps]



Preguntas