



**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**

BRNO UNIVERSITY OF TECHNOLOGY

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**

FACULTY OF INFORMATION TECHNOLOGY

**ÚSTAV INFORMAČNÍCH SYSTÉMŮ**

DEPARTMENT OF INFORMATION SYSTEMS

**APLIKACE PRO ŘÍZENÝ PŘÍSTUP KE VZDÁLENÝM  
DOKUMENTŮM PRO GNU/LINUX**

APPLICATION FOR CONTROLLED ACCESS TO REMOTE DOCUMENTS FOR GNU/LINUX

**BAKALÁŘSKÁ PRÁCE**

BACHELOR'S THESIS

**AUTOR PRÁCE**

AUTHOR

**JAN BERNARD**

**VEDOUCÍ PRÁCE**

SUPERVISOR

**RNDr. MAREK RYCHLÝ, Ph.D.**

**BRNO 2020**

## Zadání bakalářské práce



Student: **Bernard Jan**

Program: Informační technologie

Název: **Aplikace pro řízený přístup ke vzdáleným dokumentům pro GNU/Linux**  
**Application for Controlled Access to Remote Documents for GNU/Linux**

Kategorie: Operační systémy

Zadání:

1. Seznamte se s požadavky týkající se zabezpečení přístupu k dokumentům v projektu Validované datové úložiště (VDU). Prozkoumejte možnosti virtuálních souborových systémů a integrace aplikací do desktopového prostředí v operačním systému GNU/Linux.
2. Navrhněte klientskou aplikaci pro VDU, která se připojí k úložišti a bude zpřístupňovat obsah úložiště pod zadaným přístupovým klíčem jako soubor virtuálního souborového systému s řízeným přístupem a správou verzí. Navrhněte automatické testy požadovaných vlastností takové aplikace.
3. Po konzultaci s vedoucím navrženou aplikaci implementujte včetně automatických testů.
4. Výsledek popište, vyhodnoťte a zveřejněte jako open-source.

Literatura:

- Interní dokumentace projektu Validované datové úložiště.
- VANGOOR, Bharath Kumar Reddy; TARASOV, Vasily; ZADOK, Erez. To FUSE or Not to FUSE: Performance of User-Space File Systems. In: 15th USENIX Conference on File and Storage Technologies (FAST 17). 2017, s. 59-72. Dostupné z: [https://www.usenix.org/conference/fast17/technical-sessions/presentation/vangoor]
- BURIHABWA, Dorian, et al. SGX-FS: Hardening a File System in User-Space with Intel SGX. In: 2018 IEEE International Conference on Cloud Computing Technology and Science (CloudCom). IEEE, 2018, s. 67-72. Dostupné z: [https://doi.org/10.1109/CloudCom2018.2018.00027]
- xdg-utils. *freedesktop.org* [online], 2019 [cit. 2020-10-26]. Dostupné z: [https://freedesktop.org/wiki/Software/xdg-utils/]

Pro udělení zápočtu za první semestr je požadováno:

- Body 1, 2 a rozpracovaný bod 3.

Podrobné závazné pokyny pro vypracování práce viz <https://www.fit.vut.cz/study/theses/>

Vedoucí práce: **Rychlý Marek, RNDr., Ph.D.**

Vedoucí ústavu: Kolář Dušan, doc. Dr. Ing.

Datum zadání: 1. listopadu 2020

Datum odevzdání: 12. května 2021

Datum schválení: 27. října 2020

## Abstrakt

Práce se zaměřuje na synchronizaci a životní cyklus souborů stažených do uživatelského počítače ze vzdáleného validovaného datového uložště. Z analýzy trhu vyplynulo, že na trhu je velký výběr aplikací, ale rozdíly mezi nimi jsou relativně velké. Na základě požadavků a stanovené komunikace s validovaným datovým uložštěm byla navržena a implementována aplikace pro systém GNU/Linux. Testování aplikace probíhalo na dvou vybraných Linuxových distribucích s mock serverem zastupující vzdálené uložště.

## Abstract

The aim of this thesis is synchronization and life cycle of files downloaded to user's computer from validated data storage. Based on market analysis there is large selection of applications in market but there are relatively large differences along them. Application was designed and implemented for GNU/Linux according to requirements and given validated data storage communication. The application was tested on two selected Linux distributions with mock server representing remote storage.

## Klíčová slova

Validované datové uložště, VDU, Linux, souborový systém, HTTP, FUSE, C++, synchronizace, soubor, dokument, oprávnění

## Keywords

Validated data storage, VDS, Linux, filesystem, HTTP, FUSE, C++, synchronization, file, document, permission

## Citace

BERNARD, Jan. *Aplikace pro řízený přístup ke vzdáleným dokumentům pro GNU/Linux*. Brno, 2020. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce RNDr. Marek Rychlý, Ph.D.

# Aplikace pro řízený přístup ke vzdáleným dokumentům pro GNU/Linux

## Prohlášení

Prohlašuji, že jsem tuto bakalářskou práci vypracoval samostatně pod vedením pana RNDr. Marka Rychlého, Ph.D. a uvedl jsem všechny literární prameny, publikace a další zdroje, ze kterých jsem čerpal.

.....

Jan Bernard  
5. května 2021

## Poděkování

Rád bych poděkoval vedoucímu práce, doktorovi Markovi Rychlému, za vynikající vedení práce a jeho podnětné poznatky a rady. Také bych rád poděkoval všem svým kolegům, se kterými jsem procházel celé bakalářské studium. A v neposlední řadě děkuji všem svým přátelům a rodině za podporu a pochopení.

# Obsah

<b>1</b>	<b>Úvod</b>	<b>3</b>
<b>2</b>	<b>Motivace a existující řešení</b>	<b>4</b>
2.1	Současná řešení . . . . .	4
2.2	Google Drive . . . . .	5
2.3	Dropbox . . . . .	5
2.4	pCloud . . . . .	5
2.5	Syncthing . . . . .	6
2.6	Shrnutí . . . . .	6
<b>3</b>	<b>Návrh řešení a použité technologie</b>	<b>7</b>
3.1	Případy užití . . . . .	7
3.2	Architektura . . . . .	8
3.2.1	Diagram tříd . . . . .	9
3.3	Technologie . . . . .	11
3.3.1	GNU/Linux . . . . .	11
3.3.2	C++/C . . . . .	11
3.3.3	OpenAPI . . . . .	11
3.3.4	HTTP . . . . .	11
3.3.5	Mock server . . . . .	11
3.3.6	Souborový systém v uživatelském prostoru (FUSE) . . . . .	11
3.3.7	DBus . . . . .	11
3.3.8	Keyring . . . . .	11
<b>4</b>	<b>Popis implementace a nasazení aplikace</b>	<b>12</b>
4.1	Definice API . . . . .	12
4.2	Mock server . . . . .	12
4.3	Vyžívané cesty/soubory v Linuxové adresářové struktuře . . . . .	12
4.3.1	Konfigurační soubor . . . . .	12
4.3.2	SQLite databáze . . . . .	12
4.4	Autentifikace . . . . .	12
4.5	Práce se soubory . . . . .	12
4.5.1	Otevření souboru v příslušné aplikaci . . . . .	12
4.6	Nasazení aplikace . . . . .	12
4.6.1	DPKG . . . . .	12
4.6.2	RPM . . . . .	12
4.6.3	Sestavení ze zdrojových souborů . . . . .	12

<b>5</b>	<b>Testovací sada</b>	<b>13</b>
5.1	Google Test . . . . .	13
<b>6</b>	<b>Závěr</b>	<b>14</b>
	<b>Literatura</b>	<b>15</b>
<b>A</b>	<b>Obsah přiloženého média</b>	<b>17</b>

# Kapitola 1

## Úvod

V době vysokorychlostního a stabilního internetu je využívání vzdálených uložišť běžnou záležitostí. Pro většinu uživatelů je to nástroj pro jednoduchou synchronizaci dat napříč několika zařízeními od mobilního telefonu po stolní počítač. Pokud vezmeme v úvahu pouze velká datová centra, jedná se pravděpodobně o nejlevnější a nejspolehlivější způsob uchování dat, protože tyto centra mají velkou kapacitu úložného prostoru a také zálohy na softwarové i hardwarové úrovni. Pod pojmem vzdálené uložení si nemusíme představit jen velká datová centra se stovkami serverů, může se jednat o relativně malé uložení ve firmě nebo dokonce o osobní/domácí řešení. Pro využívání osobních/firemních uložení se lidé uchylují v případech, je-li třeba uložit osobní nebo určitým způsobem citlivá data, která by nemohla být poskytnuta třetí straně. Nabízená řešení také nemusí poskytovat potřebnou funkcionalitu nebo jejich finanční model nesplňuje zákaznická kritéria.

V průběhu let byly vyvinuty protokoly řešící sdílení souborů jako FTP, NFS a mnohé další. Sdílení souborů je komplexní záležitostí a obsahuje velké množství parametrů. Každý protokol se zaměřuje jen na určité parametry jako zabezpečení a spolehlivost přenosu, oprávnění pro jednotlivé uživatele a soubory, synchronizace modifikovaných souborů apod nebo pojme daný parametr odlišným způsobem. V dnešní době je většina komerčně využívaných aplikací vystavěna na proprietárních protokolech nebo na protokolech umožňující obecnější použití. Jedním z nejpoužívanějších aplikačních protokolů bude Hypertext Transfer Protocol neboli HTTP.

Cílem této práce je prozkoumat aktuální nabídku a možnosti na poli vzdálených uložení a navrhnout aplikaci pro řízený přístup ke vzdáleným dokumentům pro platformu GNU/Linux. Aplikace bude zaměřena na řízený životní cyklus souborů s vynucenými oprávněními pro jednotlivé soubory daná vzdáleným uložení. Komunikace skrze HTTP protokol byla definována externím zadavatelem. Zdrojové kódy jsou dostupné na veřejném Github repozitáři.<sup>1</sup>

---

<sup>1</sup><https://github.com/PlayerBerny12/VUT-IBT-Code>

## Kapitola 2

# Motivace a existující řešení

Popularita a využití vzdálených uložišť roste, protože lidé využívají více zařízení a potřebují mezi nimi jednoduchou synchronizaci nebo na jednom souboru potřebuje spolupracovat více lidí. Dalším podnětem využívání vzdálených uložišť je většího množství dat a potřeba tyto data efektivně sdílet a bezpečně uložit. Postupně se digitalizují další systémy například ve státní správě nebo dalších podnikatelských sektorech.

Na trhu je velké množství služeb zaměřujících se na různé klientské potřeby. Diverzita nabídky je až překvapivě velká. Většina se zaměřuje na ukládání a sdílení dat ve svém ekosystému, které jsou určené pro široké použití. Pokud jsou požadavky specifické, tak možnost vlastní konfigurace není možná. Pro částečnou úpravu nebo automatizaci procesů lze využít poskytované API, ve většině případů se jedná o variantu REST API. Poslední možností je vystavení obdobné služby z několika existujících aplikací nebo vytvoření nové.

V kontextu této práce jsou požadavky následující:

- Dodržování oprávnění i na klientském systému
  - read-only
  - write-only
  - read/write
- Životní cyklus souboru (stažení)
  - stažení
  - případná modifikace
  - smazání souboru po vypršení platnosti
- Autentifikace
  - uživatelské jméno a klientský certifikát
  - uživatelské jméno a heslo

### 2.1 Současná řešení

Pro bližší analýzu byl vybrán vzorek služeb a programů obsahující velké ekosystémy služeb po open source aplikace. Zkoumáno bylo několik parametrů, jako poskytovaná funkcionality pro jednotlivce a firmy, cena, integrace s ostatními systémy a aplikacemi apod. Obecně



nelze jednoznačně určit nejlepší uložení, ale je možné poukázat na rozdílné vlastnosti a vyzdvihnout kladné. Koncový uživatel se následně může rozhodnout dle svých potřeb.

## 2.2 Google Drive

Pro synchronizaci obsahu z Google Drive na osobní počítač se využívá aplikace pojmenovaná Google File Stream. Je možné ji provozovat pouze na systémech Windows a Mac OS.[5] Pro Linuxové distribuce lze využít například neoficiální open source aplikaci Google Drive OCamlFUSE<sup>1</sup> využívající technologii FUSE, která bude popsána v následující kapitole 3.3.6. Google Disk poskytuje plně funkční webové rozhraní, ve kterém je možné dokumenty přímo upravovat bez nutnosti stahování. Problém nenastal ani v případě konkurenčních Microsoft Office dokumentů.

Google Workspace (dříve Google Suite) je možné pořídit v několika balíčcích. Například balíček Business Standard nabízí 2 TB uložení za 10,40 EUR měsíčně. Balíček obsahuje další služby jako firemní email a schůzky až o 150 účastnících na Google Meet.[9]

Každý uživatel má vlastní disk s omezenou kapacitou podle balíčku. Na disku lze vytvářet standardní složkovou hierarchii a je možné sdílet jednotlivé soubory nebo obsah složek s ostatními Google uživateli. Verze Enterprise umožňuje vytváření dalších disků, na které je možné přiřadit seznam uživatelů. Každý uživatel má nastavenou jednu z 6 rolí, které vymezují jeho možnosti. K dispozici je API, která pokrývá veškeré možnosti webového rozhraní jako vytvoření souboru, sdílení atd.[8]

## 2.3 Dropbox

Dropbox má oficiální balíčky své aplikace pro Ubuntu a Fedoru, případně je možné si aplikaci zkompilovat ze zdrojových souborů. Podpora Windows a Mac OS je samozřejmostí. Z práce „Personal Cloud Storage Benchmarks and Comparison“ lze vyčíst, že Dropbox se zaměřuje na efektivitu přenosu a minimální zatížení sítě. Využívá menší množství TCP spojení oproti jeho konkurentům a také stejně jako Google Drive komprimuje data před odesláním.[2] Toto řešení nemusí dosahovat nejvyšší výkonosti, ale nezatěžuje tolik infrastrukturu.

Webové rozhraní je více orientováno na správu souborů a jejich historii. Nabízí jednoduché obnovení smazaných souborů nebo návrat k předchozí verzi. Nenabízí tolik možností jako Google Drive, ale upravovat dokumenty ve webovém prohlížeči lze také. Na výběr je mezi Microsoft Office Online nebo Google Workspacce. Sdílení souborů a složek je možné i s uživateli, kteří nemají Dropbox účet.

Balíček Plus za 9,99 EUR měsíčně dává uživateli přístup ke 2 TB uložení. Firemní balíčky obsahují rozšíření pro lepší správu uživatelů, vytváření skupin a admin panel/konzole. Dropbox má také API nabízející dostatečnou funkcionalitu pro případnou integraci s jinými systémy apod.[6]

## 2.4 pCloud

Poskytovatel pCloud podporuje nejpoužívanější platformy, a to včetně mobilních. Za 9,99 EUR měsíčně dostane uživatel 2 TB uložení. Jako jediný z poskytovatelů v analýze nabízí

---

<sup>1</sup><https://github.com/astura/google-drive-ocamlfuse>

balíček s jednorázovou platbou za cenu 350 EUR. Jedná se o identický balíček, pouze forma platby se liší. Všechna data jsou přenášena přes šifrovaný kanál a všechny kopie souborů na pěti různých serverech jsou šifrovaná 256bitovým AES klíčem.[11]

Webové rozhraní je jednoduché, avšak oproti konkurentům má méně funkcí. Lze zobrazit náhled dokumentů, ale upravovat je nelze. Každý soubor má tzv. revize a je možné obnovit obsah souboru na vybranou revizi. Revize jsou uchovány po dobu jednoho roku. Soubory je možné sdílet i s uživateli, kteří nemají účet u pCloudu.

Pro firemní zákazníky je nabízený rozšířený management uživatelů a monitoring s logy aktivity jednotlivých uživatelů.

## 2.5 Syncthing

Open source aplikace Syncthing synchronizuje soubory peer-to-peer. Nejedná se o čistou peer-to-peer architekturu, protože jeden z uzlů může být nastaven jako server a veškeré data budou synchronizována vůči tomuto uzlu. Syncthing je možné provozovat na většině dnešních systémů jako Windows, Linux, BSD a Mac OS.[12]

Aplikace poskytuje webové rozhraní, které je určeno pouze pro nastavení parametrů jako discovery protokol pro objevování ostatních uzlů, jaké složky mají být synchronizovány nebo kolik verzí jednotlivých souborů má být uchováváno a mnoho dalšího. Velká přizpůsobivost umožňuje upravit fungování aplikace vlastním potřebám, na druhou stranu bude náročné udržovat systém s větším množstvím uživatelů vystavěný na této aplikaci. Jedná se tedy spíše o domácí řešení.

## 2.6 Shrnutí

Trendy na poli vzdálených uložišť určují velké firmy jako Google nebo Microsoft. Aplikace One Drive od Microsoftu nebyla blíže analyzovaná, protože se jedná o službu pouze pro platformu Windows. Důkazem tohoto jevu je například integrace aplikací obou těchto firem do webového rozhraní Dropboxu.

Všechny služby mají webové rozhraní, ke kterému je aplikace synchronizující obsah uložště občas braná jako doplněk. Sdílení nebo obnova předchozí verze souboru je vždy možná ve webové aplikaci, ale ne vždy jsou tyto akce dostupné v desktopové aplikaci.

Aplikace na jejíž základě je vytvořena tato práce nebude klasickou synchronizační aplikací. Bude se pracovat jen s jednotlivými soubory, na kterých bude uživatel chtít pracovat (zobrazit nebo upravit obsah). Nebude možné synchronizovat celé uložště, ale vždy jen jeden specifický soubor. Jedná se tedy spíše o náhradu možnosti upravovat soubor přímo ve webovém prohlížeči jako to umožňuje například Google Drive.

## Kapitola 3

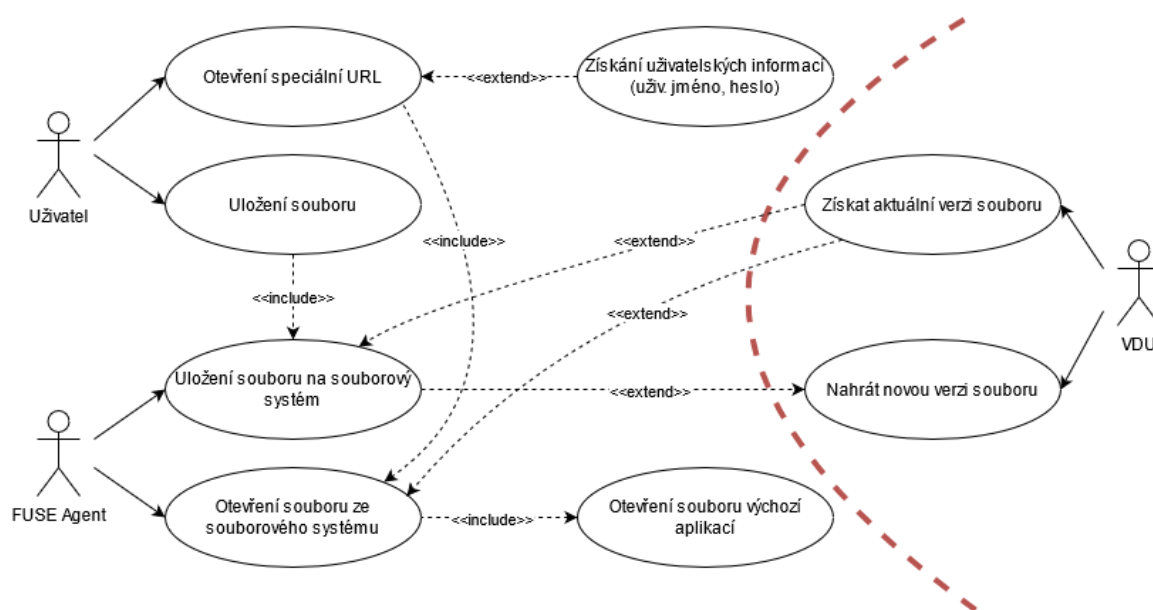
# Návrh řešení a použité technologie

Prvním krokem navrhovací fáze bylo seznámení se zadáním a požadavky. Externí zadavatel dodal podrobnou dokumentaci REST API pro validované datové uložště, která byla stěžejním dokumentem při návrhu. API obsahuje funkce pro ověření spojení, autentifikaci a práci se soubory. VDU nebylo při vzniku této práce k dispozici a proto byl pro potřeby vývoje vytvořen mock server na základě API dokumentace.

Pro vynucení práv jednotlivých souboru je standartní souborový systém nevyhovující, protože po stažení souboru do počítače uživatele se ztrácí veškerá kontrola nad tímto souborem. Po konzultaci s vedoucím práce, dokterem Markem Rychlým, byla zvolena technologie FUSE neboli Filesystem in userspace. FUSE umožňuje vytvořit virtuální souborový systém a implementovat jednotlivé operace jako čtení, zápis atd.

### 3.1 Případy užití

Diagram případu užití zachycuje celý systém jako celek sestavený z jednotlivých funkcí a jejich návazností, které je třeba dále analyzovat a specifikovat v dalších fázích návrhu.



Obrázek 3.1: Diagram případu užití

Na diagramu 3.1 vystupují tři aktéři a to uživatel, FUSE agent/démon a VDU. Implementace VDU není obsahem této práce, proto je oddělena červenou přerušovanou čarou. V diagramu není zanesená například autentifikace, na které závisí veškerá Komunikace s VDU. Diagram by byl poté nepřehledný, proto byla tato část vypuštěna.

Všechny akce jsou spouštěny přímo uživatelem, některé vyvolají spuštění funkce FUSE démona. Jedná se například o uložení, otevření nebo přejmenování souboru ve virtuálním souborovém systému. Speciální případ nastává při otevření url s vlastním protokolem, který má spustit aplikaci, jež je k danému protokolu v systému přiřazena. Na tuto URL bude uživatel přesměrován z webového rozhraní VDU. Příkladem protokolu spouštějící aplikaci v systému může být `mailto`, využívaný často u webových aplikací. Přesměrování na adresu s tímto protokolem otevře správce elektronické pošty.

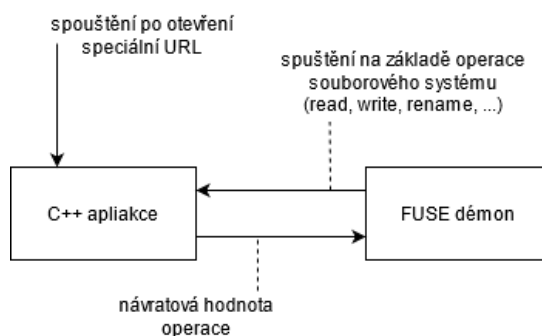
K námi definovanému protokolu pojmenovanému „vdu“ je třeba mít v systému přiřazenou aplikaci. Na základě přístupového tokenu uloženého v URL (`vdu://<přístupový token>`) se spustí daná aplikace, která stáhne žádaný soubor z VDU na virtuální souborový systém. Na základě této znalosti byly uskutečněné zásadní rozhodnutí při návrhu architektury.

## 3.2 Architektura

Zvažovány byly dvě implementační varianty, jedna verze pracoval pouze s jednou aplikací a to byl FUSE démon. Druhá varianta rozdělila funkčnost do dvou aplikací, jednou z nichž byl FUSE démon a druhou byla klasická aplikace napsaná například v C++. V tomto případě by samostatná aplikace obsluhovala komunikaci s VDU a byla by spouštěna na základě volání FUSE démona nebo uživatele.

První variantu by šlo zpracovat dvěma podobnými způsoby vedoucí ke stejnému řešení. FUSE démon je napsaný v jazyce C, takže první zvažovaná varianta byla implementace celá aplikace jedním jazyce. Druhou možností by bylo vytvoření C++ knihovny, která by měla interface pro jazyk C, aby mohl vytvořenou knihovnu využít FUSE démon. Nejedná se však o nijak kriticky náročnou aplikaci na výkon, takže by byla varianta s C++ knihovnou upřednostňována.

Se znalostí z diagramu případu užití, že je třeba mít registrovanou aplikaci, která bude spuštěna po otevření speciální URL. I kdybychom vzali v úvahu místo URL soubor se speciální koncovkou (např. `*.vdu`) obsahující přístupový token, který by si uživatel stáhnul z VDU, tak k otevření tohoto souboru je třeba mít také přiřazenou aplikaci, která by danou akci obsloužila. Démon běžící na pozadí po celou dobu běhu systému, nemůže být přiřazen jako obsluhující aplikace otevření speciální URL. Na základě těchto okolností byla vybrána varianta se dvěma aplikacemi.



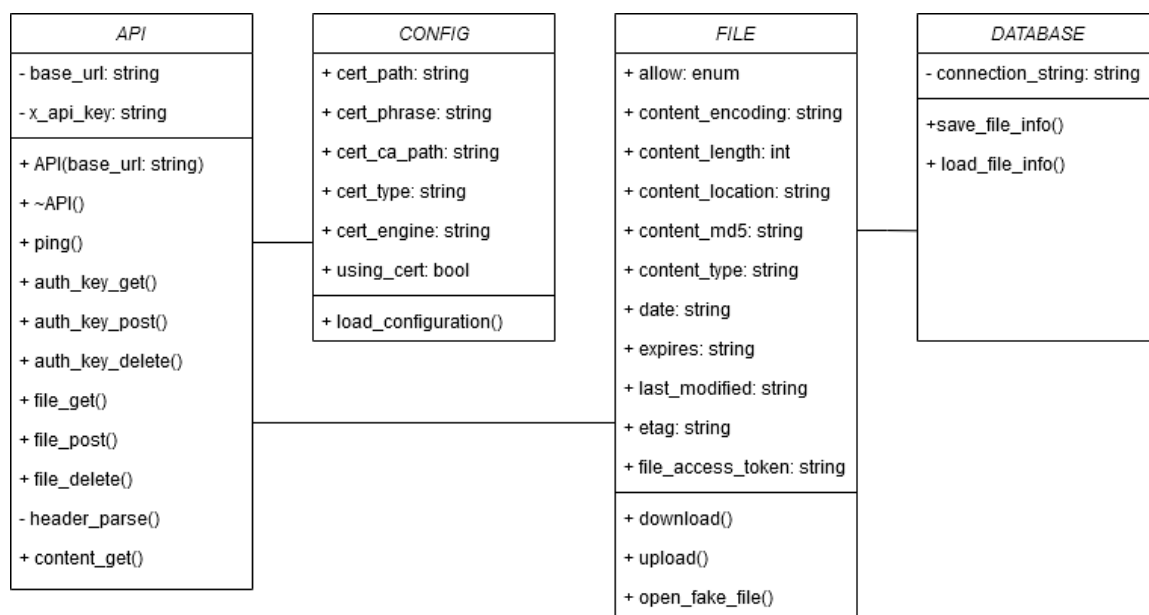
Obrázek 3.2: Architektura

Pokud FUSE démon má vykonat akci, pro kterou nemá dostatečné informace, spustí jako další subprocess druhou hlavní aplikaci a počká na návratovou hodnotu, na základě které se rozhodne, jakým způsobem onu akci dokončí. Může se jednat o případy jako jestli je soubor určený jen pro četní nebo i zápis, nebo jestli daný soubor už není validní, protože vypršel datum expirace apod.

V této variantě je FUSE démon velmi minimalistický a zbytek funkcionality je implementován do druhé aplikace, která je spouštěna jen na vyžádání, narozdíl od neustále běžícího démona. Démon tedy po celou dobu běhu zabírá méně místa, protože nemusí mít načtená v paměti všechny potřebné knihovny. Na druhou stranu potřebné knihovny mohou být v paměti namapované, protože jsou využívány jinými aplikacemi. Démon musí být spuštěný s právy superuživatele `root` a menší množství kódu spuštěného s vysokými oprávněními implikuje menší náchylnost na bezpečnostní chyby. Démon bude totiž pouštět druhou aplikaci s oprávněními přihlášeného uživatele.

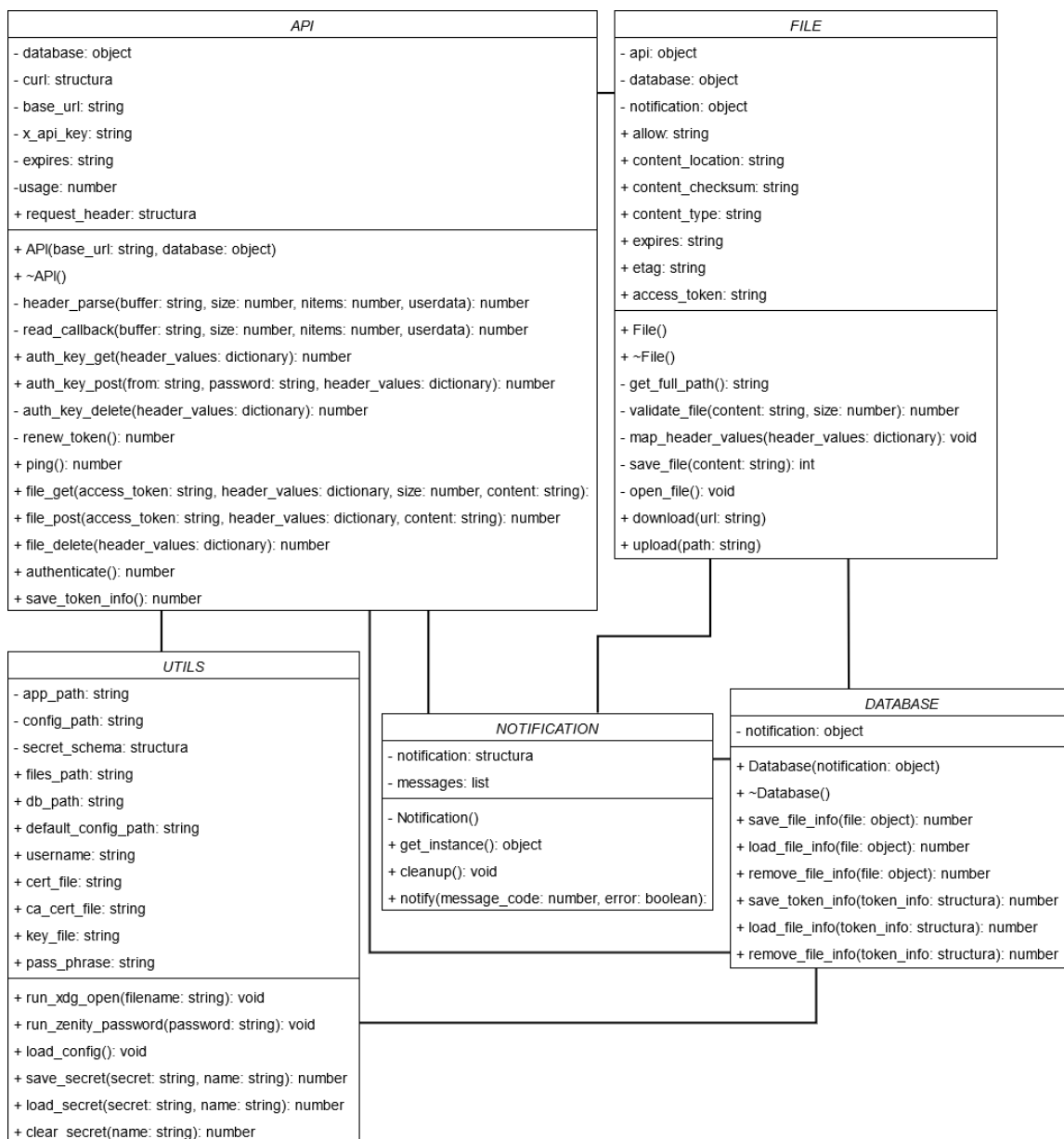
### 3.2.1 Diagram tříd

Aby implementace byla co nejjednodušší a nebylo třeba již uvažovat nad dalšími rozhodnutími, byl vytvořen diagram tříd 3.3, který relativně podrobně vizualizuje jaké třídy s atributy a metodami je třeba implementovat. Jedná se pouze o návrh, který se v průběhu vývoje bude dynamicky měnit. Pokud je návrh kvalitní a programátoři se ho sanží dodržet, neměl by být výsledek zásadně rozdílný.



Obrázek 3.3: Diagram tříd - návrh

Na obrázku 3.4 je možné vidět diagram tříd naimplementovaného programu. Na první pohled vypadá velmi rozdílně, po bližším prozkoumání lze vidět jednu novou třídu oproti návrhu. Jedná se třídu zobrazující notifikace pro uživatele, takže nemá vliv na hlavní funkcionality.



Obrázek 3.4: Diagram tříd - implementace

### **3.3 Technologie**

#### **3.3.1 GNU/Linux**

#### **3.3.2 C++/C**

#### **3.3.3 OpenAPI**

#### **3.3.4 HTTP**

[\[7\]](#) [\[4\]](#)

#### **3.3.5 Mock server**

[\[1\]](#)

#### **3.3.6 Souborový systém v uživatelském prostoru (FUSE)**

[\[13\]](#) [\[3\]](#)

#### **3.3.7 DBus**

#### **3.3.8 Keyring**

[\[10\]](#)

## Kapitola 4

# Popis implementace a nasazení aplikace

- Seznam použitých knihoven

### 4.1 Definice API

### 4.2 Mock server

### 4.3 Vyžívané cesty/soubory v Linuxové adresářové struktuře

#### 4.3.1 Konfigurační soubor

#### 4.3.2 SQLite databáze

### 4.4 Autentifikace

### 4.5 Práce se soubory

#### 4.5.1 Otevření souboru v příslušné aplikaci

[14]

### 4.6 Nasazení aplikace

#### 4.6.1 DPKG

#### 4.6.2 RPM

#### 4.6.3 Sestavení ze zdrojových souborů



## Kapitola 5

# Testovací sada

- Testy pro funkce třídy API

### 5.1 Google Test

## Kapitola 6

## Závěr

# Literatura

- [1] BLOOM, J. *MockServer* [online]. Mock Server, 2020 [cit. 2021-04-28]. Dostupné z: <https://www.mock-server.com/#what-is-mockserver>.
- [2] BOCCHI, E., DRAGO, I. a MELLIA, M. Personal Cloud Storage Benchmarks and Comparison. *IEEE Transactions on Cloud Computing*. 2017, sv. 5, č. 4, s. 751–764. DOI: 10.1109/TCC.2015.2427191. Dostupné z: <https://ieeexplore.ieee.org/abstract/document/7096995>.
- [3] BURIHABWA, D., FELBER, P., MERCIER, H. a SCHIAVONI, V. SGX-FS: Hardening a File System in User-Space with Intel SGX. In: *2018 IEEE International Conference on Cloud Computing Technology and Science (CloudCom)*. 2018, s. 67–72. Dostupné z: <https://ieeexplore.ieee.org/document/8590996>.
- [4] CARLUCCI, G., DE CICCIO, L. a MASCOLO, S. HTTP over UDP: An Experimental Investigation of QUIC. In: *Proceedings of the 30th Annual ACM Symposium on Applied Computing*. Association for Computing Machinery, 2015, s. 609–614. DOI: 10.1145/2695664.2695706. ISBN 9781450331968. Dostupné z: <https://doi.org/10.1145/2695664.2695706>.
- [5] *Deploy Google Drive for desktop* [online]. Google, 2021 [cit. 2021-04-21]. Dostupné z: <https://support.google.com/a/answer/7491144?hl=en>.
- [6] *Dropbox* [online]. Dropbox, 2021 [cit. 2021-04-21]. Dostupné z: <https://www.dropbox.com/?landing=dbv2>.
- [7] FIELDING, R. a RESCHKE, J. *Hypertext Transfer Protocol (HTTP/1.1): Semantics and Content* [Requests for Comments]. RFC Editor, červen 2014 [cit. 2021-04-25]. DOI: 10.17487/RFC7231. Dostupné z: <https://www.rfc-editor.org/info/rfc7231>.
- [8] *Google Drive for developers - API Reference* [online]. Google, 2021 [cit. 2021-04-21]. Dostupné z: <https://developers.google.com/drive/api/v3/reference>.
- [9] *Google Workspace* [online]. Google, 2021 [cit. 2021-04-21]. Dostupné z: <https://workspace.google.com/>.
- [10] *What is: Linux keyring, gnome-keyring, Secret Service, and D-Bus* [online]. RTFM: Linux, DevOps, and system administration, 12. července 2019 [cit. 2021-05-02]. Dostupné z: <https://rtfm.co.ua/en/what-is-linux-keyring-gnome-keyring-secret-service-and-d-bus/>.
- [11] *PCloud* [online]. pCloud, 2021 [cit. 2021-04-21]. Dostupné z: <https://www.pcloud.com/eu>.

- [12] *Syncthing* [online]. Syncthing, 2021 [cit. 2021-04-21]. Dostupné z: <https://syncthing.net/>.
- [13] VANGOOR, B. K. R., TARASOV, V. a ZADOK, E. To FUSE or Not to FUSE: Performance of User-Space File Systems. In: *15th USENIX Conference on File and Storage Technologies (FAST 17)*. 2017, s. 59–72. Dostupné z: <https://www.usenix.org/conference/fast17/technical-sessions/presentation/vangoor>.
- [14] *Xdg-utils* [online]. freedesktop.org, 2019 [cit. 2021-04-20]. Dostupné z: <https://freedesktop.org/wiki/Software/xdg-utils/>.

## **Příloha A**

### **Obsah přiloženého média**