



PROGRAMACIÓN EN RED (SOCKETS)

(©) Moreno, A. M. & Bravo, S. & Vázquez. A. (Redes I, 2020)

Contenido

2

□ Presentación

- ▣ Objetivos y entorno de desarrollo
- ▣ NNTP (*Network News Transfer Protocol*) simplificado
 - Origen
 - Conceptos
 - Mensajes
 - Ejemplos de dialogo
 - Gestión de datos en el servidor

Objetivos y entorno de desarrollo

3

- El objetivo de esta práctica es implementar una aplicación en red como
 - ▣ Usuario del nivel de transporte y
 - ▣ Según el modelo cliente-servidor
- Se implementará un servicio de transferencia de noticias NNTP simplificado
- Entorno de desarrollo
 - ▣ Estación de trabajo con S.O. Debian GNU/Linux 9 (*stretch*) (nopal.usal.es)
 - ▣ Sockets de Berkeley
 - ▣ Lenguaje de programación C

Protocolo de transferencia de noticias en red - NNTP (I)

4

- NNTP define un protocolo para la distribución, consulta, recuperación, y publicación de artículos de noticias ([RFC3977](#))
- Servicio estándar que se proporciona en TCP por el puerto 119
- Los grupos de noticias nacieron en 1979
- Sus inventores fueron Tom Truscott y Jim Ellis, dos estudiantes de la universidad estadounidense de Duke
 - ▣ Truscott y Ellis concibieron esta red, a la que llamaron Usenet (*Users Network*)
- Usenet es una red mundial de decenas de miles de servidores interconectados que almacenan los mensajes de miles de foros
- En 1991, RedIRIS puso en marcha los grupos españoles, cuyos nombres empiezan por "es" y se organizan por temáticas
 - ▣ Por ejemplo, "es.comp.os.linux"
- Google ha comprado los archivos de Usenet
 - ▣ Se puede acceder desde <https://groups.google.com/g/es.comp.os.linux>

NNTP (II)

5

□ Conceptos NNTP

- ▣ Los artículos se agrupan por temas/subtemas y se separan por puntos
 - Ej: es.comp.os.linux.misc
- ▣ Se identifican de forma única: "<número-consecutivo@nombre_servidor>"
- ▣ Los **ocho temas originales de Usenet** son los siguientes:
 - **comp:** Ordenadores
 - **sci:** Ciencia y tecnología
 - **soc:** Sociedad
 - **talk:** Temas generales
 - **rec:** Temas relacionados con el ocio, la recreación, el arte y la cultura
 - **news:** Información y noticias sobre la propia Usenet
 - **misc:** Miscelánea
 - **humanities:** Humanidades, Asuntos Culturales

Mensajes NNTP (I)

- Los mensajes NNTP son siempre líneas de caracteres terminadas con un par CR-LF (retorno de carro "\r" (ASCII 13 (0x0D)), - avance de línea "\n" (ASCII 10 (0x0A)))
- No deben exceder los 512 caracteres de longitud, contando todos los caracteres, incluido el CR-LF final. Por lo tanto, hay un máximo de 510 caracteres permitidos para el comando y sus parámetros
- Cualquiera de los caracteres que forman los comandos se pueden escribir indistintamente en mayúsculas o en minúsculas
- Cuando un comando o respuesta tenga varias líneas la última ha de ser una línea que sólo contenga un punto
- Los parámetros de los comando se separan por un espacio en blanco

Mensajes NNTP (II)

7

Comando NNTP	Parámetros	Función
LIST	Ninguno	Solicita una lista de los grupos de noticias disponibles. El servidor devuelve el nombre, el número de artículos y el número del primero y del último
NEWGROUPS	Día y hora en formato YYMMDD HHMMSS	Consulta todos los nuevos grupos disponibles desde la fecha indicada
NEWNEWS	Grupo de noticias, día y hora en formato YYMMDD HHMMSS	Consulta todos los nuevos artículos disponibles de un determinado grupo de noticias creados a partir de una fecha determinada
GROUP	Nombre del grupo de noticias	Se selecciona el grupo al que se realiza la solicitud. El servidor devuelve el número de artículos y el número del primero y del último
ARTICLE	número de artículo	Permite consultar un artículo específico
HEAD	número de artículo	Solicita solo la cabecera de un artículo
BODY	número de artículo	Solicita solo el cuerpo de un artículo
POST	Newsgroups: misc.test Subject: test article Texto del artículo .	Añadir un artículo a un grupo. El servidor guarda el artículo en un fichero misc/test/número añadiendo en la cabecera Date: YYMMDD HHMMSS, fecha legible. Message-ID: <número@nombre-servidor>
QUIT	None	Cierra la conexión con el servidor

Mensajes NNTP (III)

Código de estado del servidor de NNTP	Significado
2xx	Comando recibido, ejecución completada
3xx	Comando recibido, se requiere más información
4xx	Comando recibido, ejecución no completada
5xx	Comando desconocido

Mensajes NNTP (IV)

Código de estado del servidor de NNTP	Significado
200	Servidor <nombre-servidor> preparado
205	Despedida
211	Nº de artículos, primero y último nombre-grupo seleccionado
215	Listado de los grupos de noticias
221	Nº artículo id-artículo cabecera del artículo recuperada
222	Nº artículo id-artículo cuerpo del artículo recuperado
223	Nº artículo id-artículo artículo recuperado
230	Nuevos artículos desde día y hora
231	Nuevos grupos desde día y hora
240	Artículo recibido correctamente
411	No existe ese grupo de noticias
423	No existe el artículo en este grupo de noticias
430	No se encuentra ese artículo
441	Subida fallida de un artículo
500	Comando desconocido
501	Error de sintaxis en (Ejemplo de la sintaxis correcta)

Ejemplos de diálogo: leyendo artículos I

S: (Escucha en el puerto 119)

C: (Solicita conexión al puerto 119)

S: 200 Servidor de noticias nogal.usal.es preparado

(El cliente pregunta por los grupos)

C: LISTADO

S: 500 Comando no reconocido

C: LIST

S: 215 listado de los grupos en formato “nombre ultimo primero fecha descripcion”

S: net.wombats 0000000543 0000000501 201128 000000 Descripción del grupo

S: net.unix-wizards 0000010125 0000010011 201127 000000 Descripción del grupo
(más información)

S: net.idiots 0000000100 0000000001 191128 000000 Descripción del grupo

S: .

(el cliente selecciona un nuevo grupo)

C: GROUP Linux

S: 501 Error de sintaxis en GROUP newsgroup

C: GROUP net.unix-wizards

S: 211 104 10011 10125 net.unix-wizards
(hay 104 artículos, del 10011 al 10125)

(el cliente selecciona un artículo para leer)

C: ARTICLE 401

S: 423 El artículo 401 no existe en el grupo de noticias

Ejemplos de diálogo: leyendo artículos II

C: ARTICLE 10110

S: 223 10110 <23445@nagal.usual.es> artículo recuperado

S: (Cabecera y cuerpo del artículo)

S: .

(el cliente consulta la cabecera)

C: HEAD 10110

S: 221 10110 <23445@nagal.usual.es> cabecera del artículo recuperada

S: (Cabecera)

S: .

(el cliente desea ver el cuerpo del artículo)

C: BODY 10110

S: 222 10110 <23445@nagal.usual.es> cuerpo del artículo recuperado

S: (Cuerpo del artículo)

S: .

(el cliente finaliza la sesión)

C: QUIT

S: 205 Adios

Ejemplos de diálogo: nuevos grupos y artículos

C: NEWGROUPS 20201130 000000

S: 231 Nuevos grupos desde 30/11/20 00:00:00

S: .

(No hay ninguno)

C: NEWGROUPS 201126 120000

S: 231 Nuevos grupos desde 26/11/20 12:00:00

S: net.music.gdead

S: net.games.sources

S: .

C: NEWNEWS net.music.gdead 201127 120000

S: 230 Nuevos artículos desde 27/11/20 12:00:00

S: (lista de artículos número, id y tema (subject))

S: .

Ejemplos de diálogo: Subiendo un artículo

C: POST

S: 340 Subiendo un artículo; finalice con una línea que solo contenga un punto

C: Newsgroups: misc.test

C: Subject: I am just a test article

C:

C: Este es el texto del artículo.

C: .

S: 240 Artículo recibido correctamente.

(o)

S: 441 No se ha podido subir el artículo por tal causa

(La sintaxis no es correcta o no se ha podido escribir el fichero)

Gestión de los datos en el servidor I

- Crear un directorio relativo a la carpeta donde se encuentren los ejecutables llamado noticias y en este directorio:
 - ▣ Crear un directorio llamado artículos y dentro de este crear una estructura jerárquica de directorios con igual nombre que los grupos de noticias: tema/subtema/n_articulo
 - Siendo n_articulo el ID del mensaje n_articulo@nogal.usal.es
 - El número, que ha de ser único en un determinado servidor, se puede almacenar en un fichero llamado n_articulos que contenga el último número asignado
 - ▣ En n_articulo se almacena el artículo tal cual ha sido transmitido.
 - ▣ Ejemplo: more 1
Newsgroups: local.test
Subject: primero
Date: 201127 111307 Fri, 27 Nov 2020 11:13:07 -0000 (UTC)
Message-ID: <1@nogal.usal.es>

Este es el contenido del primer artículo
.
(La fecha y el ID-mensaje lo rellena el servidor)
- Si es el primer artículo de un determinado grupo de noticias habrá que crear los directorios. En caso contrario ya estarán creados

Gestión de los datos en el servidor II

- ▣ Fichero grupos que contiene:

- Nombre, último artículo (10 caracteres), primer artículo (10 caracteres), fecha de creación en formato YYMMDD HHMMSS y descripción separados por un blanco

- Ejemplo:

local.redes 0000000000 0000000001 201127 200000 Grupo general local
(No hay artículos)

local.deporte 0000000001 0000000001 201127 200000 Grupo de test local
(Hay un artículo)

- noticias

- ▣ grupos

- ▣ articulos

- local

- deporte

- 1

Requisitos

□ Programa Servidor

- Aceptará peticiones de sus clientes tanto en TCP como en UDP
- Registrará todas las peticiones en un fichero de "log" llamado nntpd.log en el que anotará:
 - Fecha y hora, comunicación realizada: nombre del host, dirección IP, protocolo de transporte y nº de puerto efímero del cliente
 - Fecha y hora, comando enviado o recibido
- Se ejecutará como un "daemon"

□ Programa Cliente

- Se conectará con el servidor bien con TCP o UDP
- Leerá por parámetros el nombre del servidor y el protocolo de transporte TCP o UDP de la siguiente forma:
 - cliente nombre_o_IP_del_servidor TCP
- Consultará y enviará noticias

Requisitos: pruebas

17

- Durante la fase de pruebas el cliente podrá ejecutarse como se muestra en el ejemplo de diálogo anterior, pero en la versión para entregar el cliente
 - ▣ Leerá las órdenes desde un fichero .txt que se leerá como tercer parámetro. Así los clientes se ejecutarán:
 - `./cliente nogal TCP ordenes1.txt &`
 - ▣ Escribirá los mensajes de progreso y los mensajes de error y/o depuración en un fichero con nombre el número de puerto efímero del cliente y extensión .txt

Requisitos: versión entregable

18

- Para verificar que esta práctica funciona correctamente y permite operar con varios clientes, se utilizará el *script* `lanzaServidor.sh` que ha de adjuntarse obligatoriamente en el fichero de entrega de esta práctica
- El contenido de `lanzaServidor.sh` es el siguiente:

```
# lanzaServidor.sh
# Lanza el servidor que es un daemon y varios clientes
./servidor
./cliente nogal TCP ordenes1.txt &
./cliente nogal TCP ordenes2.txt &
./cliente nogal TCP ordenes3.txt &
./cliente nogal UDP ordenes1.txt &
./cliente nogal UDP ordenes2.txt &
./cliente nogal UDP ordenes3.txt &
```

- Entregar la estructura de directorios del servidor que permita ejecutar las diferentes órdenes

Requisitos: documentación

19

- Entregar un informe en formato PDF que contenga:
 - ▣ Detalles relevantes del desarrollo de la práctica
 - ▣ Documentación de las pruebas de funcionamiento realizadas