

LICENCIATURA EM ENGENHARIA INFORMÁTICA
Sistemas Operativos

Relatório Projeto
2017/2018



Gonçalo Pereira – N.º 35302

João Brito – N.º 37880

Introdução

Neste projeto, proposto pelo docente, pretendemos demonstrar os conhecimentos apreendidos nas aulas da unidade curricular.

O objetivo principal é o de pôr em prática os conhecimentos obtidos relativamente a uso de *FIFO's (named pipes)* e a sua implementação para a comunicação entre duas aplicações/processos, neste caso entre um servidor e um cliente. Aplicámos, também, algum conhecimento acerca de *threads* (que adquirimos ao longo do semestre).

Ficheiros

1. *Makefile*

Ficheiro que compila os ficheiros de código para ser possível executá-los

2. *cs.h*

Ficheiro com as bibliotecas que vão ser usadas pelo cliente e servidor, bem como uma estrutura usada pelo servidor.

3. *server.c*

Recebe as informações proveniente do cliente, em forma de comando, dando uma resposta adequada. No servidor é possível:

- Entrar como utilizador
- Entrar como administrador
- Obter permissões de ficheiros
- Observar o conteúdo de ficheiros
- Verificar a informação guardada (*log*)

O servidor começa por verificar se o utilizador, enviado pelo cliente, corresponde a algum dos *logins* guardados. Se for, poderá enviar pedidos ao servidor e obter a informação desejada (se tiver permissões para tal). Se não estiver de acordo com a informação de autenticação detida pelo servidor, aparecerá, do lado do cliente, uma mensagem de erro a informar exatamente isso.

Adicionalmente, o servidor “dá” ao cliente 5 minutos em que este pode executar qualquer comando. Após este período, o cliente receberá um *prompt* a informá-lo de que o período de autenticação se esgotou, podendo de seguida introduzir os dados de autenticação outra vez.

4. client.c

Este programa serve de interface com o servidor e é onde podemos inserir o nome e *password* para podermos aceder ao servidor. Neste podemos inserir os seguintes comandos:

- “**exit**”- sair do programa
- “**show [file]**”- mostra o conteúdo do ficheiro cujo nome é inserido a seguir (serve também para os outros comandos)
- “**info [file]**” – mostra informações do ficheiro, tal como as suas permissões
- “**log**” – mostra toda a informação processada como input e mantida pelo servidor (apenas acessível a *admins*)

```

brito@britopc: ~/Desktop/projeto
brito@britopc:~/Desktop/projeto$ cc -o server server.c -pthread
brito@britopc:~/Desktop/projeto$ ./server &
[1] 4466
brito@britopc:~/Desktop/projeto$ cc -o client client.c
brito@britopc:~/Desktop/projeto$ ./client

Enter username and password > aluno 1234
OK, you're authorized for 5 minutes!

Enter prompt > info cs.h
You ran out of time, please authenticate again!

Enter username and password >
  
```

Autenticação seguida de pedido de reintrodução de dados (após 5 minutos)

```

brito@britopc: ~/Desktop/projeto
brito@britopc:~/Desktop/projeto$ ./client

Enter username and password > aluno 1234
OK, you're authorized for 5 minutes!

Enter prompt > info cs.h

--- File Information ---

User permissions: S_IRUSR
Group permissions: S_IRGRP
Other permissions: S_IROTH

Enter prompt >
  
```

Comando “info [file]”

```

brito@britopc: ~/Desktop/projeto
brito@britopc:~/Desktop/projeto$ ./server &
[1] 4618
brito@britopc:~/Desktop/projeto$ ./client

Enter username and password > aluno 1234
OK, you're authorized for 5 minutes!

Enter prompt > show cs.h

--- File Content ---

#include <stdio.h>
#include <stdlib.h>
#include <string.h>
#include <math.h>
#include <sys/types.h>
#include <sys/stat.h>
#include <sys/wait.h>
#include <fcntl.h>
#include <unistd.h>
#include <pthread.h>
#include <dirent.h>

typedef struct{
    char msg[100];
    int tempo;
} aviso_t;

Enter prompt >
  
```

Comando “show [file]”

```

brito@britopc: ~/Desktop/projeto
[1]+  Terminated                  ./server
brito@britopc:~/Desktop/projeto$ ./server &
[1] 4543
brito@britopc:~/Desktop/projeto$ ./client

Enter username and password > aluno 1234
OK, you're authorized for 5 minutes!

Enter prompt > log
Sorry, but you're not authorized to access this information!

Enter prompt >
  
```

*Comando “log”
(com e sem permissão)*

```

brito@britopc: ~/Desktop/projeto
brito@britopc:~/Desktop/projeto$ ./server &
[1] 4587
brito@britopc:~/Desktop/projeto$ ./client

Enter username and password > admin 4333
OK, you're authorized for 5 minutes!

Enter prompt > log

--- Logged Information ---

Successful Authentication by aluno 1234
Failed Authentication by aluno 1234
log
Successful Authentication by aluno 1234
Failed Authentication by aluno 1234
show cs.h
Successful Authentication by aluno 1234
  
```

Conclusão

O desenvolvimento desta avaliação final ajudou-nos a enriquecer os nossos conhecimentos na área da comunicação entre processo usando *pipes*, pondo em prática os conhecimentos adquiridos ao longo do semestre. Com este trabalho ficámos com uma ideia formada de como é possível a comunicação entre duas aplicações.

Optámos por dar uma utilidade à comunicação entre o cliente e o servidor, dando a este último a capacidade de executar comandos e fazer o “trabalho pesado”, cujos resultados são enviados para o cliente (que por sua vez pediu informação ao servidor). Tentámos sempre colocar todas as operações mais sensíveis (autenticação, abertura e leitura de ficheiros, criação de *FIFO's*) do lado do servidor, algo que a nosso ver faz sentido.

Existiram pontos do que nos levantaram algumas dúvidas e problemas para os quais tentámos arranjar a solução mais adequada. Com isto, infelizmente, alguns aspetos no trabalho ficaram passíveis de melhoria.

Podemos concluir que o projeto proposto na cadeira de Sistemas Operativos foi bastante útil, uma vez que implementámos toda a matéria lecionada durante o semestre e deu-nos alicerces que nos serão úteis a nível profissional num futuro próximo.