# pman

Architecture and design description

09.04.2021

# Contents

# 1 Introduction

This documents serves as an introduction to the architecture and design of `pman`, which is a command-line based password manager for Linux platforms. `pman` will be written in C and it revolves around managing a file-based password database that the user can manage with different commands.

Before the introduction of the architecture, the build system of `pman` is shortly described. This will include the chosen build system, build tools, some relevant compiler options, and the different linters and static analyzers used in the project.

Given the relatively simplicity `pman` as a program, the architecture is succinct and contains only a few critical architectural views. Despite this, making an architecture design is critical, as then secure design and, most importantly, secure handling of passwords can be emphasized in the construction of `pman` as early as possible.

The utilized architecture views consist of activity, logical and deployment views. The activity view was decided instead of the usual use-case view due to only real user being the user of the command-line. Logical view was an obvious decision given the security requirements described in the previous paragraph. Finally, the deployment view provides a look in to the different libraries that will be built as a part of the `pman` project.

At the very end, this document will also go in to the detailed design of `pman`, which will include detailed interfaces of the different components described in the architecture description including the component's respective interface documentations. Even though C does not support classes, to which UML is quite heavily biased to, class diagrams will be utilized in the interface descriptions.

# 2 Build tools

## 2.1 CMake build system

### 2.1.1 Clang and C17

### 2.1.2 Compiler options

### 2.1.3 Clang power tools

### 2.1.4 Integrated CMake tasks

# 3 Architecture

## 3.1 Activity view

### 3.1.1 Common tasks

### 3.1.2 Main commands

## 3.2 Logical view

## 3.3 Deployment view

# 4 Detailed design

## 4.1 Component 1

## 4.2 Component 2

## 4.3 Component 3

## 4.4 Encryption library