



Universidade da Beira Interior

## Projeto Final de Sistemas Operativos



**Trabalho realizador por:**

Tiago Roxo Nº 37032

Daniel Fernandes Nº37197

**Docente:** Paul Crocker

Covilhã, junho de 2018

## Objetivos do trabalho

O objetivo do trabalho foi estabelecer uma comunicação entre servidor e administrador, permitindo a este aceder a ficheiros da diretoria do servidor, e permitir a conexão de clientes ao servidor, registando as mensagens enviadas destes.

## Descrição do programa

Este programa consiste numa comunicação entre administrador e servidor e permitir que o administrador conseguia ler/ter acesso a ficheiros da diretoria deste. Para tal o administrador necessitará de introduzir *password* que, a ser validada, iniciará um temporizador relativo ao tempo disponível para usar as funcionalidades disponibilizadas pelo servidor. O formato de introdução de *password* será “pass XXX” onde X será um número. A introdução de texto com outro formato dará a reposta de “Sem permissão” ou “Comando inválido”, com a exceção de “sair”, conduzindo ao encerramento da aplicação. A introdução de *password* inválida manterá o administrador no menu de introdução de *password*. A contabilização do tempo proporcionado ao administrador será controlado via uma *thread*, com um contador e uma mensagem de aviso associada, à semelhança da funcionalidade embutida na soshell durante as aulas práticas de Sistemas Operativos. À chegada do valor zero do contador o administrador perderá acesso ao servidor, tendo de reintroduzir a sua passa para voltar a tê-lo. O administrador terá a opção de ler ficheiros, por introdução do ser nome, que dará mensagem de erro caso este não existe. Os ficheiros presentes neste trabalho serão “log.txt”, contendo os dados dos clientes, e “dino.txt”, contendo um *easter egg*.

O formato considerado para acesso dos clientes ao servidor foi o 1 (da ficha das notas do professor). Neste formato os clientes iniciarão sessão e enviarão a sua mensagem para o servidor, contendo o PID de processo, conteúdo da mensagem e data de envio. O *prompt* dos utilizadores conterà o identificador de processo destes, servindo de fator de identificação dos clientes. O envio de mensagens para o cliente será por *message queues* e o servidor receberá a informação destes e adicionara-a ao final do ficheiro “log.txt”, repositório de informação de clientes, acedível apenas por administrador.

O servidor receberá mensagens de clientes e pedidos de visualização de ficheiros por administrador. Para conseguir responder a estes dois tipos de envios/pedidos o servidor terá um *fork*, onde o processo filho tratará das *message queues* dos clientes e o processo pai dos pedidos do administrador. Este último implica a abertura de ficheiro e leitura de blocos de dados deste, de tamanho MAXBUFF (1024), a ser enviado para FIFO, permitindo a leitura destes dados do lado do administrador e consequente mostragem em ecrã. Para determinar o tamanho de dados em blocos, o servidor

utilizará os primeiros quatro bytes para registar esta informação. Do ponto de vista do administrador, este lerá os *buffers* recebido de servidor até que o tamanho de dados do buffer recebido seja inferior a MAXBUFF. Quando o administrador necessitar de requerer uma leitura de ficheiro será incorporado no primeiro byte do *buffer* a sua *flag* de acesso ('0' = sem acesso, '1' = com acesso); para o administrador ter acesso terá que introduzir uma *password* válida, com validade de 10 segundos (tempo exemplificativo). O envio de dados, por parte do servidor, para o administrador, apenas ocorrerá se este tiver acesso. O diagrama seguinte exhibe as principais interações entre as entidades do sistema.

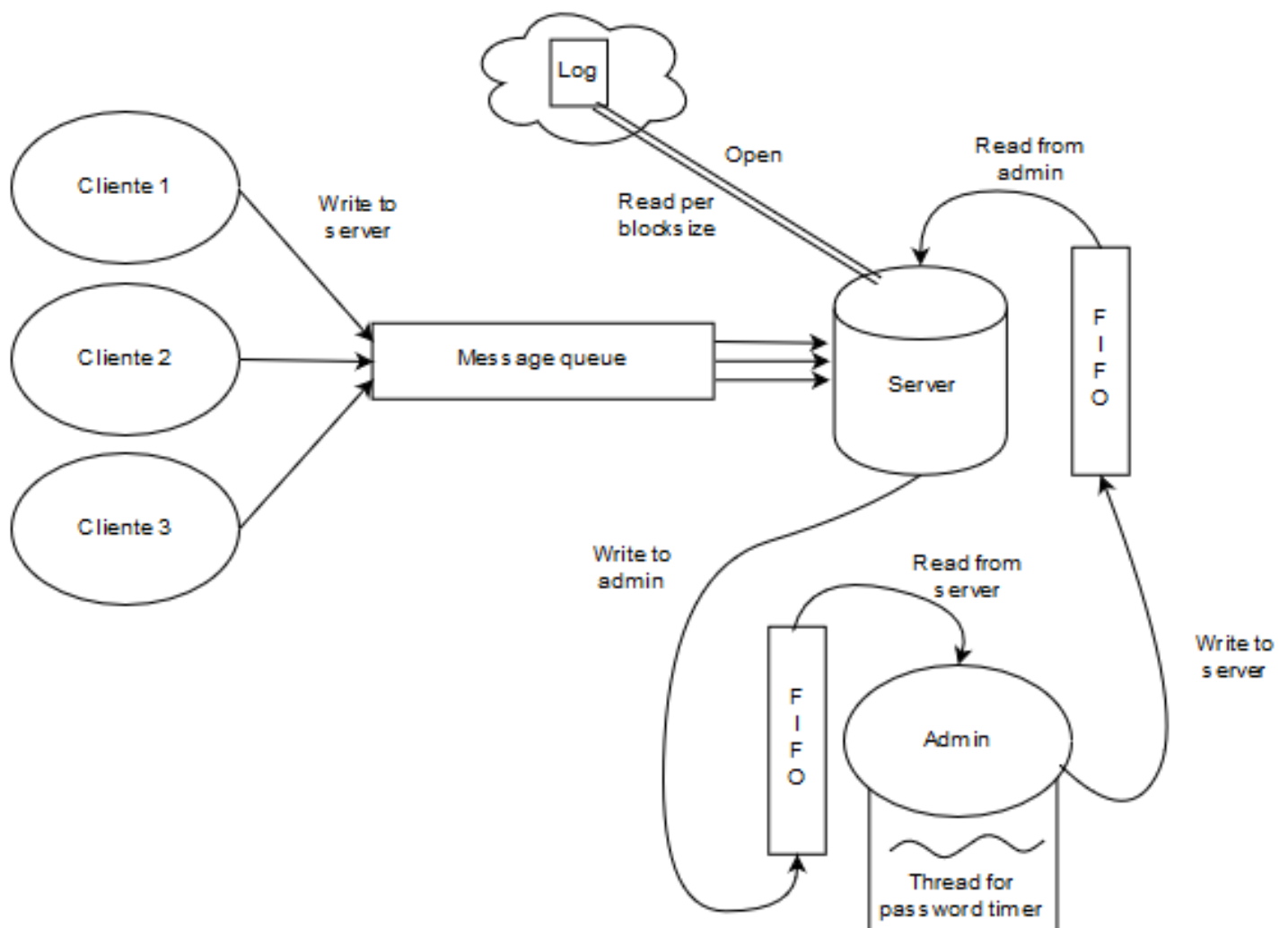


Figura 1 - Interações entre componentes do sistema

## Conclusão

Este projeto permitiu consolidar conceitos abordados durante as aulas práticas e teóricas de Sistemas Operativos, fornecendo liberdade suficiente para implementar projeto personalizados. O trabalho realizado contém todas as funcionalidades exigidas pelo enunciado do projeto; mantém a simplicidade nas suas funcionalidades mas exhibe as componentes requeridas, nomeadamente uso de *threads*, *fifos* e *message queues*. Possíveis melhorias ao realizado seria a implementações de início, paragem, recomeço e pausa de servidor por parte do administrador; o término de sessões de cliente seriam outras possíveis melhorias.

## Referências

Fichas das Aulas Praticas - IPC - Comunicação entre processos 1ª e 2ª parte

Ficheiros disponibilizados pelo docente via <https://spocs.it.ubi.pt>, nomeadamente das pastas “fifo” e “filas”