

Manuel Technique - Lanceur de Commandes

Frédéric Tischler

21 décembre 2023

1 Introduction

Ce manuel technique décrit les solutions développées pour mettre en œuvre les différents modes de communication entre le lanceur de commandes et ses clients. Le système utilise des tubes de communication, une mémoire partagée et des signaux pour assurer une communication fiable et efficace.

2 Communication via Tubes de Requêtes

Le tube de requêtes permet au lanceur de recevoir des commandes des clients. Les étapes suivantes illustrent le processus de communication via le tube de requêtes :

1. Le lanceur crée un tube de requêtes nommé `/tmp/request_pipe`.
2. Chaque client crée une structure de requête contenant les noms des tubes de sortie pour la sortie standard et les erreurs.
3. Le client écrit la structure de requête dans le tube de requêtes.
4. Le lanceur lit la structure de requête depuis le tube de requêtes et traite la commande.

3 Communication via Tubes de Sortie et d'Erreurs

Le lanceur va écrire le résultat de la commande dans le tube de sortie, ou dans le tube de sortie erreur s'il y a une erreur

1. Le client crée des tubes nommés `/tmp/output_pipe_pidclient` et `/tmp/error_pipe_pidclient`.
2. Le thread du lanceur qui va exécuter la commande, redirige sa sortie standard et sa sortie erreur vers les tubes appropriés.
3. Le client va ensuite créer un thread spécifique pour lire dans le tube `/tmp/output_pipe_pidclient`, et dans le tube `/tmp/error_pipe_pidclient`, puis écrire leur contenu dans un fichier `output.txt` ou `error.txt`

4 Communication via Mémoire Partagée

La mémoire partagée est utilisée pour transmettre la file de commandes du client au lanceur. Les étapes sont les suivantes :

1. Le lanceur crée une mémoire partagée nommée `/my_shared_memory`.
2. Le client écrit la file de commandes dans la mémoire partagée.
3. Le lanceur lit la file de commandes depuis la mémoire partagée.

5 Gestion des Signaux

La gestion des signaux est mise en œuvre pour détecter la terminaison inattendue du lanceur de commandes. Un gestionnaire de signaux est défini pour intercepter les signaux `SIGINT`, `SIGTERM` et traiter correctement la situation. Pour le client, un gestionnaire de signaux est défini pour intercepter les signaux `SIGINT` pour sortir proprement du programme.

6 Conclusion

Ce manuel technique a présenté les solutions mises en place pour assurer une communication efficace entre le lanceur de commandes et ses clients. Ces mécanismes permettent une exécution fiable des commandes et une gestion appropriée des erreurs.