

Trabajo Práctico Especial
Protocolos de Comunicación
Grupo 10



Morrone, Franco

Padula Morillo, Alejo

Pugliano, Tobias

Zapico, Bernardo

Índice

Índice.....	1
Descripción.....	2
Aplicación.....	2
Protocolo.....	2
Problemas Encontrados.....	3
Limitaciones y posibles extensiones.....	3
Conclusiones.....	3
Ejemplos de Prueba.....	4
Guía de Instalación.....	6
Monitoreo.....	6

Descripción

Aplicación

La aplicación es un servidor que implementa el protocolo pop3 para que un usuario manipule sus correos. Se cuenta con todos los comandos obligatorios definidos en RFC 1939 y funcionan de la misma manera que en software libre común como Dovecot.

Por ende el flujo es el siguiente: Al conectarse al servidor uno necesita autenticarse con los comandos USER y PASS para ingresar su usuario y contraseña respectivamente. Con una autenticación exitosa se pasa al estado de transacciones.

En el estado de TRANSACTION se tiene acceso a todos los comandos que se necesitan para manipular los correos. Si se quiere ver la lista de IDs de cada correo se usa LIST, para leer correo específicos se usa el comando RETR, para borrarlos con DELE, STAT para ver estadísticas, NOOP para verificar si el servidor está escuchando y RSET para que mensajes previamente marcados como “borrados” pierdan sus marcas.

Requiere de un directorio con estructura ./root/user/ donde el root es el directorio que se asigna en el comando de ejecución (véase Guía de Instalación) y user es un directorio con el nombre del usuario. Este último tiene que tener las 3 carpetas estándar de POP3; cur, new y tmp.

La implementación es no-bloqueante porque se usa el Selector provisto por la cátedra.

Protocolo

Para asistir en el uso y mantenimiento del servidor se implementó el protocolo Push3. Similar a pop3, cuenta con un estado de autenticación y otro de transacciones. En autenticación se ingresa las credenciales con “login [nombre de usuario] [contraseña]” se utilizan las mismas credenciales que para el servidor de pop3.

Una vez está autenticado el usuario, consigue acceso a los comandos del estado de transacciones. Primero se encuentra “metrics” para ver cuántas conexiones hay hacia el servidor, la cantidad total de conexiones que se hayan realizado en todo el runtime del servidor y la cantidad total de bytes que se hayan leído a través de todos los correos.

Problemas Encontrados

- Hubo una carga de tiempo muy pesada antes de que se pueda iniciar el trabajo en la forma de aprender programación con sockets no bloqueantes
- En casos donde posiblemente se tiene que imprimir una cantidad muy alta de líneas, es posible que se llene el buffer de escritura. Para solucionar esto se utiliza un estado pending que pone los datos en una fila e itera para escribir todo en caso de que no entre de una vez en el buffer.
- Se consumió bastante tiempo validando todos los casos posibles de error para tener una cobertura de mensajes adecuada.
- Se necesitó una cantidad no-trivial de tiempo para comprender el manejo del buffer provisto por la cátedra.

Limitaciones y posibles extensiones

- El protocolo Push3 solo muestra las métricas del sistemas. A futuro se implementaría la configuración del servidor
- Añadir transformación de correo

Conclusiones

Éste proyecto ofreció una oportunidad única para concretar en integrar los conocimientos adquiridos a lo largo de la cursada. Sea el vaivén de la comunicación cliente-servidor, la implementación de un protocolo nuevo (con la documentación RFC que eso implica) así como el uso de sockets y selectores.

Ejemplos de Prueba

Veamos el paso-a-paso de la inicialización y uso del servidor

Primero iniciamos el servidor

```
$ ./pop3.out -d ~/Maildir -u ale:passAle
```

Esto inicializa el servidor con el directorio de correo en /home/Maildir y voy a tener el usuario "ale" con contraseña "passAle".

```
Listening for POP3 on TCP port 2252
Listening for config on TCP port 2254
```

Esto nos indica que se inicializó correctamente el el servidor en 2252 y el protocolo de monitoreo en 2254. Dejámos el servidor corriendo y abrimos una terminal nueva:

```
$ ncat -C localhost 2252
```

Nos conectamos al puerto del servidor

```
+OK PushPop3
```

El servidor nos responde con un OK y nos empieza a escuchar

```
STAT
-ERR You must be logged in to use this command
USER ale
+OK
PASS alePass
-ERR Invalid user & pass combination, try again
user ale
+OK
pass passAle
+OK maildrop locked and ready
```

Como la conexión comienza en estado de AUTHENTICATION, no tenemos acceso a los comandos de TRANSACTION. Por ende nos autenticamos asegurándonos que las credenciales sean correctas

Pedimos los correos del usuario:

```
LIST
+OK 6 messages (2437 octects)
1 401
2 382
3 424
4 418
5 379
6 433
.
```

De aquí podemos utilizar RETR para pedir y DELE para borrar correos

```
RETR 1
+OK 401 octects
Return-Path: <alex dg@alepm.deb>
X-Original-To: ale@alepm.deb
Delivered-To: ale@alepm.deb
Received: from postfix (localhost [IPv6:::1])
        by alepm.deb (Postfix) with ESMTP id DAFE3A4129
        for <ale@alepm.deb>; Thu, 24 Oct 2024 22:50:52 -0300 (-03)
Message-Id: <20241025015119.DAFE3A4129@alepm.deb>
Date: Thu, 24 Oct 2024 22:50:52 -0300 (-03)
From: alex dg@alepm.deb

.....

Papopepoparapapapapiparapopepo
.
```

```
DELE 6
+OK message 6 marked to be deleted
.
```

Y naturalmente podemos resetear la marca para borrar

```
RSET
+OK
```

De aquí podemos realizar QUIT para terminar la conexión

```
QUIT
+OK signing off
Ncat: Connection reset by peer.
```

Guía de Instalación

Para instalar el programa solo se necesita clonar el repositorio con el código. Una vez hecho esto, uno se para en el directorio raíz y corre el comando "make all".

Al correr el comando se generan dos ejecutables: pop3.out y client.out. Para inicializar el servidor se ejecuta pop3.out, que dispone de los siguientes flags:

- -d: Obligatorio. Señala la ubicación del directorio de correos
- -u: Opcional: Agrega un usuario para que pueda comunicarse con el servidor. Hasta 10. Si bien es optativo, si no se usa no habrá un usuario alguno que pueda conectarse. Usa el formato "usuario:contraseña"
- -p: Opcional. Indica en qué puerto se quiere inicializar el socket

Monitoreo

Push3 cuenta con el comando Metrics que marca:

- La cantidad total de conexiones activas
- La cantidad total de conexiones que se hayan establecido
- La cantidad total de bytes de correo leídos