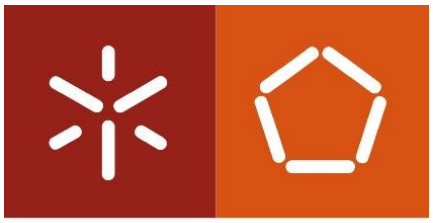


Universidade do Minho

Mestrado em Engenharia Informática


# Tecnologias de Segurança

## *Trabalho Prático 3* - Djumbai: Serviço Local de Troca de Mensagens


José de Matos Moreira (PG53963) José dos Santos Mendes (PG53967)
Pedro Freitas (PG52700)


Ano Letivo 2023/2024

# Índice

# Capítulo 1

# Introdução

Na era tecnológica em que vivemos, os serviços de conversação assumem um papel importantíssimo na comunicação entre os diversos utilizadores do mundo digital. Estes mesmos serviços garantem o contacto próximo entre os referidos utilizadores e oferecem diversas funcionalidades, garantindo, não só, a eficiência da comunicação síncrona, como, também, o potencial dinâmico da comunicação síncrona.

Com o objetivo de se explorar as diferentes funcionalidades dos serviços mencionados, propôs-se o desenvolvimento de um serviço de conversação entre utilizadores locais de um sistema **Linux**. Deste modo, o serviço a desenvolver deve permitir o envio e a respetiva leitura de mensagens, suportando comunicação assíncrona, tal como acontece no mail, e comunicação síncrona, exatamente como se observa nos serviços de comunicação instantânea. Porém, o serviço proposto não se limita à troca de mensagens entre utilizadores. Em concreto, o mesmo deve fornecer funcionalidades de adição e remoção de utilizadores e deve suportar a noção de grupos privados de conversação, oferecendo mecanismos para criação de grupos, remoção de grupos e gestão dos seus membros.

Por outro lado, o projeto foca-se num tema principal: a **segurança**. Com isto, reitera-se a ideia de que o serviço deve garantir a confidencialidade e a integridade das mensagens trocadas. Não menos importante, a preocupação central do projeto deve, também, focar-se em manter a disponibilidade do serviço.

Ao longo deste documento, discutir-se-ão os detalhes de implementação do serviço, incluindo as decisões arquiteturais tomadas para garantir o seu correto funcionamento num ambiente seguro. No mesmo, encontrar-se-ão todas as linhas de raciocínio que levaram a que o serviço fosse desenvolvido da maneira que foi, bem como todas as justificações necessárias.

# Capítulo 2

# Arquitetura Funcional

**Djumbai**, o serviço de conversação desenvolvido, é constituído por sete módulos independentes. Passa-se, assim, a explicar, detalhadamente, cada um dos mesmos.

## 2.1 Módulos

- **djumbai-user-manager** - oferece as funcionalidades de adição e remoção de utilizadores locais ao sistema e pode ser invocado através dos comandos **djumbai-user-manager adduser <*username*>** e **djumbai-user-manager deluser <*username*>**
- **djumbai-group-manager** - permite a criação e a destruição de grupos e as respetivas operações de adição e remoção de membros, sendo invocado através de quatro diferentes comandos, sendo, cada um, responsável por uma ação diferente: **djumbai-group-manager groupadd <groupname>**, **djumbai-group-manager groupdel <groupname>**, **djumbai-group-manager addusergroup <username> <groupname>** e **djumbai-group-manager delusergroup <username> <groupname>**
- **djumbai-private-manager** - responsável por permitir a criação e a destruição de *chats* privados entre dois utilizadores, recorrendo aos comandos **djumbai-private-manager addprivate <username1> <username2>** e **djumbai-private-manager delprivate <username1> <username2>**
- **djumbai-group-chat** - permite a troca de mensagens entre um grupo privado de conversação e pode ser utilizado através de dois comandos distintos que permitem enviar e ler mensagens de/para um grupo: **djumbai-group-chat sendgroup <groupname>** e **djumbai-group-chat readgroup <groupname>**

- **djumbai-private-chat** - oferece funcionalidades de troca de mensagens em *chat* privado, entre dois utilizadores, podendo, os mesmos, recorrer aos comandos **djumbai-private-chat sendprivate <username>** e **djumbai-private-chat readprivate <username>** de forma a enviar e ler mensagens para/de um utilizador, respetivamente

- **djumbai-mail-server** - único módulo a ser executado em *background*, que permite que os diversos utilizadores tenham acesso à funcionalidade de comunicação assíncrona (do estilo mail) e que deve ser inicializado através do comando **djumbai-mail-server**

- **djumbai-mail-client** - módulo que permite, aos utilizadores, a troca de mensagens, de forma assíncrona, oferecendo funcionalidades de envio e leitura de mensagens, recorrendo aos comandos **djumbai-mail-client sendmail <username>** e **djumbai-mail-client readmails**

## 2.2 Ficheiros

De forma a garantir o correto funcionamento do serviço, surgiu a necessidade de se recorrer à criação e consequente utilização de diversos ficheiros. Deste modo, criou-se a pasta **djumbai-memory**, sendo a mesma constituída por quatro diretorias, explicadas de seguida:

- **server** - pasta que armazena os ficheiros utilizados pelo módulo **djumbai-mail-server**, sendo, a mesma, responsável por armazenar as mensagens enviadas por um utilizador, de forma assíncrona, até que o destinatário as requisite

- **users** - pasta que reúne os ficheiros utilizados para guardar as mensagens, enviadas de forma assíncrona, já lidas pelo destinatário

- **grpups** - pasta que guarda os ficheiros utilizados na comunicação de grupos privados de conversação, sendo, cada um deles, constituído pelas mensagens enviadas para um grupo

- **privates** - pasta que organiza os ficheiros utilizados na comunicação síncrona efetuada entre dois utilizadores, sendo, os mesmos, constituídos pelas mensagens trocadas entre esses mesmos utilizadores

## 2.3 Operabilidade

Pelos módulos descritos anteriormente, conseguem-se perceber, facilmente, as diferentes funcionalidades implementadas. Porém, mostra-se necessário explicar a forma como o serviço opera, recorrendo aos módulos referidos e a outro tipo de objetos, como, por exemplo, ficheiros.

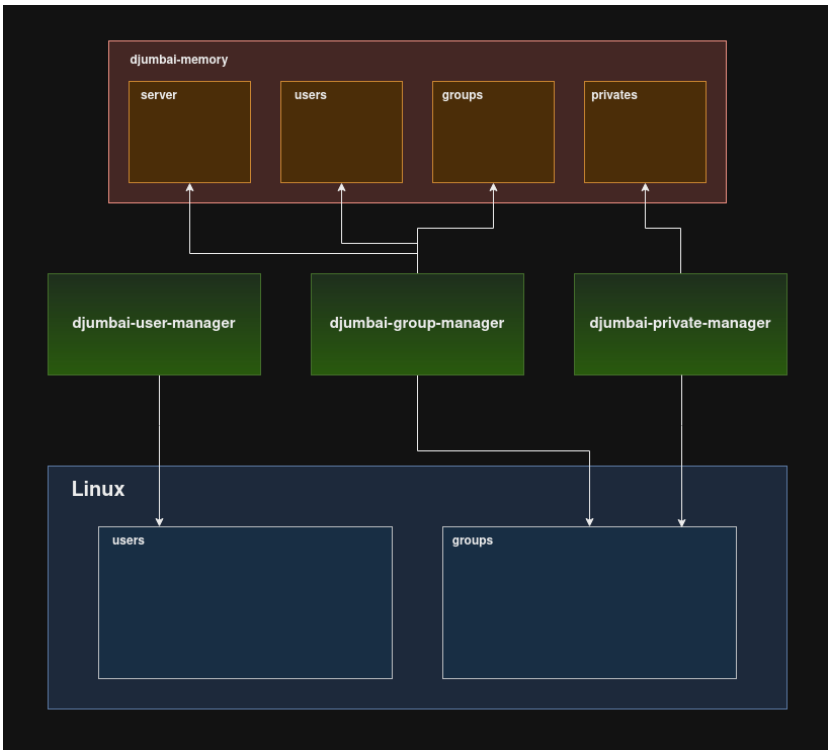


Figura 1: **Diagrama Arquitetural I**

Pelo diagrama apresentado, conseguem-se detetar as diferentes interações entre três dos módulos do serviço e outro tipo de componentes. Em primeiro lugar, o módulo **djumbai-user-manager** tem a capacidade de adicionar/remover utilizadores locais, conseguindo, portanto, interagir com o respetivo parâmetro do sistema operativo. Por outro lado, o módulo **djumbai-group-manager**, ao conseguir gerir os diferentes grupos, consegue, consequentemente, interagir com o sistema operativo. De forma a que o serviço funcione, o mesmo também cria/elimina ficheiros armazenados na memória do serviço, **djumbai-memory**, mais especificamente nas pastas **server**, **users** e **groups**. Em último lugar, o módulo **djumbai-private-manager**, ao gerir os *chats* entre os diversos utilizadores, tem a capacidade de interagir com o sistema operativo naquilo que é a criação/destruição de grupos. O mesmo também consegue

criar/remover ficheiros, apesar de ser, desta vez, na pasta **privates** da memória referida.

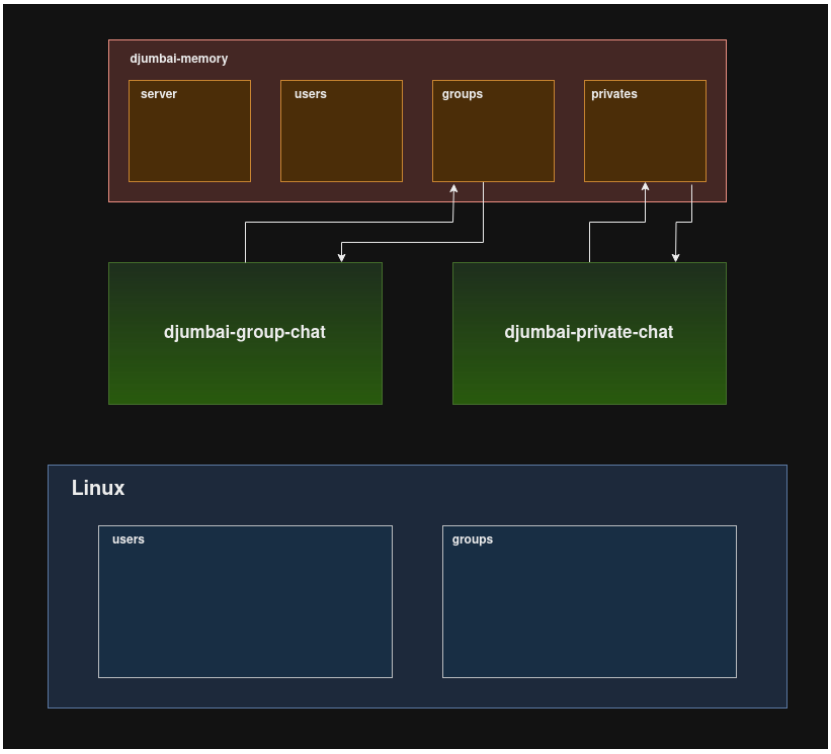


Figura 2: **Diagrama Arquitetural II**

Através do segundo diagrama, conseguem-se perceber as interações entre dois módulos do serviço e os ficheiros criados, previamente, por dois dos três serviços anteriormente explicados. Em primeiro lugar, ao relacionar-se com a comunicação em grupos privados de conversação, o módulo **djumbai-group-chat** consegue interagir com os ficheiros presentes na pasta **groups**, presente na pasta de memória do serviço. Deste modo, o mesmo permite ler e escrever nos respetivos ficheiros. Por outro lado, o módulo **djumbai-private-chat** ao permitir, aos utilizadores, comunicarem entre si (dois a dois), fornece mecanismos de leitura e escrita em ficheiros presentes na pasta **privates**.

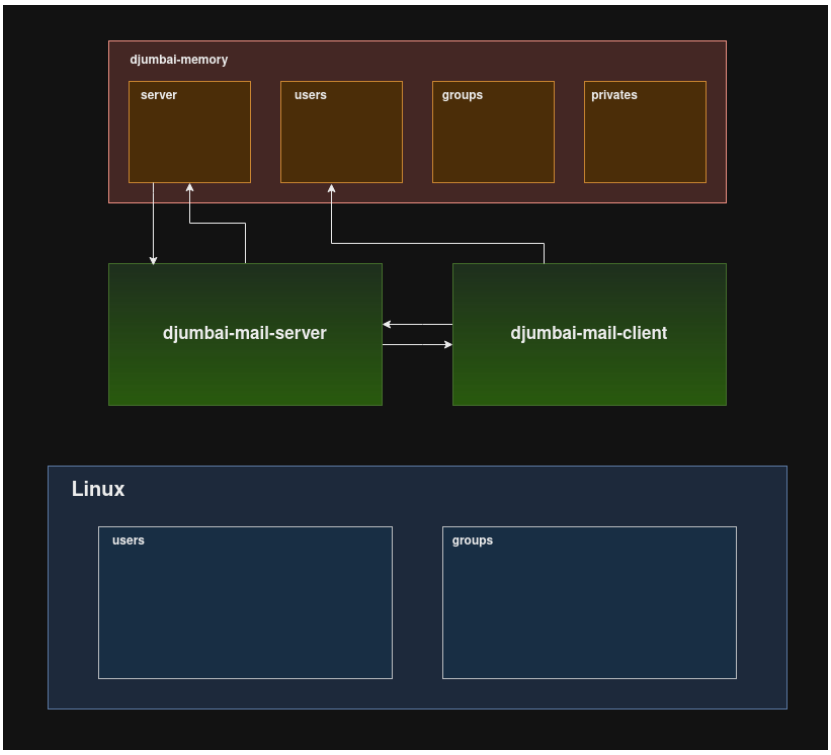


Figura 3: **Diagrama Arquitetural III**

Em último lugar, surgem as interações estabelecidas pelos dois últimos módulos. O módulo **djumbai-mail-server**, ao ser responsável por guardar as mensagens, enviadas de forma assíncrona, e enviá-las ao devido destinatário, tem a capacidade de escrever e ler nos/dos ficheiros presentes na pasta **server**, presente na memória do serviço. Por outro lado, o módulo **djumbai-mail-client** interage com os ficheiros presentes na pasta **users**, uma vez que guarda as mensagens já lidas, de cada utilizador. De forma a garantir a operabilidade do serviço, os dois módulos mencionados também comunicam entre si, através do modelo **client - server**.

# Capítulo 3

# Segurança

No presente capítulo, explicam-se as decisões tomadas relativamente à segurança do serviço. Justificam-se, assim, as escolhas feitas em relação aos donos e às permissões de cada objeto do sistema de ficheiros (**djumbai-memory**) e às permissões de execução de cada um dos módulos, referindo-se, também, o serviço de *log* implementado.

Em primeiro lugar, os três módulos responsáveis pelas funcionalidades de gestão de utilizadores e grupos apenas podem ser executados por algum administrador do sistema, ou seja, algum utilizador que pertença ao grupo **sudo**. Isto garante que nenhum utilizador comum possa interferir diretamente com o sistema operativo, reforçando a segurança do serviço e as restrições impostas sob os diversos utilizadores do *software* desenvolvido.

Em segundo lugar, sob todoos os módulos executados no processo de conversação, existe uma restrição de execução, podendo apenas recorrer aos mesmos utilizadores que tenham sido inseridos no grupo **djumbai** (à exceção do módulo **djumbai-mail-server**). Isto faz com que haja um processo de filtragem dos diversos utilizadores do serviço, reforçando, assim, a segurança do **djumbai**. Por outro lado, o módulo **djumbai-mail-server** apenas pode ser executado pelo utilizador **djumbai-server**, utilizador esse que é criado aquando a instalação do serviço.

Em teceiro lugar, reitera-se a importância de se restringir o acesso aos ficheiros presentes na pasta **djumbai-memory**. Na diretoria **server**, apenas pode haver acesso por parte dos membros do grupo **djumbai-server**, grupo esse formado única e exclusivamente pelo utilizador responsável por correr, em *background*, o módulo **djumbai-mail-server**. Por outro lado, cada um dos ficheiros presentes na pasta **users** apenas pode ser acedido pelo utilizador relativo a esse mesmo ficheiro. Em relação aos ficheiros presentes em **groups**, apenas há o acesso ao mesmo por parte dos utilizadores que pertencem ao grupo respetivo. Em último

lugar, os ficheiros presentes na diretoria **private** apenas podem ser acedidos, cada um deles, pelos dois utilizadores participantes do respetivo *chat*.

Em último lugar, explica-se o sistema de *log* desenvolvido e inserido no serviço de conversação. Todas as ações que recorrem ao comando **sudo** são devidamente guardadas num ficheiro, denominado **djumbai.log**. Este registo detalhado permite guardar as diferentes ações sensíveis realizadas e fornece uma trilha de auditoria para monitorizar o uso dos serviços mais restritos. O ficheiro de *log* inclui informações, como a data e hora da ação, o módulo sob o qual foi executado o comando e uma pequena descrição da ação efetuada. Além disso, o ficheiro de *log* pode ser consultado para investigar possíveis incidentes de segurança, identificar atividades suspeitas ou analisar o histórico de uso dos módulos críticos. Deste modo, reitera-se a importância do sistema de *log*, mostrando-se essencial no serviço de troca de mensagens, ajudando a fortalecer a segurança e a integridade do sistema como um todo.

# Capítulo 4

# Reflexão

No presente capítulo, analisam-se as decisões tomadas aquando o desenvolvimento do serviço de conversação, especialmente no que diz respeito aos parâmetros funcionais e de segurança.

Primeiramente, a funcionalidade, fornecida pelo serviço, de gestão de membros e grupos do sistema operativo mostra-se essencial na capacidade de organizar e gerir o contacto com o **djumbai**. A ação de agrupar utilizadores mostra-se útil na medida em que permite acoplar utilizadores com permissões idênticas.

A decisão de restringir o acesso aos módulos e ficheiros do sistema assume-se como uma das características mais importantes naquilo que é a utilização segura do serviço, uma vez que ajuda a evitar que utilizadores não autorizados realizem operações indesejadas e acedam a ficheiros sensíveis.

Por outro lado, a modularidade e o encapsulamento do serviço são fundamentais para garantir a manutenção e a escalabilidade do sistema. A divisão do mesmo em diferentes módulos independentes apresenta-se como uma decisão fulcral, facilitando o desenvolvimento, os testes e a manutenção de cada um dos componentes, de forma individual e independente.

# Capítulo 5

# Conclusão

O desenvolvimento do serviço **djumbai** representa uma abordagem abrangente e segura para a troca de mensagens entre utilizadores locais num ambiente **Linux**. Ao longo deste trabalho prático, tomaram-se diversas decisões, visando garantir, não apenas a funcionalidade esperada do serviço, mas, também, a sua segurança e a sua integridade.

A modularidade do serviço, dividido em sete módulos independentes, permitiu uma abordagem mais organizada e escalável no desenvolvimento e manutenção do sistema. Isso facilitou a implementação de cada funcionalidade de forma individual e a realização de testes específicos em cada componente.

A ênfase na segurança foi uma prioridade desde o início do projeto. Restringir o acesso aos módulos e aos ficheiros do sistema contribuiu significativamente para a proteção dos dados e a prevenção de acessos não autorizados. Além disso, a implementação de um sistema de *log* mostrou-se muito importante na tarefa de deteção e investigação de atividades suspeitas.

No entanto, reconhece-se que ainda existem áreas de melhoria e possíveis limitações. Por exemplo, embora se tenha desenvolvido o serviço com o foco principal do tema da segurança, assume-se que possíveis vulnerabilidades podem afetar o desenvolvimento seguro e eficaz do **djumbai**. A implementação de métodos criptográficos, por exemplo, pode assumir-se como uma próxima etapa a realizar para aumentar a segurança do serviço.

Em resumo, o **djumbai** apresenta-se como um serviço eficiente e seguro, oferecendo funcionalidades úteis na troca de mensagens entre utilizadores locais de um sistema **Linux**. Todos os desafios enfrentados mostraram-se fundamentais naquilo que é o contacto com o desenvolvimento seguro de *software*. Encerra-se, assim, o relatório relativo ao trabalho prático **Djumbai: Serviço Local de Troca de Mensagens**.