

Sistemas Operativos 2016/2017

Relatório do Projecto

Servidor HTTP

Tiago Miguel Vitorino Simões Gomes – 2015238615

Leonardo Machado Alves Vieira – 2015236155

1 Manual do Utilizador

- **Servidor:** Apenas utilizar o comando “./server” num terminal Linux
- **Cliente:** Abrir num browser localhost:<port> <page>.
- **Consola:** Correr o comando “./console” num terminal Linux e escolher uma das opções apresentadas no menu.
- **Estatísticas:** Num novo terminal Linux executar o comando “kill -10 <stat_pid>” para imprimir as estatísticas e “kill -12 <stat_pid>” para lhes dar *reset*.

2 Processo Principal

O processo principal é onde esta a correr tudo o que é necessário para o bom funcionamento do servidor. No arranque irão ser carregadas para uma estrutura todas as informações definidas no ficheiro *config.txt*. De seguida é criada a memória partilhada para a comunicação com o processo de estatísticas, sendo este também criado de seguida. É criado o *named pipe* para comunicar com o processo de gestão de configurações. Depois são criadas as *threads* necessárias. Uma *pipe_thread* que chama a função *listen_console* que vai ficar à espera de informação enviada pela consola de configurações. Outra *scheduler_thread* que chama a função *scheduler* que trata da organização e gestão do buffer de pedidos dos clientes. As restantes são para o tratamento de pedidos e a quantidade é dependente do que está especificado em *config->threadp*, sendo que esta estrutura tem o valor inicial definido em *config.txt*, mas pode ser alterado pela consola de configurações. As *worker threads*, antes de responder a um pedido vão ver que tipo de página foi pedida (estática ou comprimida), e caso seja comprimida irá descomprimir, enviar e por fim apagar o ficheiro descomprimido. Todos os pedidos irão ficar guardados num *buffer* dinâmico, isto para facilitar alterações no caso da mudança da política de escalonamento e para evitar ter um limite fixo para o tamanho do *buffer*.

O programa também está preparado para, após inicializar todos os recursos, receber um sinal *SIGINT* que irá encerrar todas as *threads*, todos os processos, eliminar a memória partilhada, libertar todas as alocações dinâmicas e por fim desligando por completo o servidor.

3 Processo Gestão de Estatísticas

O processo de gestão de estatísticas, que é iniciado no arranque do servidor, está encarregue de calcular as estatísticas relacionadas com o tipo páginas pedidas e o tempo de processamento delas, desde que o servidor iniciou. Após o pedido do cliente ser respondido, o processo lê a estrutura do pedido da memória partilhada e, sendo este um pedido novo, este altera as estatísticas e escreve-as no ficheiro *server.log* que é constantemente mapeado e desmapeado na memória, para que, com algumas funções auxiliares, este seja mapeado com o tamanho pretendido para receber uma *string* com a informação estatística do pedido. Ao receber o sinal *SIGUSR1* ou *SIGUSR2* irá imprimir as estatísticas ou fazer *reset* nas mesmas, respetivamente.

4 Processo Gestão de Configurações

O processo de gestão de configurações está encarregue de enviar para o processo principal, as alterações à configuração pedidas pelo utilizador. É possível alterar o *scheduling*, a *threadpool* e os ficheiros comprimidos permitidos. Estas alterações são pedidas através de um terminal Linux separado e são enviadas numa estrutura *message* através de um *named pipe* para o processo principal onde se encontra uma *thread* sempre pronta para receber informação desse *named pipe*, de modo a que as alterações possam ser realizadas instantaneamente. Isto porque é apenas do lado do servidor que as alterações se vão ser efetuadas. O *console.c* apenas tem algumas proteções.