Project Documentation

„Remote Device Configuration Registry Management Service"

Author: Hubert Durnaś

Course: Network Programming

Date: January 24, 2025

Introduction

This project involves the development of a client-server application for managing a remote registry of device configurations within a network. The application allows users to perform operations such as:

1. Adding new configurations.
2. Retrieving existing configurations by their identifier.
3. Deleting configurations from the server.

Data is exchanged between the client and server in a structured text format, enabling straightforward parsing and handling. The system supports multiple concurrent clients and is designed to be easily extensible.

Functional Overview

Server:

- Listens for client connections on a specified port.
- Maintains configurations using a singly linked list data structure.
- Supports the following request types:
  - Add configuration
  - Retrieve configuration by ID
  - Delete configuration
- Provides informative responses to client requests.

Client:

- Offers a user interface to select and perform operations (add, retrieve, delete).
- Formats and sends appropriate requests to the server.
- Displays server responses.
- Supports multiple operations during a single session.

Implementation Details

- Language: C
- Networking: POSIX sockets for client-server communication.
- Data Structures: Singly linked list for storing configurations.
- Libraries Used:

  <sys/socket.h>, <netinet/in.h>, <arpa/inet.h> for networking

  <string.h>, <stdio.h>, <stdlib.h>, <unistd.h> for system utilities

Error Handling

- Validates received data format and request types.
- Handles malformed or incomplete requests gracefully.
- Manages edge cases such as missing configurations by ID.

Summary

The project delivers a functional and user-friendly client-server application to manage device configuration registries over a network. It fulfills both the functional and technical requirements using tools and techniques learned during the course.

All aspects of the application, from data management to communication and error handling, were implemented and tested by the author.