VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

BAKALÁŘSKÁ PRÁCE

Brno, 2022 Tadeáš Zachoval

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY
A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

OCHRANA AUTORSKÝCH PRÁV ELEKTRONICKÝCH DOKUMENTŮ

COPYRIGHT PROTECTION OF ELECTRONIC DOCUMENTS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE — Tadeáš Zachoval

AUTHOR

VEDOUCÍ PRÁCE — Ing. Vlastimil Člupek, Ph.D.

SUPERVISOR

BRNO 2022

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Bakalářská práce

bakalářský studijní program **Informační bezpečnost**

Ústav telekomunikací

Student: Tadeáš Zachoval ***ID:*** 211820

Ročník: 3 ***Akademický rok:*** 2021/22

NÁZEV TÉMATU:

Ochrana autorských práv elektronických dokumentů

POKYNY PRO VYPRACOVÁNÍ:

V bakalářské práci definujte autorská práva pro elektronické dokumenty a analyzujte možnosti jejich ochran. Navrhněte aplikaci zajišťující dlouhodobou ochranu autorských práv pro elektronické dokumenty. Implementujte navrženou aplikaci a ověřte její funkčnost. Popište softwarové požadavky aplikace a ohodnoťte získanou úroveň ochrany autorských práv u elektronických dokumentů pomocí vytvořené aplikace. Získané výsledky přehledně prezentujte.

DOPORUČENÁ LITERATURA:

[1] TAYAN, Omar; KABIR, Muhammad N.; ALGINAHI, Yasser M. A hybrid digital-signature and zero-watermarking approach for authentication and protection of sensitive electronic documents. The Scientific World Journal, 2014, 2014.

[2] JING, Nan; LIU, Qi; SUGUMARAN, Vijayan. A blockchain-based code copyright management system. Information Processing & Management, 2021, 58.3: 102518.

Termín zadání: 7.2.2022 ***Termín odevzdání:*** 31.5.2022

Vedoucí práce: Ing. Vlastimil Člupek, Ph.D.

doc. Ing. Jan Hajný, Ph.D.
předseda rady studijního programu

Abstrakt

Bakalářská práce je zaměřena na autorskoprávní ochranu elektronických dokumentů. Práce se skládá z části teoretické a části praktické. Teoretická část popisuje elektronický dokument a různé typy souborů, se kterými se lze při práci na počítači setkat a považovat je za elektronické dokumenty. Stěžejní částí teoretické části je analýza různých způsobů ochrany autorských práv elektronických dokumentů. Na základě uvedené analýzy byla v praktické části vytvořena aplikace v jazyce Python. Aplikace implementuje vybrané způsoby ochrany autorských práv.

Klíčová slova

elektronický dokument, typy souborů, autorská práva elektronických dokumentů, digitální vodoznak, digitální podpis, elektronické časové razítko, blockchain, Python

Abstract

The bachelor thesis is focused on the copyright protection of electronic documents. The thesis consists of a theoretical part and a practical part. The theoretical part describes the electronic document and various types of files that can be encountered while working on a computer and consider them as electronic documents. The fundamental part of the theoretical part is the analysis of various copyright protection methods for electronic documents. In the practical part was created a Python application based on mentioned analysis. The application implements selected methods of copyright protection.

Keywords

electronic document, types of files, copyright of electronic documents, digital watermark, digital signature, electronic time stamp, blockchain, Python

ZACHOVAL, Tadeáš. *Ochrana autorských práv elektronických dokumentů*. Brno, 2022. Dostupné také z: https://www.vutbr.cz/studenti/zav-prace/detail/141357. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2022, 62 stran. Vedoucí práce Ing. Vlastimil Člupek, Ph.D.

Prohlášení autora o původnosti díla

Jméno a příjmení studenta:	Tadeáš Zachoval
VUT ID studenta:	211820
Typ práce:	Bakalářská práce
Akademický rok:	2021/22
Téma závěrečné práce:	Ochrana autorských práv elektronických dokumentů

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

V Brně dne: ----------

podpis autora

Poděkování

Děkuji vedoucímu bakalářské práce Ing. Vlastimilu Člupkovi, Ph.D. za účinnou metodickou, pedagogickou a odbornou pomoc a další cenné rady při zpracování mé bakalářské práce.

Tato práce vznikla jako součást klíčové aktivity KA6 - Individuální výuka a zapojení studentů bakalářských a magisterských studijních programů do výzkumu v rámci projektu OP VVV Vytvoření double-degree doktorského studijního programu Elektronika a informační technologie a vytvoření doktorského studijního programu Informační bezpečnost, reg. č. CZ.02.2.69/0.0/0.0/16_018/0002575.

EVROPSKÁ UNIE
Evropské strukturální a investiční fondy
Operační program Výzkum, vývoj a vzdělávání

MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY

Projekt je spolufinancován Evropskou unii.

Obsah

SEZNAM OBRÁZKŮ

SEZNAM TABULEK

Úvod

S postupem času, tak jak se vyvíjí lidstvo, tak se vyvíjí i oblast technologií. V dnešní době, kdy je přístup k novým technologiím snazší, je kladen důraz na rozšířenost jejich použití v široké škále napříč odvětvími. Tento vývoj přinesl mnoho výhod, avšak se vším dobrým jde ruku v ruce i něco špatného.

Jedním z odvětví, které se s vývojem technologií rozšířilo, je digitalizace dokumentů. Dokumentem může být jakákoliv obrazová, zvuková, písemná nebo jiným způsobem zaznamenávaná informace v analogové podobě, v případě digitálního dokumentu v podobě elektronické. Převedení dokumentů do elektronické podoby má mnoho pozitiv. Jedním takovým může být urychlení celého procesu výměny jednotlivých dokumentů mezi subjekty. V dřívějších dobách bylo nutné jednotlivé dokumenty mít ve fyzické podobě a doručit je protistranám, ostatním příjemcům. Což v případech, kdy by se jednalo například o mezinárodní smlouvy, tak by byla doba doručení několikanásobně delší než v případě dnešní doby, kdy je možné využít internet a elektronickou výměnu dokumentů. Na druhou stranu jako protiklad, že vše nemá jen dobré stránky, lze uvést, že například každý nemá možnost přístupu k počítači nebo nemá schopnosti k jeho ovládání v takové míře, aby dokázal řešit záležitosti elektronicky.

S rozšířením elektronických dokumentů bylo nutné vyřešit mnoho otázek jako například, jak docílit, že se jedná o původní dokument a nebyl nikde pozměněn nebo otázku, kdo je vlastníkem takového dokumentu a mnoho dalších. V otázce autorství je nutné prokázat, ať už například nějakým certifikátem nebo vloženým vodoznakem, že se jedná o autorovo dílo bez nutnosti dalšího hlubšího zkoumání. V dnešní době je v praxi používán například již zmíněný vodoznak nebo je také možné využívat elektronických podpisů, které jsou spojeny s konkrétní fyzickou osobou.

Bakalářská práce je rozdělena do dvou částí, v teoretické části je rozebráno, co elektronický dokument je a jaké mohou být jeho formy vyjádření. Také je zde probrán pohled na elektronický dokument z pohledu práva a v neposlední řadě jsou v práci uvedeny možné ochrany autorských práv těchto dokumentů. Praktická část práce se zaměřuje na některé možnosti ochrany autorských práv elektronických dokumentů a implementuje je ve vytvořené aplikaci. Vybranými mechanismy ochrany jsou vložení vodoznaku do vybraného obrázku, využití steganografie, časového razítka, digitálního podpisu a nastínění možné ochrany autorských práv za pomocí technologie blockchain.

1 TEORETICKÁ ČÁST

V teoretické části závěrečné práce je popsáno, co pojem elektronický dokument může znamenat. Elektronický dokument je zde rozebrán z pohledu informatiky i z pohledu právní formy a je porovnáván s dokumentem listinným. Dále teoretická část popisuje různé typy a formáty souborů, na které je možné při práci na počítači narazit a je možné je jako elektronický dokument využít. Poté se práce věnuje autorským právům, které při tvorbě dokumentů vznikají. Nakonec jsou zde rozebrány různé kryptografické způsoby ochrany těchto autorských práv.

1.1 Elektronický dokument

Na začátek je nutné vysvětlit, co dokument je a znamená. Pod pojmem dokument se každému mohou vybavit odlišné věci. Mohlo by se jednat jak o písemnou smlouvu, tak o obraz. Z hlediska zákonů není dokument nijak specifikován jedinou definicí. V různých právních předpisech je pojem jinak definován. Nejvíce obecné vymezení pojmu elektronického dokumentu je v zákonu o archivnictví a spisové službě a změně některých zákonů, který elektronický dokument vymezuje jako: „*každá písemná, obrazová, zvuková nebo jiná zaznamenaná informace, ať již v podobě analogové či digitální, která byla vytvořena původcem nebo byla původci doručena*“ (dle §2 odst. e) zákona č. 499/2004 Sb., zákon o archivnictví a spisové službě a o změně některých zákonů). [1]

Dokument má několik znaků (dle [2]):

- Informační hodnota – dokument je nositelem informace, která má hodnotu (a relevanci – dle [3])
- Stálost – dokument je neměnný a stálý
- Jazyk – dokument je vyjádřen v určitém jazyce nebo symbolice
- Strukturovanost – vnitřní struktura dokumentu je závislá na jeho určení, povaze, okolnostech vzniku a dalších parametrech
- Ucelenost – s dokumentem je jednáno jako s jednotkou
- Funkční zabarvení – funkce dokumentu, která je určující pro mnoho jeho specifických vlastností

Definici elektronického dokumentu lze také převzít z nařízení Evropského Parlamentu a Rady (Evropské unie) č. 910/2014 ze dne 23. července 2014 o elektronické identifikaci a službách vytvářejících důvěru pro elektronické transakce na vnitřním trhu a o zrušení směrnice 1999/93/ES (známé též pod názvem „nařízení eIDAS“). Toto nařízení elektronický dokument definuje jako „*jakýkoli obsah uchovávaný v elektronické podobě, zejména jako text nebo zvuková, vizuální nebo audiovizuální nahrávka*“. [4]

Ani v informatice nelze elektronický dokument nijak přesně definovat. Ale obecně se jedná o jakýkoliv dokument, resp. soubor, který byl vytvořen pomocí softwaru bez ohledu na jeho formu vyjádření. [5] Může se tak jednat o jakýkoliv soubor, který vznikl

pomocí počítače, mobilu apod. Takovýto soubor může mít tedy podobu textu, fotografie, videa, a mnoho jiných.

1.1.1 Rozdíl mezi analogovým a digitálním dokumentem

Hlavní rozdíl mezi analogovým (listinným) a digitálním (elektronickým) dokumentem je, jak už název samotný napovídá, v jejich formě vyjádření. Za digitální dokument lze označit práce vytvořené v textových editorech, digitální fotografie apod. Laicky řečeno, digitální dokument vznikl pomocí softwaru a má nehmatatelnou formu. Jeho forma je vyjádřena až pomocí dalšího hardwaru jako například stolní počítač, mobilní telefon, a jiné. Naproti tomu analogové dokumenty nemusí být nutně jen dokumenty listinné, jedná se i o dokumenty jako například film do fotoaparátu.

Jako další rozdíl je možné uvést archivaci takovýchto dokumentů. V případě dokumentů vytvořených v listinné podobě můžeme mluvit o archivaci jako o uložení dokumentu do prostor tomu určených (archivy). Pokud by se jednalo o citlivé informace, které by byly na písemnostech uvedeny, měly by být tyto prostory zabezpečeny (např. proti neoprávněnému vniknutí), aby se předešlo odcizení těchto informací. Jestliže mluvíme o uložení elektronických dokumentů po déle trvající dobu, je zde možné využít více variant uložení. Každá variace uložení má své klady i zápory. Jeden způsob archivace elektronických dokumentů je uložení do cloudových služeb. Nesmírnou výhodou (avšak i nevýhodou z důvodu bezpečnosti) tohoto způsobu je jeho dostupnost takřka odkudkoliv. Pakliže bychom chtěli uchovat digitální dokument v „relativní“ bezpečnosti a vyhnout se tak možnému zcizení dat lze využít možnosti uložení informací do uložiště (NAS, počítač, a další). Takovéto uložiště by však nemělo být nijak připojeno k internetu, eliminuje se tím pravděpodobnost úniku dat. Další možností, jak uložit elektronická data je využití přenosných nosičů. Tato možnosti byla využívaná hojně v minulosti, kdy se data vkládala na diskety (floppy disky) a posléze kompaktní disky (CD). Postupem času však tyto paměťová media přestala být dostatečně objemná a nahradily je pevné disky, paměťové karty a paměťové disky.

V případě, kdy jsou elektronické dokumenty součástí soudního či správním jednání, nesmějí být, dle nařízení eIDAS, právně znevýhodňovány, jestliže mají elektronickou podobu. [6]

1.1.2 Konverze listinného a elektronického dokumentu

V současnosti je možnost konverze listinného dokumentu do elektronického a obráceně. Ke konverzi je vždy nutné přiložit doložku o provedení konverze. [7]

Konverzí dokumentu je myšleno:

- převedení z listinného dokumentu do dokumentu v datové zprávě či datovém souboru
- převedení dokumentu v datové zprávě nebo datovém souboru do listinného dokumentu

Jestliže je požadováno, aby byly zachovány právní účinky dokumentu při jeho převedení z jedné formy vyjádření do druhé, je nutné, aby konverzi takového dokumentu provedl zákonem stanovený subjekt (dle §23 zákona č. 300/2008 Sb., zákon o elektronických úkonech a autorizované konverzi dokumentů). Jedním takovým příkladem subjektu je Czech POINT (Podací Ověřovací Informační Národní Terminál). V případě konverze dokumentů neplatí, že by jeden druh byl právně zvýhodňován či naopak. Tuto skutečnost zajišťuje §22 odstavec 2 zákona č. 300/2008 Sb., zákon o elektronických úkonech a autorizované konverzi dokumentů, který říká: „Dokument, který provedením konverze vznikl (dále jen „výstup"), má stejné právní účinky jako dokument, jehož převedením výstup vznikl (dále jen „vstup")." [8]

1.2 Soubor a jeho rozdělení

Ve světě počítačů se vyskytuje několik různých typů souborů a nelze jednoznačně říct, že existuje jen určité množství druhů. S postupem času se toto odvětví neustále rozšiřuje. Nejprve však pro určení typů souboru je nutné definovat, co si pod pojmem soubor představit. Soubor je *„množina informací (dat), které mají určité společné vlastnosti. Tyto informace jsou uloženy většinou na disku a je nějakým způsobem zajištěno, že k sobě patří. Z pohledu OS je soubor definován jménem."* [9] Mimo jiné je soubor ještě definován i příponou. Dle přípony dokážeme na první pohled odhadnout, o jaký typ souboru by se mohlo jednat. Občas také dochází k záměně termínů „formát souboru" a „typ souboru". Formát souboru definuje obsah a strukturu souboru, avšak typ souboru definuje už konkrétní typ souboru. [10] Např. při názvu souboru „Soubor1.wav" poznáme že se jedná o formát WAV (Waveform audio format) a soubor bude typu audio.

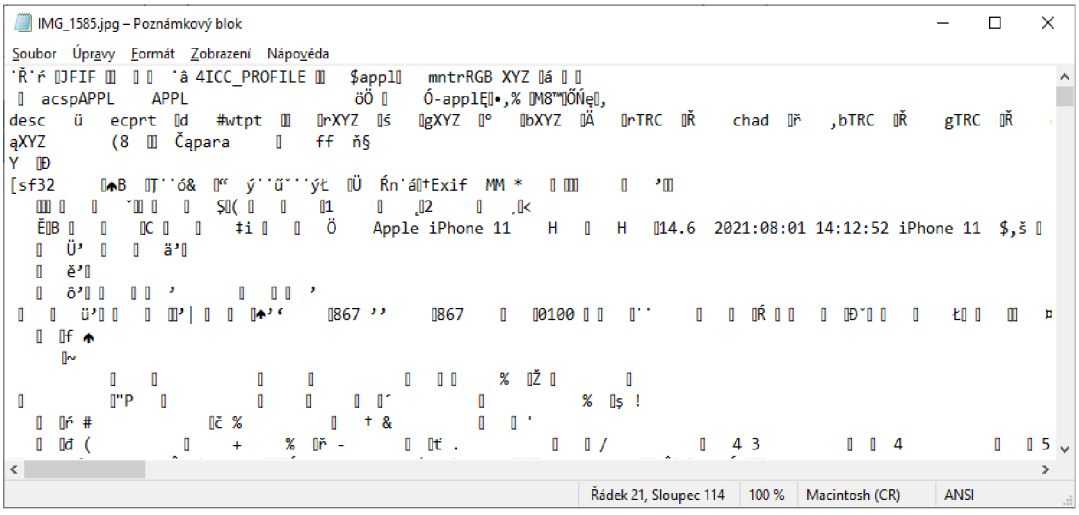
Jedním z možných rozdělení je rozdělení na binární a textové soubory. Jejich rozdíl tkví v rozdílném kódování dat. Oba typy reprezentují data jako sérii bitů (0 a 1). Avšak v textových souborech bity představují znaky, ale v bitových typech bity reprezentují různá data. [11]

1.2.1 Binární soubory

Jak bylo řečeno výše, binární soubory jsou reprezentovány bity. Při otevírání daných binárních souborů jsou zpracovávány po sekvencích bajtů (skupin po 8 bitech). Na rozdíl od souborů textových je těžké, spíše nemožné, tyto soubory rozluštit bez příslušných programů. Hlavními typy binárních souborů jsou soubory představující zvuk, obrázek, video. Mimo jiné binární soubory obsahují hlavičku. Ta je klíčem k souboru a díky ní je možné identifikovat obsah souboru. Uvnitř hlavičky můžeme najít o jakou příponu souboru se jedná, velikost souboru, či jiná další metadata. Jestliže je hlavička poškozená, či pozměna v textovém editoru, je soubor nemožné otevřít. [12]

Obrázek 1.1: Ukázka binárního souboru s formátem JFIF (JPEG File Interchange Format).

Jak je možné si povšimnout, binární soubor otevřený v textovém editoru je opravdu lidským okem nečitelný (viz obrázek 1.2). Jediné, co je možné pouhým okem rozeznat, jsou informace, že se jedná o soubor s formátem JFIF (první řádek textu), který byl vytvořen pomocí zařízení Apple iPhone 11 dne 1. srpna 2021 v 14:12:52 (přibližný prostředek textového editoru).



Obrázek 1.2: Ukázka binárního souboru v textovém editoru.

1.2.2 Textové soubory

Na rozdíl od binárních souborů jsou textové soubory méně náchylné na změnu. To je zapříčiněno tím, že textové soubory ukládají jen a pouze textová data. Výchozí znaková

sada pro tyto soubory je ASCII, avšak to zamezuje použití znaků např. pro dolar nebo euro, z toho důvodu je možné textové soubory ukládat se znakovou sadou Unicode s použitím kódování znaků UTF (Unicode Text Format).

Mezi příklady textových souborů je možné zařadit například soubory s formátem TXT (Plain Text File), RTF (Rich Text Format File) nebo XML (Extensible Markup Language). Rozdíl mezi formáty TXT a RTF je takový, že soubory s .txt příponou jsou ukládány v prostém textu, což je neformátovaný text s označeným koncem řádku (znakem EoL – End of Line). [11] Zatímco soubory s příponou .rtf jsou soubory, kde text může být formátovaný, silný, tučný, či psaný kurzívou.

Pro otevírání textových souborů slouží tzv. textové editory. V případě operačního systému Windows, jsou zde dva výchozí editory. Pro editaci TXT souboru je možné využít výchozí aplikace ve Windows – Poznámkový blok, pro tvorbu souboru RTF lze naopak použít aplikaci WordPad. Pokud je využíván operační systém macOS lze využít aplikace TextEdit. Avšak soubory s prostým textem lze otevírat i pomocí webových prohlížečů – např. Chrome.

1.3 Typy souborů

Jak již bylo řečeno výše, nelze jednoznačně určit kolik typů souborů ve skutečnosti existuje, neboť se stále s postupem času rozšiřují a každý, kdo vyvine vlastní aplikaci, může spolu s ní vytvořit nový formát a typ souborů. Soubor tak může být otevíratelný pouze danou aplikací pro kterou byl vytvořen nebo může být otevírán i vícero aplikacemi, jestliže bude podporován. Rozdělit typy souborů je možné podle několika faktorů. Mezi nejjednodušší patří rozdělení dle přípony souboru. Přípona je dvou až čtyř znaková část názvu souboru, která je oddělena tečkou na konci názvu souboru. Příkladem přípony v názvu souboru „Nový dokument.docx“ je „.docx“. Rozdělit tak lze soubory na několik typů, mezi ty nejpoužívanější lze zařadit kancelářské, obrazové, zvukové a video soubory. [13] Nicméně vypsat všechny přípony, které by zde mohly být vypsány je nemožné, proto je možné na internetu najít databáze, seznamy, které je shromažďují na jednom místě.

1.3.1 Kancelářské soubory

Mezi nejznámější soubory lze zařadit textové soubory vytvořené pomocí aplikací z balíku Microsoft Office (MS Word, MS Excel a MS PowerPoint). Nejpoužívanější přípony takovýchto souborů jsou .docx, .xlsx, .pptx. Avšak do kancelářských souborů lze zařadit i soubory s příponami .txt, .pdf (Poratable Document Format File). Takovéto soubory se používají v případech, kdy je cílem práce s textem (úpravy, ukládání jednoduchého textu, sdílení apod.) bez ohledu na celkový vzhled dokumentu (formátování, tučné písmo apod.).

Soubory s příponou .docx

Tato přípona patří souborům vytvořeným softwarem MS Word, či případně jiným textovým editorem jako je například LibreOffice. Uvnitř může obsahovat formátovaný text, obrázky, grafy atd. Díky tomu je možné tvořit různé dokumenty jako jsou dopisy, články, smlouvy a mnoho dalších.

DOCX byl představen s názvem „Office Open XML“, neboť byl založen na XML, což způsobilo, že byl daleko efektivnější v ohledu zabírání místa na disku než jeho předchůdce doc. [14] Ten pro ukládání informací využíval jediný binární soubor, kdežto docx informace ukládá do více menších souborů, které jsou následně komprimovány.

Soubory s příponou .pdf

PDF aneb Portable Document Format lze přeložit jako přenosný formát dokumentu. Formát byl vytvořen společností Adobe. Obsahem takových souborů je obdobně jako v případě DOCX text, obrázky apod. Avšak výhodou takovýchto souborů oproti zmíněnému DOCX je jeho stálá podoba. Na každém zařízení v každém programu (který formát PDF dokáže přečíst) bude soubor vždy vypadat totožně. Díky tomu se jedná možná o nejvíce rozšířený formát při výměně souboru obsahující text. Jeho rozšiřitelnosti taky napomáhá fakt, že je to volně přístupný formát. [15]

Formát PDF má různé standardy dle normy ISO (International Organization for Standardization) 32000-2:2020, kde jsou definovány například standardy pro dlouhodobou archivaci – PDF/A, standard pro tiskařský průmysl – PDF/X, standard pro univerzální přístup – PDF/UA. Standard pro archivaci PDF/A byl zpočátku jen náhrada naskenovaných papírů a TIFF (Tagged Image File Format) formátu. Nyní se však běžně používá pro archivaci souborů. Standardy mají i své různé verze, např. PDF/A-4e nahrazuje standard PDF/E ze starší verze normy ISO 32000, který určuje, jak archivovat PDF soubory s trojrozměrnými výkresy. Standard PDF/UA definuje, jak musí být daný soubor navržen, aby jej mohli využívat lidé s postižením a stroje. [16]

1.3.2 Obrazové soubory

Obrazové soubory neboli také grafické soubory. Slouží k vyjádření grafických dat jako jsou fotografie, obrázky apod. Grafické soubory lze rozdělit dle typu grafiky na vektorové a rastrové (bitmapové) formáty a dle jejich komprese na komprimované (ztrátové a bezeztrátové) a nekomprimované. [17] Pro prohlížení grafických souborů lze použít výchozího programu ve Windows – Fotky, který lze i využít pro základní úkony s obrázkem (oříznout, otočit apod.). Pro složitější práci s obrázky a fotografiemi je často využíván Adobe Photoshop.

Vektorové formáty

Tato metoda využívá pro ukládání dat geometrických popisů pomocí matematických vzorců (obrázek je složen z geometrických útvarů – přímky, body, křivky, n-úhelníky). To pomáhá ke skutečnosti, že není při úpravě dat ztracena kvalita. Vektorových formátů lze využít při tvorbě animací, log atd.

Rastrové (bitmapové) formáty

Na rozdíl od vektorových formátů jsou zde data reprezentována pomocí pixelů. Pixely jsou uspořádány a mají své vlastnosti, díky kterým vytvoří obraz. Jejich vlastnosti jsou souřadnice, kde v jsou obrazu rozmístěny a jejich barva. Rastrové formáty lze rozdělit na základě jejich komprese, s bezeztrátovou kompresí a se ztrátovou kompresí. Během bezeztrátové komprese dochází ke ztrátě některých nadbytečných informací, ale nedochází ke ztrátě kvality. Naopak u ztrátové komprese dochází ke ztrátě kvality a ztrátě nadbytečných informací, to ovšem za cenu velkého snížení velikosti souboru. Mezi nejznámější ztrátové formáty patří JPEG (Joint Photographic Experts Group) – avšak správný název je JFIF. U bezeztrátových formátu se poté bude jednat o formát PNG (Portable Network Graphics).

1.3.3 Zvukové soubory

Obsahem audio souborů je zvuk. Zvuk je fyzikální jev, jež je produkovaný chvěním předmětu ve formě analogového signálu. K reprezentaci analogového signálu na počítači je nutné ho pomocí analogově digitálního převodníku převést do digitální podoby. [18] Audio soubory běžně slouží k ukládání zvukových stop jako například hudba, různé rozhovory, či zvuková část videa. Pro přehrávání audio souborů je možné využít programů pouze pro audio eventuelně některé přehrávače pro video soubory.

Stejně jako grafické soubory, tak audio soubory lze rozdělit dle jejich komprese. V bezeztrátové kompresi dochází k rozpadu dat na menší částí, které jsou následně zpětně sjednoceny. Kdežto ve ztrátové kompresi jsou bity informací eliminovány po skončení procesu komprese. [18] Známým zástupcem ztrátových audio souborů je formát MP3 (známý také jako MPEG-2 – Moving Picture Experts Group, využívající kompresi zvuku „Layer 3“). Dalším představitelem zvukového souboru, avšak bezeztrátového, je formát FLAC (Free Lossless Audio Codec). Jeho nesmírnou je fakt, že tento formát je volně šiřitelný. Naopak jeho nevýhodu, jako u všech typů s bezeztrátovou kompresí, je velká celková velikost souboru.

1.3.4 Video soubory

Video soubory reprezentují, obdobně jako grafické soubory obraz. Avšak s rozdílem, že se jedná o pohyblivý obraz, kde na sebe jednotlivé obrázky navazují. Často jsou videa doprovázena zvukem. Jsou tvořena dvěma částmi. Kodekem, což je protokol pro kódování a dekódování videa. A kontejnerem, který obsahuje data a metadata video

souboru. O jaký kontejner se jedná prozrazuje přípona souboru, rozšířené jsou kontejnery AVI (Audio Video Interleave), MP4 (známý jako MPEG-4 Part 14) nebo MKV (Matroska Video File). Kontejner může mít různé typy kodeků a je možné do některých z nich vložit další data jako například titulky. Běžnými příklady kodeků jsou HEVC (High Efficiency Video Coding, známý také jako H.265 nebo MPEH-4 part 2), H.264 (MPEG-4 part 10), MPEG-4. [19]

1.3.5 Ostatní soubory

Jako ostatní druhy souborů můžeme uvést například archivní soubory, spustitelné soubory a jiné.

Cílem archivního souboru je sjednotit několik souborů, což napomáhá lepší distribuci souborů. Při sjednocování je možné použít kompresi dat, tak aby výsledný soubor měl menší velikost. Mezi oblíbené patří komprimované složky s příponou .rar (formát RAR – Roshal Archive), výhodou tohoto formátu je vyšší kompresní poměr než komprese formátu ZIP (Zipped File). [20] Další známým archivním formátem je formát 7z (7-Zip Compressed File).

1.4 Autorskoprávní ochrana elektronických dokumentů

Tak jako díla fyzická, i díla digitální musí být právně chráněná, aby nedošlo k jejich následnému zneužití a poškození jak autora nebo držitele práv k danému autorskému dílu. Dle §2 zákona č. 121/2000 Sb., o právu autorském, právech souvisejících s právem autorským a o změně některých zákonů (dále jen autorský zákon) může být autorské dílo vyjádřeno v elektronické podobě, a tudíž tak chráněno dle autorského zákona.

V případě přebírání pojmu elektronický dokument ze zahraničních definic se překlad často potýká s křížovým překladem. Dvojice pojmů záznam (*record*) a dokument (*document*) jsou navzájem prohozeny tak, že se *record* překládá jako dokument a *document* jako záznam. Tento překlad je však špatný z důvodu rozdílu mezi záznamem a dokumentem. Záznam na rozdíl od dokumentu je verzovatelný a není fixován v čase. [3]

V souvislosti s ochranou elektronických dokumentů lze narazit na termín správa digitálních práv (z anglického Digital Rights Management – DRM). DRM lze definovat jako „*Technologie, která kontroluje a omezuje používání digitálních děl (elektronické hudby, filmů, elektronických knih, počítačových her apod.) s cílem zabránit nelegálnímu užití těchto děl, např. zamezuje uživatelům kopírování digitálního díla, stanovuje určité časové období, po které může být toto dílo používáno, omezuje počet zařízení, na kterých může být dílo užíváno, zakazuje konverzi díla do jiných formátů apod. Technologii DRM používají poskytovatelé digitálních děl jako např. hudební vydavatelé, filmoví producenti, knižní nakladatelé, autoři apod.*“ [21]

1.4.1 Creative Commons

Creative Commons (zkráceně CC) je nezisková organizace, která napomáhá k snadnému a právně ošetřenému šíření děl na internetu. K tomu využívá licencí CC. Ty zajišťují komukoliv, kdo jimi své dílo ošetří, způsob jak je možné svou práci publikovat veřejnosti bez toho aniž by dílo nebylo chráněno autorským právem. Z pohledu veřejnosti, je-li dílo publikováno s licencí CC, je možné zjistit jaké práva a povinnosti má jedinec, který by dílo dále publikoval. [22] Mezi díla, která se nejčastěji využívají s licencemi CC patří obrázky, videa apod.

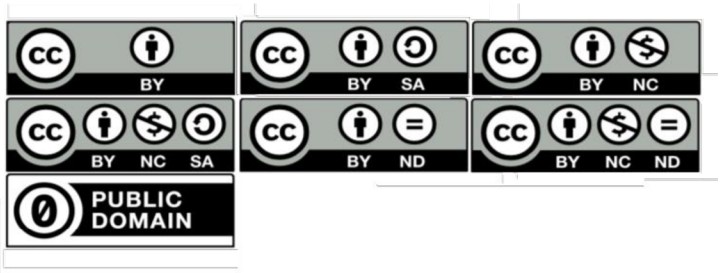
Rozlišujeme sedm možných licencí CC (viz obrázek 1.3). Každý prvek v dané licenci má svoji definici. Pouze licence s prvkem „0“ namísto „CC“ značí, že se je jedná o licenci „CC0“, u které se autor vzdal veškerých práv na dílo a je tak možné dílo volně užívat. Níže jsou definovány jednotlivé prvky licencí.

Prvek BY – Uvést autora díla (z angl. Attribution)

Prvek SA – Zachovat licenci, pod kterou je dílo publikováno (z angl. Share Alike)

Prvek NC – Neužívat dílo komerčně (z angl. Noncommercial)

Prvek ND – Nezpracovávat, dílo nesmí být pozměněno (z angl. No derivates)



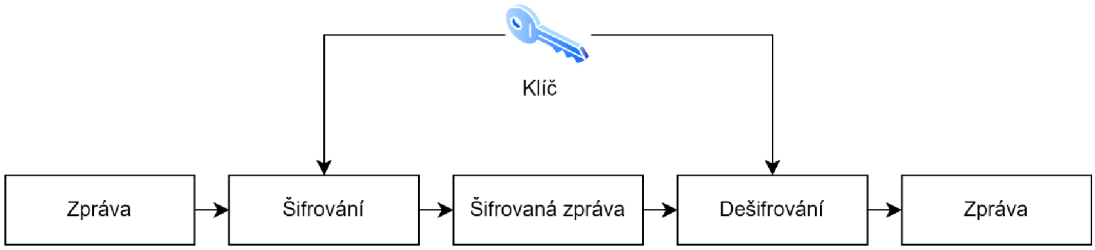
Obrázek 1.3: Licence Creative Commons.

1.5 Ochrana elektronického dokumentu

Elektronické dokumenty je nutné chránit před změnou jejich obsahu, potažmo jejich zneužití. V případě, kdy by tato skutečnost nebylo možné zajistit, jednalo by se o velký problém (např. v moment, kdy společnost pro svou internetovou reklamu použije fotografii bez svolení autora, tzn. že fotografii jednoduše stáhne a použije). K ochraně proti zneužití existuje několik možností, jak zneužití předejít a některé z nich zde budou více rozebrány. Hlavním nástrojem, který je používán k předcházení, je využití kryptografie. Jedním způsobem, jak zabezpečit data je pomocí šifrování, což je proces, během kterého se z nezabezpečených dat stávají data zabezpečená. Šifrování lze rozdělit na symetrické a asymetrické.

Symetrické šifrování

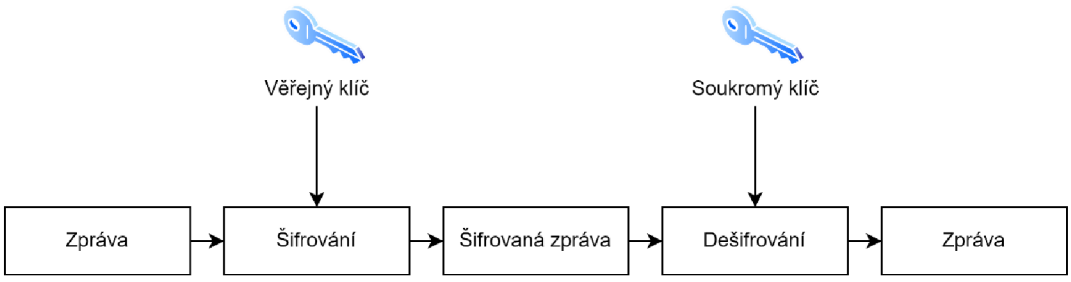
Tento druh šifrování využívá pro šifrování i dešifrování dat proudovou nebo blokovou šifru. Proudová šifra převádí otevřený text na šifrovaný text po jednom bitu či bajtu a bloková šifra převádí celé jednotky nebo bloky otevřeného textu pomocí předem určené délky klíče, jako je 128, 192 nebo 256 bitů. [23] Jak při šifrování, tak i při dešifrování se využívá stejného klíče (viz obrázek 1.4). Výhoda symetrické kryptografie tkví v rychlosti provádění kryptografických operací (šifrování a dešifrování). Pakliže lze předat klíč druhé straně bezpečnou cestou, jedná se o bezpečný typ šifrování (při použití správných šifrovacích algoritmů). Používanými algoritmy jsou například AES (Advanced Encryption Standard), 3DES (Triple Data Encryption Standard).



Obrázek 1.4: Symetrické šifrování.

Asymetrické šifrování

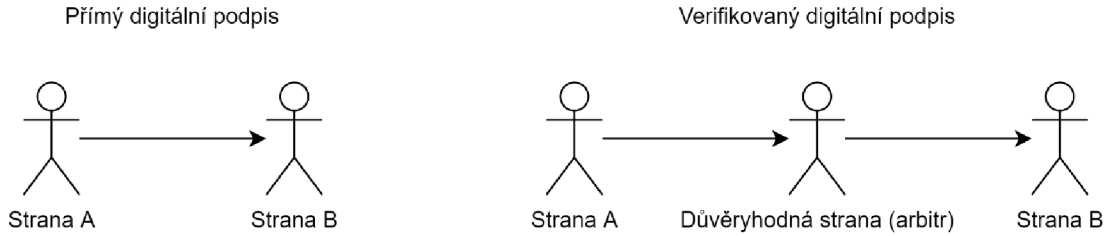
Oproti symetrickému šifrování, asymetrické využívá dva různé klíče, pro každou kryptografickou operaci jiný (viz obrázek 1.5). V situaci, kdy chceme zašifrovat zprávu asymetrickým šifrování, vezmeme veřejný klíč cílového uživatele a zašifrujeme zprávu, poté jí odešleme. Následně cílový uživatel pomocí svého soukromého (privátního) klíče zprávu dešifruje. Asymetrické šifrování je oproti symetrickému výpočetně složitější, a proto je pomalejší. Používanými algoritmy jsou např. RSA (Rivest Shamir Adleman), ECC (Elliptic Curve Cryptography) a DSA (Digital Signature Algorithm).



Obrázek 1.5: Asymetrické šifrování.

1.5.1 Digitální podpis

Pod pojmem digitální podpis se obecně chápe podpis vytvořený kryptografickými prostředky, jehož úkolem je prokázání pravosti digitální zprávy z důvěryhodného zdroje. [24] Systém digitálního podpisu pracuje na základě asymetrické kryptografie. Rozlišujeme dva typy digitálního podpisu, přímý a verifikovaný. [24] Přímý digitální podpis obsahuje pouze dvě strany (viz obrázek 1.6). Naopak u verifikovaného digitálního podpisu máme tři strany (viz obrázek 1.6), kde třetí, důvěryhodná, strana (arbitr) zajišťuje důvěrnost posílané zprávy. [25]



Obrázek 1.6: Typy digitálního podpisu.

S pojmem digitální podpis se často zaměňuje elektronický podpis. Elektronický podpis by měl splňovat vlastnosti vlastnoručně psaného podpisu. Takovými vlastnostmi ručního podpisu jsou nepopiratelnost, podepsaný dokument již nelze měnit, podpis je nepřenosný, nefalšovatelný a jednoznačně přiřazen osobě. Dle nařízení eIDAS rozlišujeme tři druhy elektronického podpisu (neboli eSignature) [26]:

- Jednoduchý elektronický podpis
- Zaručený elektronický podpis
- Kvalifikovaný elektronický podpis

Definici jednotlivých podpisů najdeme ve článku 3 nařízení eIDAS.

Jednoduchý elektronický podpis

Definice jednoduchého elektronického podpisu zní: *„data v elektronické podobě, která jsou připojena k jiným datům v elektronické podobě nebo jsou s nimi logicky spojena a která podepisující osoba používá k podepsání.“* [27]

Zaručený elektronický podpis

Podpis tohoto typu musí splňovat, dle nařízení eIDAS článku 26, identifikaci podepisující osoby, jednoznačné spojení s podepisující osobou. Dále je vytvořen pomocí dat pro vytváření elektronických dokumentů a k datům, která jsou podpisem podepsána je připojen tak, že jakákoliv změna dat je zjistitelná. [28]

Kvalifikovaný elektronický podpis

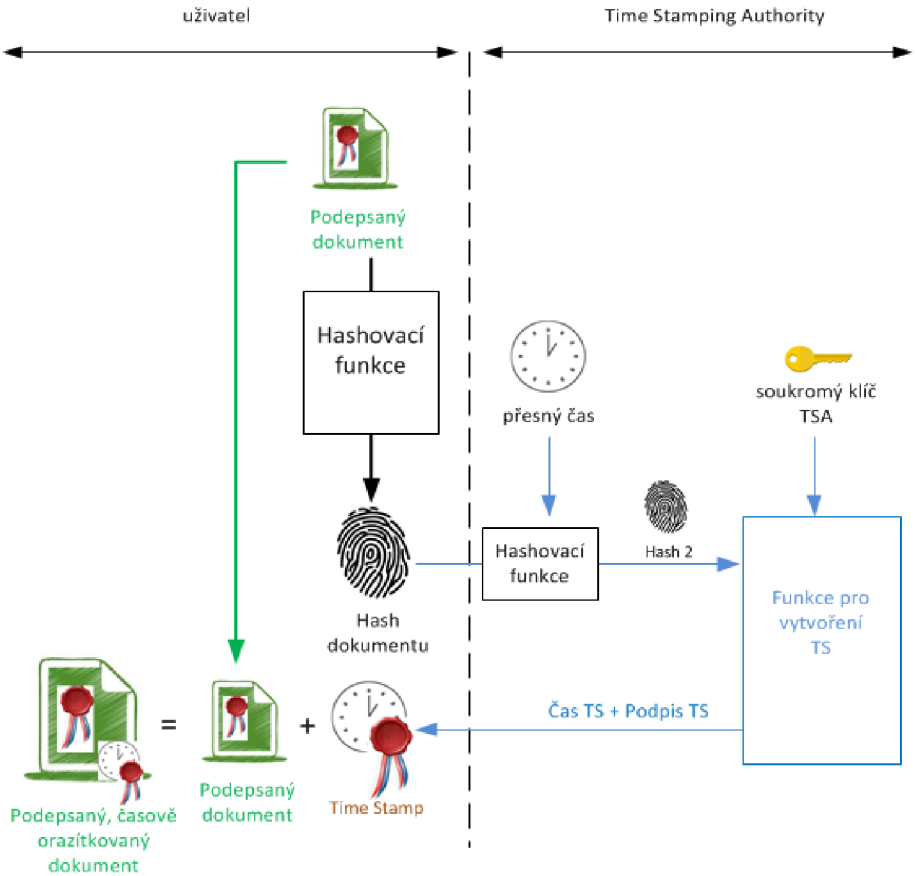
Za zaručený elektronický podpis považujeme takový podpis, který je vytvořen kvalifikovaným prostředkem pro vytváření elektronických podpisů a který je zároveň založen na kvalifikovaném certifikátu pro elektronické podpisy. [29]

1.5.2 Elektronická pečeť

Elektronická pečeť je speciální typ elektronického podpisu, který nemá úzkou vazbu na podepisující fyzickou osobu. Elektronická pečeť reprezentuje právní osobu a slouží k prokázání původu a integrity dat.

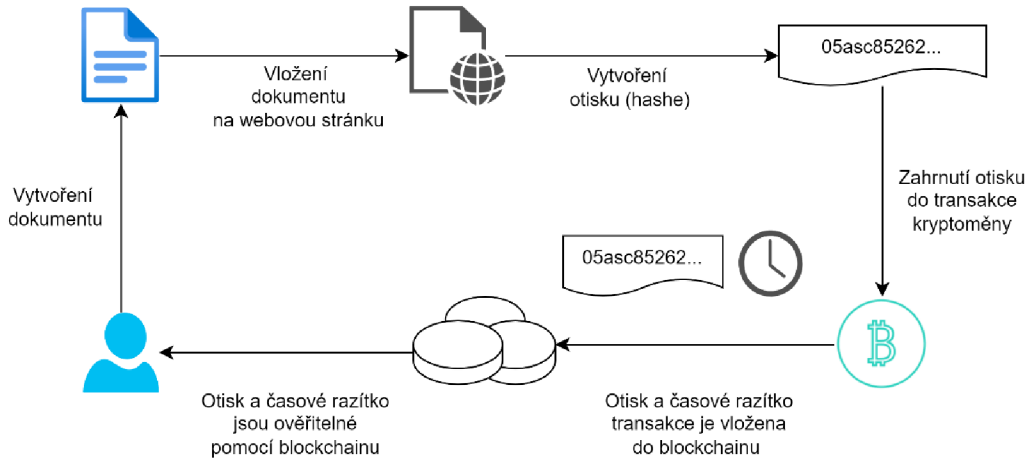
1.5.3 Elektronické časové razítko

Elektronické časové razítko slouží k prokázání skutečnosti, že v daný časový moment dokument existoval a od té doby se nezměnil. Dokumentem určeným k otisku může být například cokoli počínaje skenováním PDF, textovým souborem, digitální fotografií, videem apod. V celém procesu vytváření elektronického časového razítka vystupují dvě strany, na jedné straně uživatel a na druhé tzv. TSA (Time Stamping Authority) – autorita časových razítek. Ta zaručuje vytvoření časového razítka. Proces vytvoření elektronického časového razítka je vyobrazen níže (viz obrázek 1.7).



Obrázek 1.7: Vytvoření elektronického časového razítka [30].

Existuje typ časových razítek, která jsou spojená s kryptoměnami (přesněji řečeno s blockchain technologií). Popis funkcionality časového razítka s využitím blockchainu je uveden níže na obrázku 1.8. Technologie blockchain může být použita k záznamu, sledování a ověřování časů vytváření a úprav dokumentů. Uživatelé mohou nahradit moderní notářské služby snadno použitelnou platformou pro časové razítko, aby zajistili vlastnictví jejich díla. A to vše lze veřejně a bezpečně zpracovávat prostřednictvím digitální platformy bez pomoci prostředníka. Držitelé dokumentu tak mohou nabídnout doklad o vlastnictví v určitý čas a datum, aniž by byl odhalilen obsah dokumentu. [31] Poskytovatel takových služeb je například Projekt Certoo, OriginStamp či OpenTimestamps. První uvedený, Projekt Certoo, je český projekt, který využívá typ blockchainu (kryptoměny) Litecoin. [32] Firma OriginStamp využívá Bitcoinovou síť. OriginStamp vytvoří jedinou transakci pro všechny otisky odeslané během 24 hodin. To je možné s pomocí Merkle Tree, který umožňuje efektivní kombinování hashů, tím se celý proces zefektňuje a snižují se navíc náklady. [31] Poslední uvedený, OpenTimestamps, je open-source projekt taktéž využívající sítě Bitcoinu. [33]



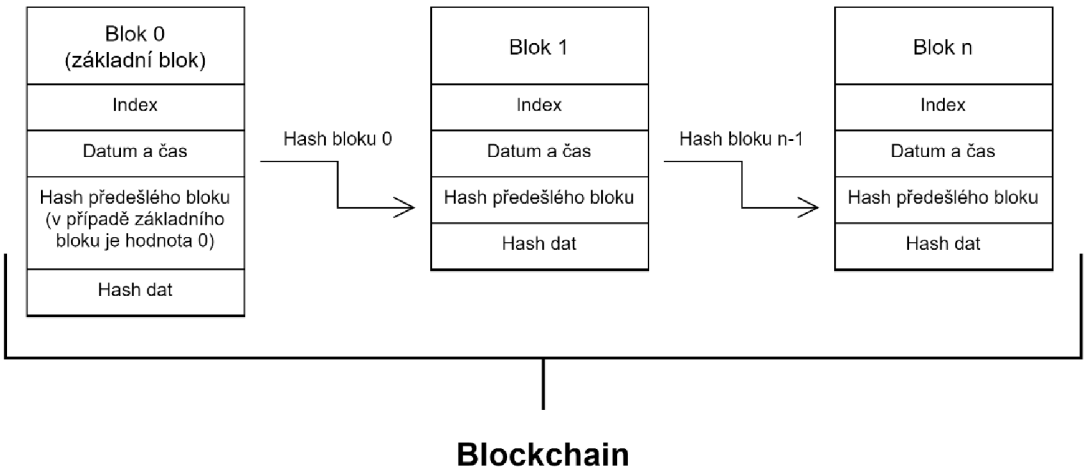
Obrázek 1.8: Časové razítko využívající kryptoměny [34].

1.5.4 Technologie blockchain

Jak již z názvu vyplívá, blockchain je řetězec bloků, kde každý blok navazuje na další blok a jsou tak celkově všechny bloky provázané. V dnešní době se blockchain více a více rozšiřuje, a to nejen díky kryptoměnám, ve kterých tato technologie našla uplatnění a je jejím stavebním kamenem. Síla blockchainu tkví v decentralizaci (řetězec bloků není umístěn jen na jednom místě, které by bylo možné zničit), průhlednosti (celý řetězec je dostupný) a je plně automatizovaný (nemá žádného správce).

Každý blok obsahuje hash souboru (digitální otisk), hash předchozího bloku a další data navíc. Tvorba blockchainu je znázorněna na obrázku 1.9. Termínem, se kterým se lze setkat u blockchainu je těžení (často spojované právě s kryptoměnami). Těžení je

operace, u které se těžař snaží „uhodnout“ hodnotu „nonce“, což je náhodné číslo, které se přidává k datům v bloku s cílem utvořit požadovanou formu otisku bloku. Pokud však dojde ke změně dat v bloku, projeví se to změnou hashe daného bloku, a tudíž i změnou hashe všech následujících bloků. Pokud tedy dojde ke skryté či sebemenší změně dat v kterémkoli z bloků, část řetězce bude neplatná, a řetězec nebude validní.

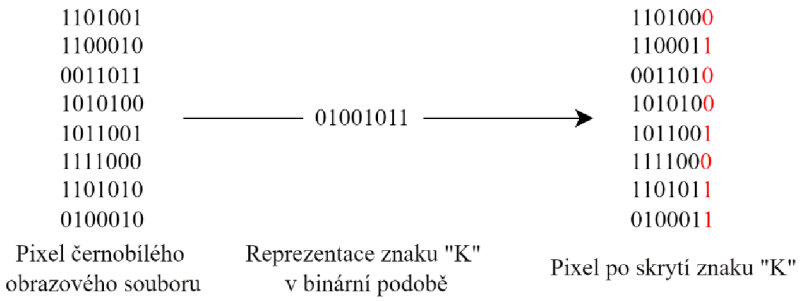


Obrázek 1.9: Tvorba řetězce bloků (blockchainu).

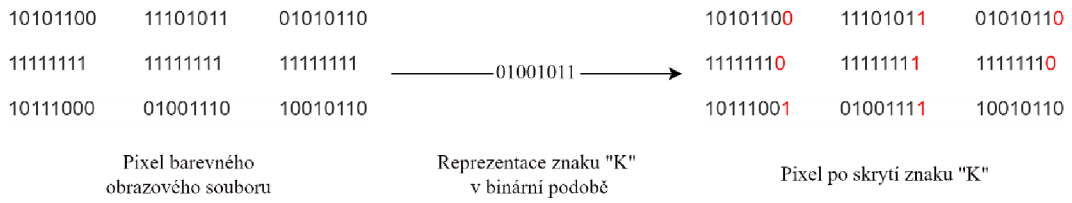
1.5.5 Steganografie

Steganografie funguje tak, že skrývá informace způsobem, který nevzbuzuje jakékoliv podezření. Pomocí steganografie lze ukrýt zprávu do textových, audio a obrazových souborů, a i například do webových stránek.

Existuje mnoho metod steganografie (vstřikování, substituce a generování nových souborů [35]), avšak v této práci bude rozebírána pouze technika metody substituce, a to steganografie s nejméně významným bitem (LSB – Least Significant Bit). Touto technikou se vkládají data do nejméně významných bitů mediálního souboru. Každý pixel v barevném obrazovém souboru je složen ze tří bajtů dat odpovídajících barvám červené, zelené a modré. Pakliže je obrazový soubor černobílý (ve stupních šedi), každý pixel je reprezentován 8 bity. Ukrývání zprávy je prováděno tak, že se zpráva převede z ASCII hodnoty na binární a následně je vložena na poslední bit (1 bit u černobílého obrazového souboru a až 3 bity u barevného). Operace vkládání znaku je znázorněna níže na obrázcích 1.10 a 1.11.



Obrázek 1.10: Vložení písmene „K“ do pixelu černobílého obrazového souboru (vytvořen na základě [36]).



Obrázek 1.11: Vložení písmene „K“ do pixelu barevného obrazového souboru (vytvořen na základě [37]).

1.5.6 Digitální vodoznak

Digitální vodoznak je jedním z možných způsobů ochrany dokumentů. Vodoznakem může být znak loga organizace nebo jednotlivce, který vlastní práva k digitálnímu obsahu a obvykle obsahuje informace spojené s autorskými právy, vlastnictvím, vydavatelem a informace o dokumentu. [38] Existuje několik možných druhů rozdělení digitálních vodoznaků, jedním takovým rozdělením je rozdělením do tří typů (dle [39]):

- Viditelný digitální vodoznak
- Neviditelný robustní vodoznak
- Neviditelný křehký vodoznak

Viditelný vodoznak na rozdíl od zbylých typů je viditelný pro každého, jedná se o překryv dvou obrazových souborů. Neviditelný robustní vodoznak je vložen takovým způsobem, že změny provedené v hodnotě pixelu nejsou zaznamenány a lze jej obnovit pouze s vhodným dekódovacím mechanismem. Neviditelný křehký vodoznak je zakomponován v souboru tak, že jakákoli manipulace nebo úprava obrázku by změnila nebo zničila vodoznak. [39]

V rámci bakalářské práce bude zpracován kodér pro vložení viditelného digitálního vodoznaku, který bude vkládat vodoznak do obrazového souboru.

2 PRAKTICKÁ ČÁST

Cílem v teoretické části práce bylo definovat elektronický dokument a dále zanalyzovat možnosti ochran autorských práv takového dokumentu. Na základě této analýzy navrhnout koncept, a i samotné řešení dané aplikace.

V této kapitole bude rozebráno řešení této aplikace pro ochranu autorských práv elektronických dokumentů a následné zhodnocení dosažené ochrany. V rámci zhodnocení ochrany bude uvedeno i možné vylepšení.

Konkrétní aplikace se zaměřuje na ochranu autorských práv pomocí vložení vodoznaku do obrazového souboru s příponou .jpg, ukrytí zprávy pomocí techniky steganografie do souboru formátu JPG, digitálního podpisu jednak za použití self-signed certifikátu a také za použití certifikátu uloženého v kontejneru formátu PFX. Nadále jsou v práci použita časová razítka a je zde nastíněna technologie blockchain s použitím v oblasti ochrany autorských práv. Funkcionalita aplikace je vytvořena pomocí programovacího jazyka Python, s využitím kryptografické knihovny OpenSSL v případech digitálního podpisu a časových razítek. Grafické uživatelské rozhraní je naprogramováno pomocí frameworku PyQt5.

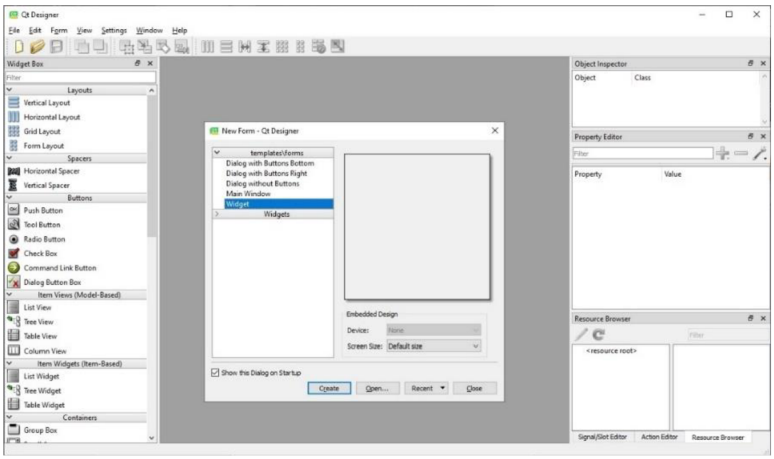
2.1 Vytvoření aplikace

Pro vytvoření aplikace je vybrán programovací jazyk python ve verzi 3.10.3. Jako vývojové prostředí pro programování (tzv. IDE – Integrated Development Enviroment) bylo zvoleno PyCharm 2021.2.2 (Community Edition).

V rámci celé aplikace je využito hned několika knihoven. K vytvoření grafického rozhraní byl využit framework PyQt5. Pro práci s obrazovými soubory je využita knihovna OpenCV (Open Source Computer Vision Library). Dále jsou použity moduly například pro vložení přesného času, modul pro práci s cestami k souborům apod.

2.1.1 Vytvoření GUI

K vytvoření grafického rozhraní byl využit program Qt designer (viz obrázek 2.1), což je nástroj z frameworku Qt pro jazyk Python. Tento program značně zjednodušuje vytváření grafického rozhraní pro Python. Původně je framework napsán v programovacím jazyce C++. Výhodou je, že díky Qt desineru je možné vytvořit rozhraní v podobě, které je žádané, a to i díky schopnosti WYSIWYG (z angl. What You See Is What You Get), která ve volném překladu znamená, že software zobrazuje skutečnou podobu, kterou bude finální verze mít. [40] Qt není ovšem jen pro vytváření GUI, zahrnuje i například SQL databáze, funkční webový prohlížeč, OpenGL, a jiné. [41]

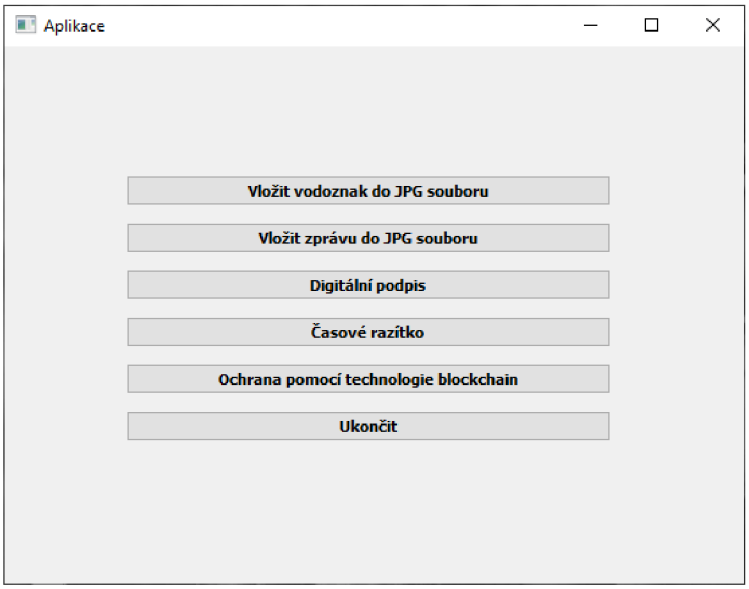


Obrázek 2.1: Ukázka aplikace Qt designer.

K vytvoření grafického rozhraní aplikace by mohlo být využito i dalších knihoven. Například balíček Tkinter. Tento balíček má velkou výhodu, že je na rozdíl od ostatních implementován v samotném Pythonu a dá se tedy téměř s jistotou říct, že je dostupný všude tam, kde je Python nainstalován. Další možné balíčky, frameworky, nástroje pro tvorbu GUI jsou Kivy, wxPython, Libavg atd.

2.2 Popis grafického rozhraní aplikace

Celá aplikace je vytvořena v jednoduché a lehce přehledné formě. Při každém kroku je otevřeno nové okno a předchozí zavřeno. Při prvotním spuštění je zobrazena úvodní obrazovka s rozcestníkem. Zde je na výběr ihned z několika možností (viz obrázek 2.2), jak lze chránit digitální dokumenty.

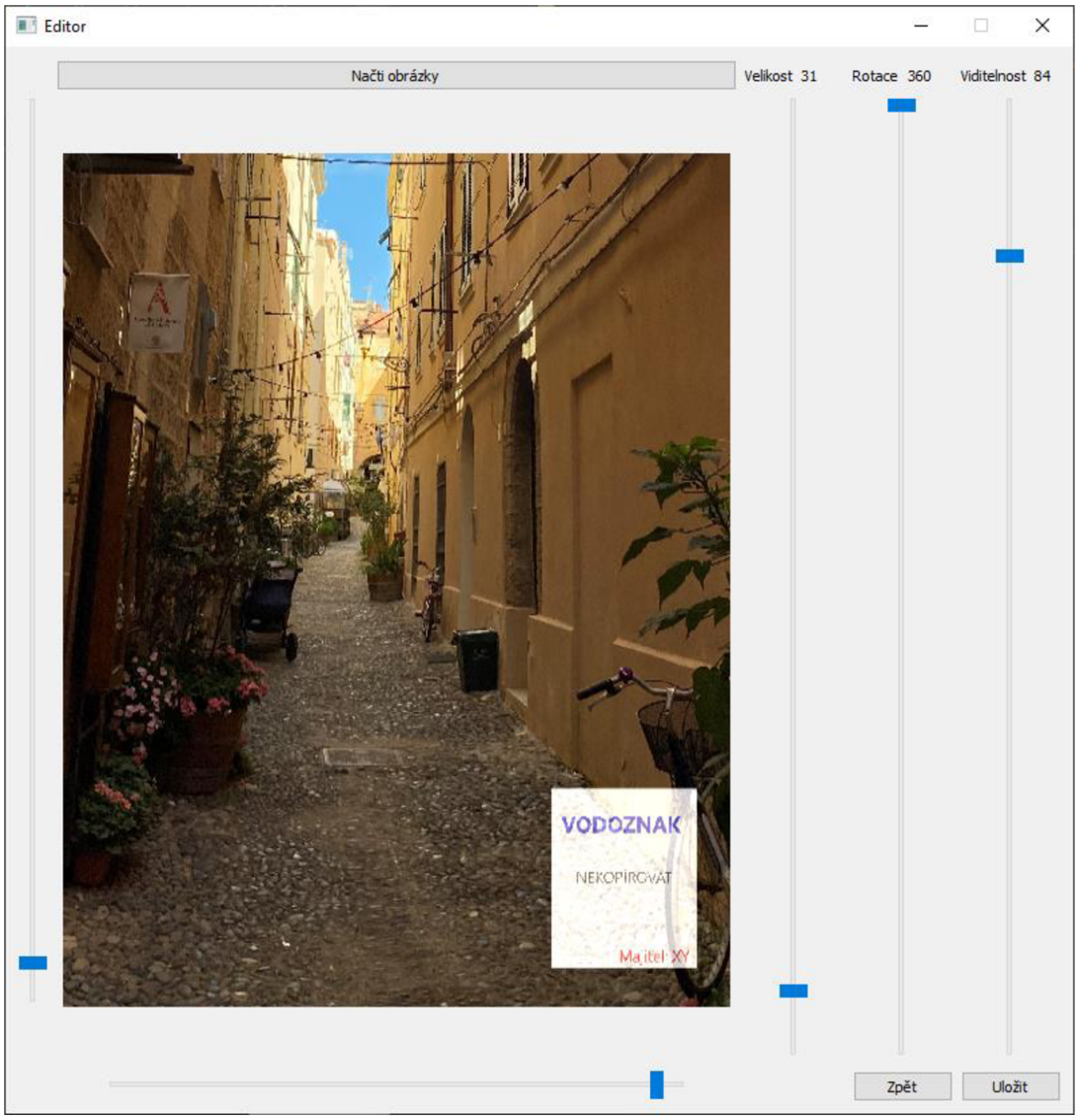


Obrázek 2.2: Zobrazení úvodního okna aplikace.

První volbou z menu je otevřeno okno, ve kterém je možné vkládat připravený vodoznak ve formátu JPG do cílového souboru s příponou .jpg. Výsledný obrázek s vodoznakem může vypadat následovně (viz obrázek 2.3).

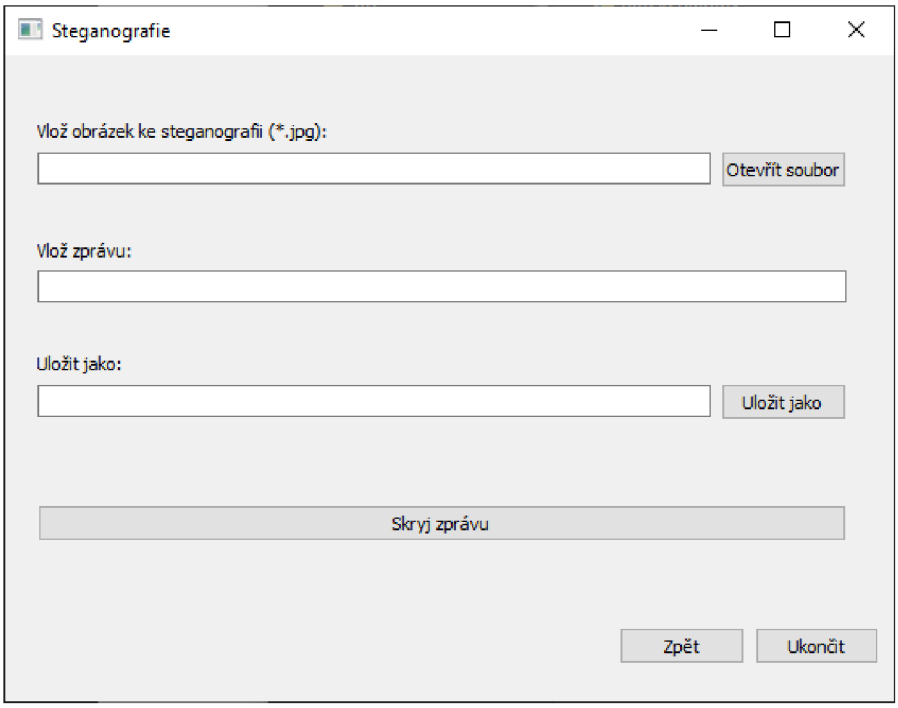
Obrázek 2.3: Ukázka obrázku s vodoznakem.

Připravené řádky slouží k vložení cest k daným souborům. Při stisknutí tlačítka „Otevřít soubor“ se otevře nové okno s adresáři, ve kterém je možné daný soubor najít a potvrzením stisknutím tlačítka „Otevřít“ se příslušná cesta k souboru vepíše do připravených řádků. V neposlední řadě je možné si vybrat kam a pod jakým názvem bude výsledný soubor uložen. K tomu poslouží poslední ze tří řádků a tlačítko „Uložit jako“, které funguje obdobně jako tlačítko „Otevřít soubor“. Další akcí je otevření editoru (viz obrázek 2.4), pomocí kterého je možné vložit požadovaný vodoznak na přesněji určené místo v cílovém souboru. Editor také dovoluje pozměnit v určité míře velikost, rotaci a průhlednost vodoznaku.



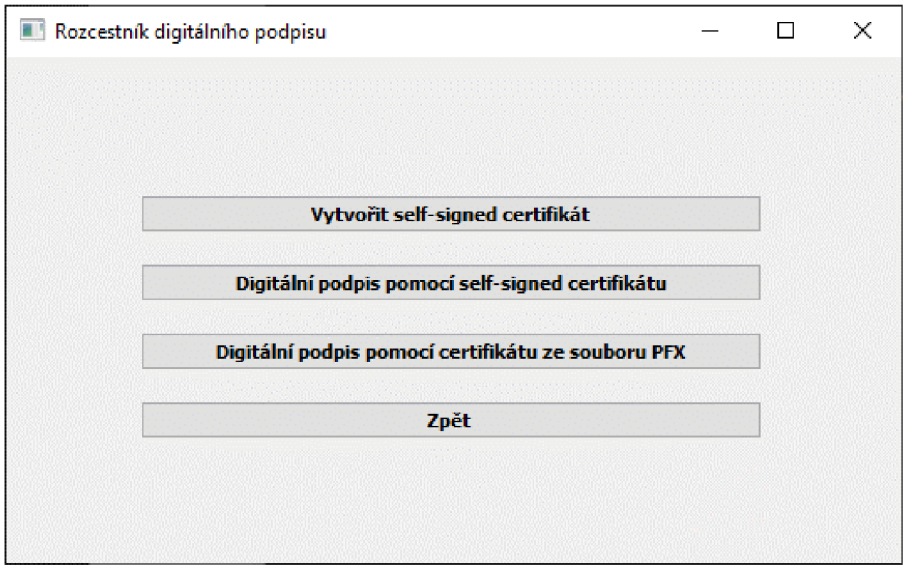
Obrázek 2.4: Editor vložení vodoznaku.

Další možností rozcestníku je vložení skryté zprávy do obrazového souboru. Tato možnost obsahuje funkci pro vložení zprávy do obrázku, aniž by na první pohled byla zřejmá při použití jednoduchého softwaru pro zobrazení souboru obrazového typu. Vzhled je opět téměř identický s předchozí volbou v menu. Hlavní část zabírají tři řádky (viz obrázek 2.5, jeden pro vložení cesty k souboru, do kterého bude text zprávy skrýván, druhý pro vložení textu zprávy a poslední pro volbu cesty, kam bude požadovaný nový soubor uložen. I spodní část okna je identická a nachází se tu trojice tlačítek pro provedení vložení skryté zprávy do souboru, možnosti vrátit se zpět do menu či ukončit aplikaci.



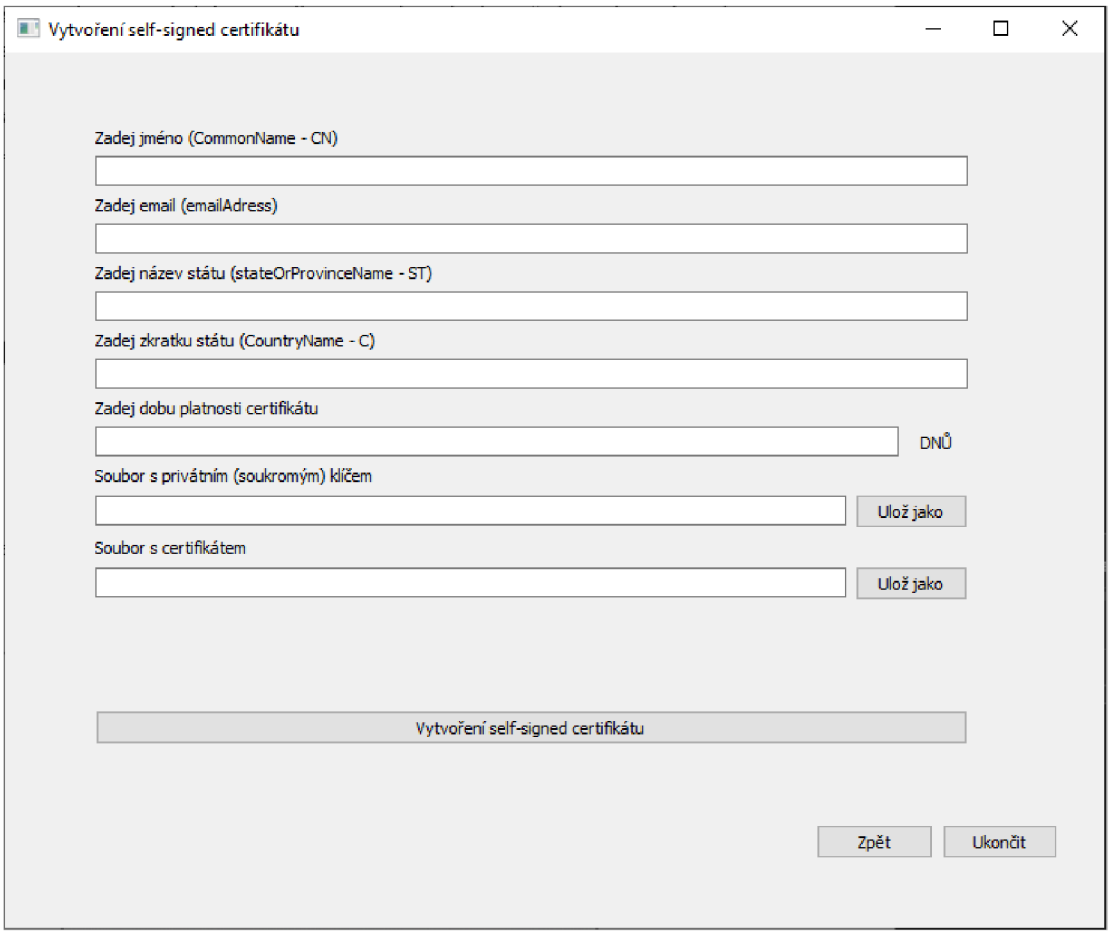
Obrázek 2.5: Zobrazení volby aplikace pro využití steganografie.

Třetí variantou rozcestníku aplikace je digitální podpis. Tato volba otevře nový rozcestník (viz obrázek 2.6: Rozcestník digitálního podpisu.), který nabízí cestu zpět do úvodní části aplikace a tři možné následující kroky – „Vytvoření self-signed certifikátu“, „Digitální podpis pomocí self-signed certifikátu“ a „Digitální podpis pomocí certifikátu ze souboru PFX“.



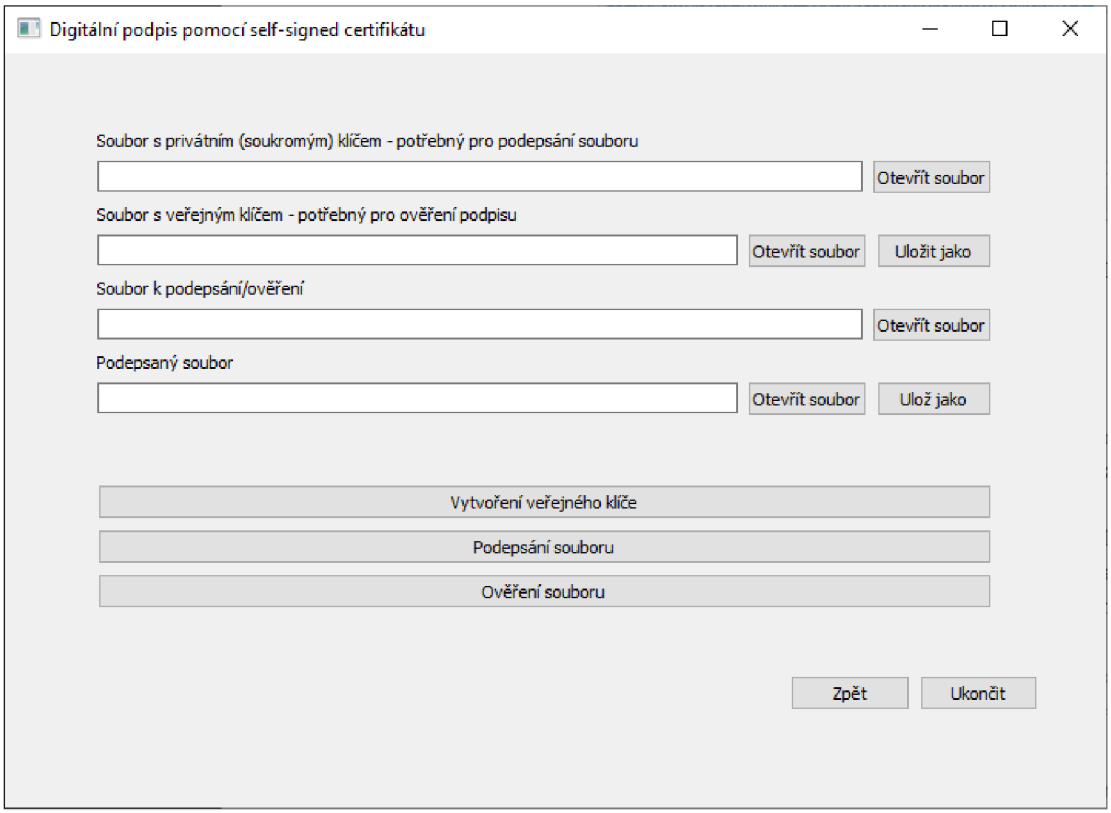
Obrázek 2.6: Rozcestník digitálního podpisu.

Jak název napovídá, stiskem tlačítka pro vytvoření certifikátu bude uživatel přenesen do okna pro vytvoření certifikátu za účelem vytvořit certifikát podepsaný sám sebou (viz obrázek 2.9).



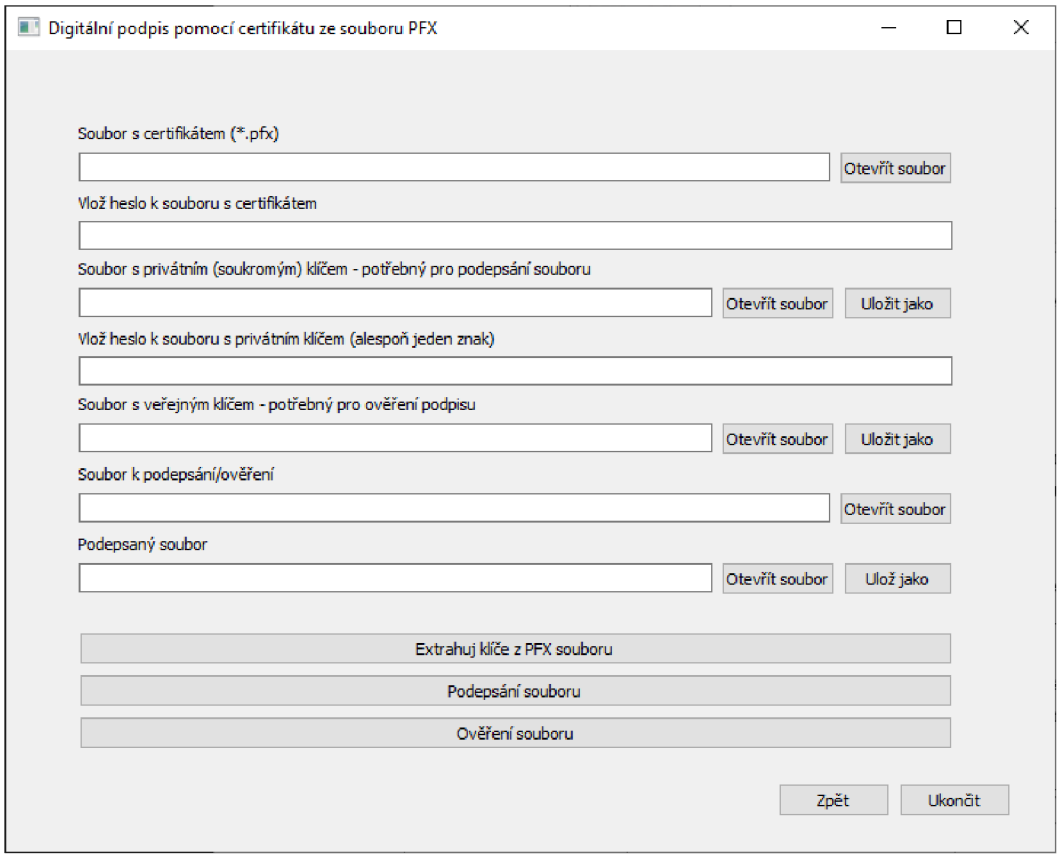
Obrázek 2.7: Okno pro vytvoření self-signed certifikátu.

Druhá alternativa je podepsání souboru právě pomocí vytvořeného self-signed certifikátu, přesněji řečeno pomocí vytvořeného privátního klíče. V této možnosti je nutné pro podepsání souboru vložit privátní klíč, soubor a podepsaný soubor uložit. Pro ověření podepsaného souboru je nutné vlastnit veřejný klíč, ten lze získat vytvořením z privátního klíče, a samotný podepsaný soubor. Okno této části aplikace je znázorněné na obrázku 2.8.



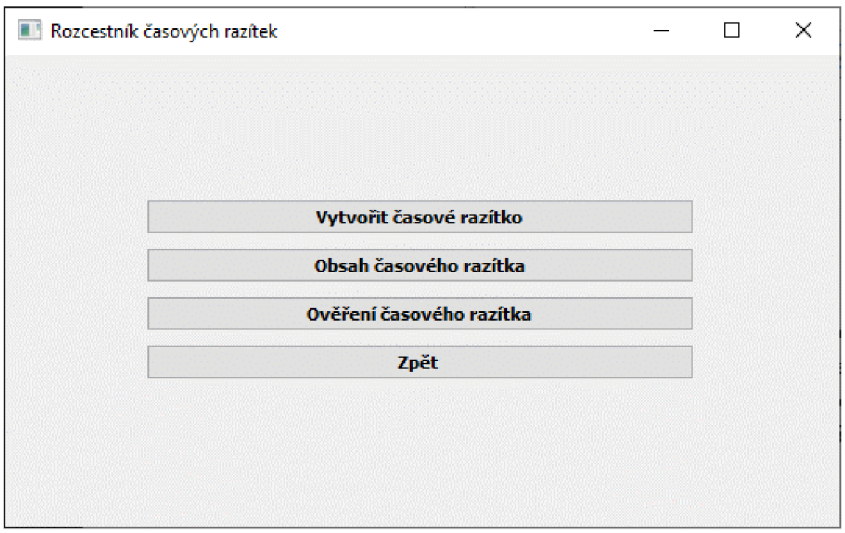
Obrázek 2.8: Okno digitálního podpisu za použití self-signed certifikátu.

Poslední volbou v rozcestníku digitálního podpisu je podpis s využitím certifikátu uloženého v souboru s příponou .pfx, znázorněné obrázkem 2.9. Z tohoto souboru – kontejneru lze extrahovat dvojici klíčů (privátní a veřejný), jenž je využito stejně jako v předchozím případě k podepsání/ověření souboru. Avšak jeden markantní rozdíl mezi variantami tkví ve využití hesel. Neboť právě soubor ve formátu PFX ukládá klíče v zašifrované podobě, přičemž je zašifrován i kontejner samotný. K zašifrování používá např. mechanismus vygenerování 40 bitového tajného klíče z hesla (vytvořený šifrou RC2 v módu Cipher Block Chaining) a hodnoty soli (získanou hash funkcí SHA-1) pro ochranu certifikátu a pro ochranu privátních klíčů je využíván mechanismus generování trojice tajných klíčů (vytvořených šifrou 3DES v módu Cipher Block Chaining) a inicializačního vektoru z hesla, a hodnoty soli získanou také hash funkcí SHA-1. A tak je nutné zadat heslo k tomuto kontejneru a zároveň zabezpečit nově vzniklý privátní klíč heslem.



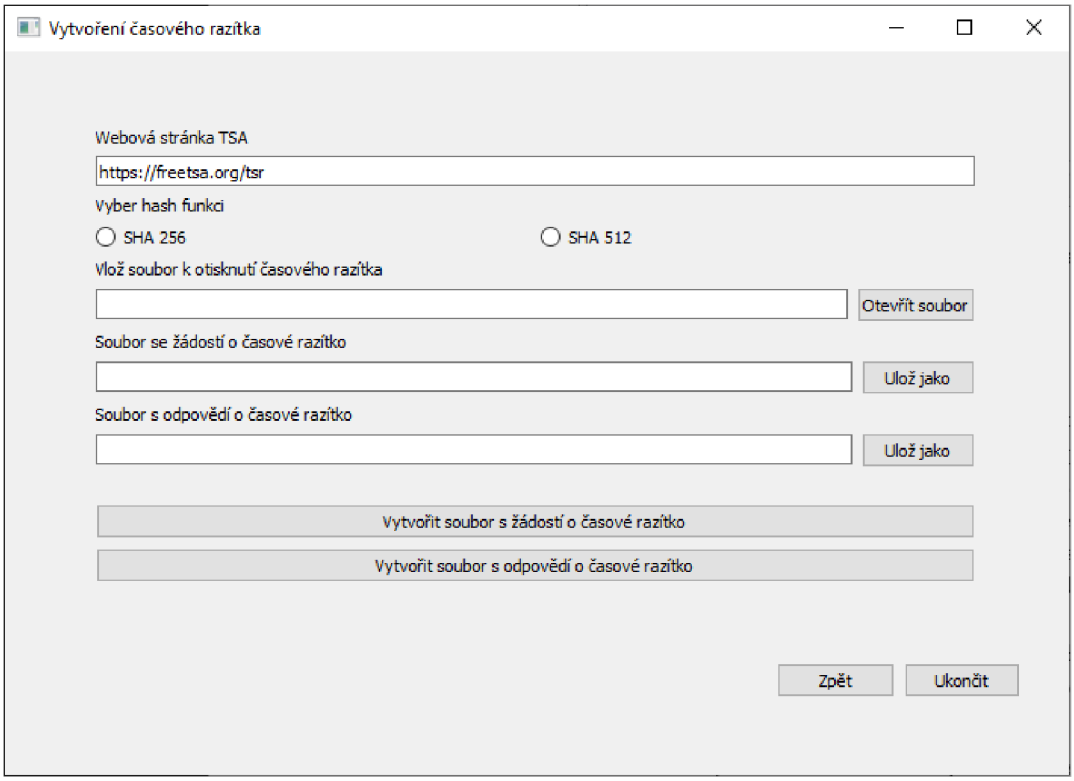
Obrázek 2.9: Digitální podpis pomocí certifikátu uloženého v souboru PFX.

Další nabídka z menu aplikace přenese uživatele na rozcestník k časovému razítku (viz obrázek 2.10), přesněji pak k jeho vytvoření, zobrazení informací o časovém razítku a jeho ověření.



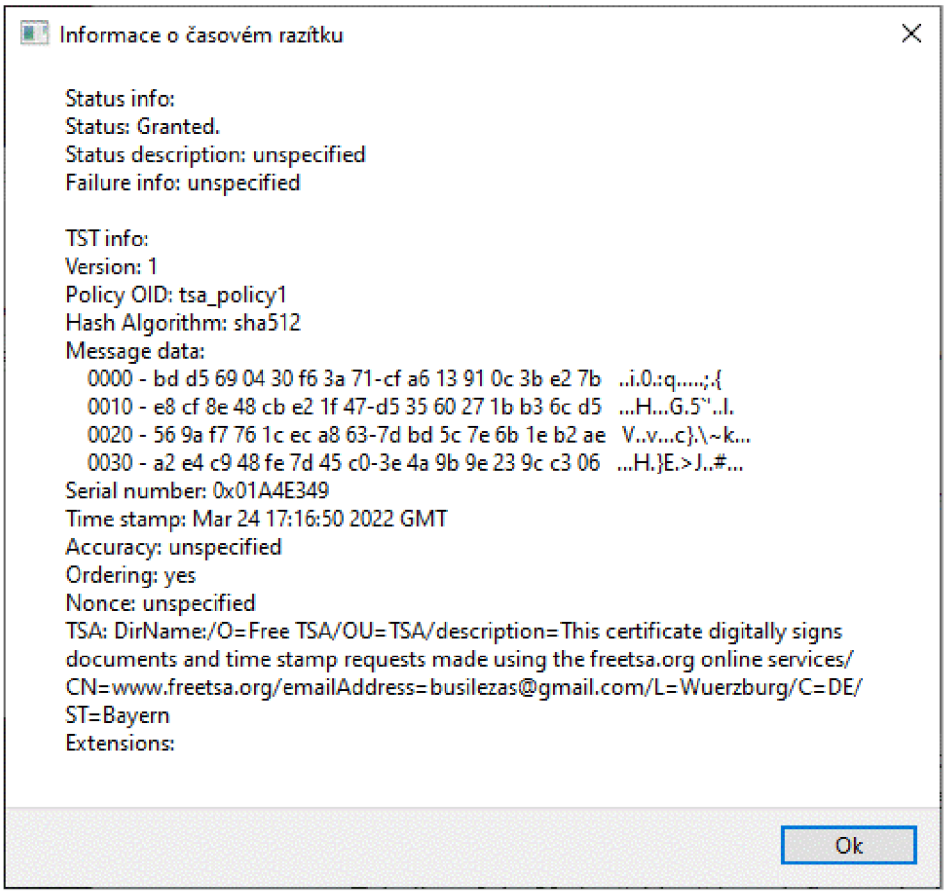
Obrázek 2.10: Rozcestník časových razítek.

Vytváření časového razítka vyžaduje zadat webovou stránku autority časových razítek (v aplikaci je použita a automaticky vložena volně dostupná autorita FreeTSA), zvolit danou hash funkci, zvolit soubor k otisknutí, uložit soubory s žádostí o časové razítko (formát souboru TSQ) a následnou odpovědí na tuto žádost. Soubor s žádostí je využit k vytvoření následné odpovědi. Soubor s příponou .tsr (soubor s odpovědí) slouží jednak k prokázání, že daný soubor existoval v daný moment, ale také dokáže zobrazit obsah časového razítka (např. datum, autoritu, hash funkci apod.). Tato možnost rozcestníku časových razítek je znázorněna na obrázku 2.11).



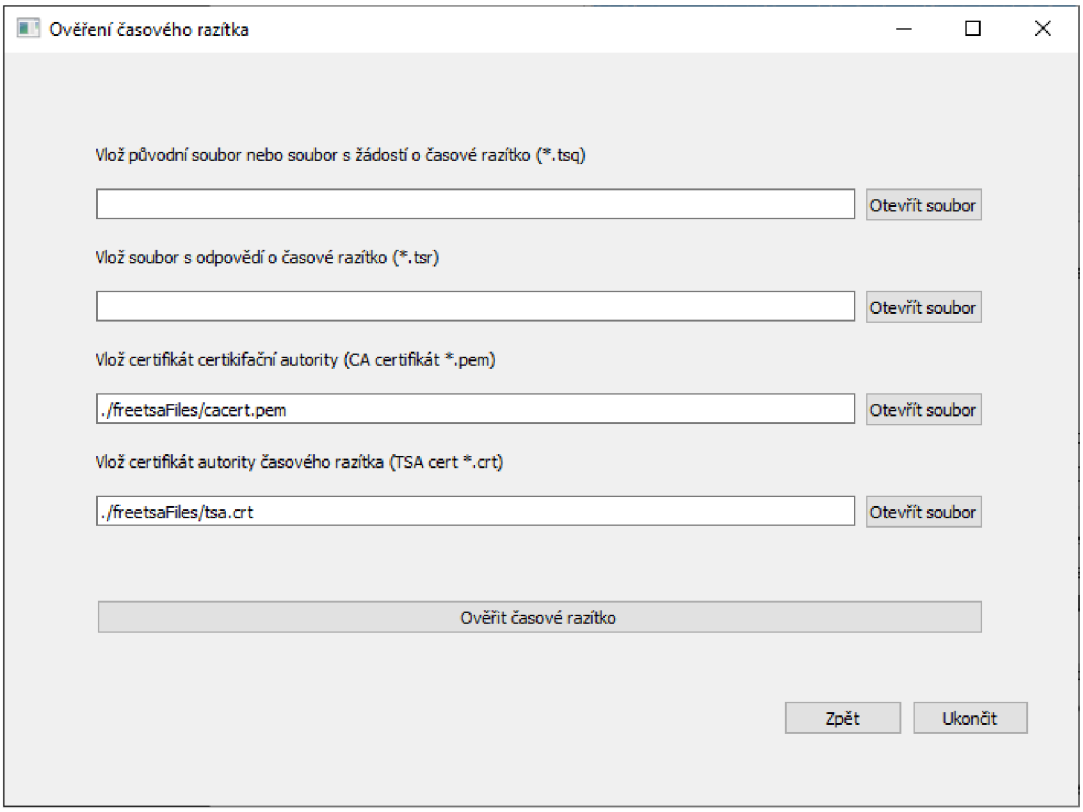
Obrázek 2.11: Vytvoření časového razítka.

Druhá volba rozcestníku vede k zobrazení obsahu časového razítka právě ze souboru ve formátu TSR, zvolit lze mezi zkráceným obsahem (zobrazí pouze datum a autoritu časového razítka) a úplným obsahem, který zobrazí veškeré informace. Výsledek zobrazení úplného obsahu časového razítka je znázorněna na obrázku 2.12.



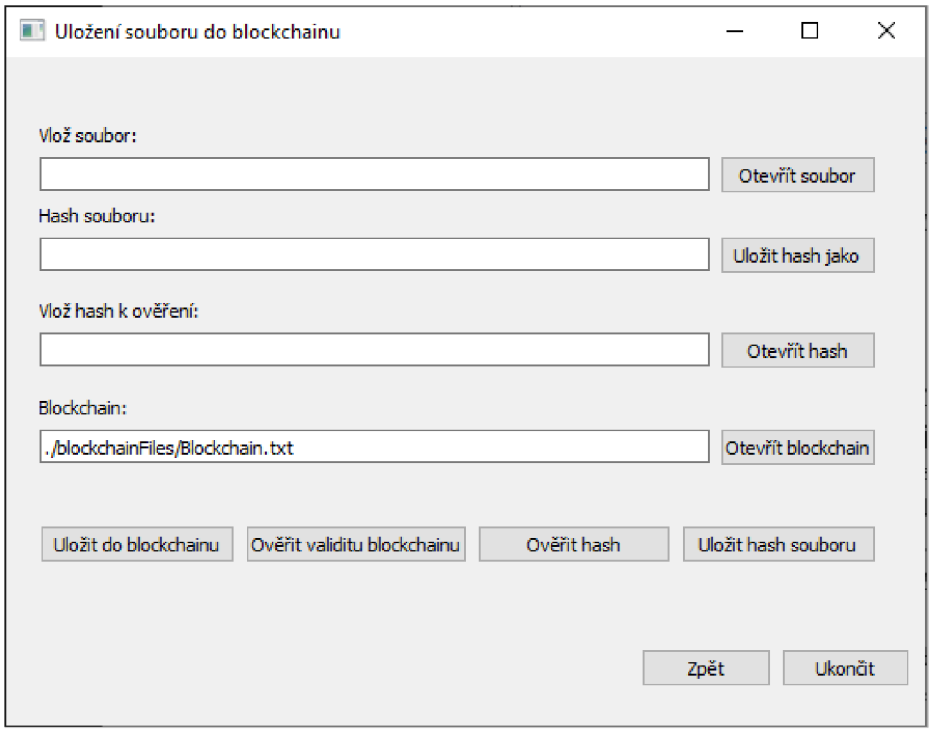
Obrázek 2.12: Zobrazení informací o časovém razítku.

Poslední tlačítko rozcestníku, nepočítaje tlačítko „Zpět“, otevře nové okno pro ověření časového razítka (viz obrázek 2.13). K ověření je zapotřebí mít soubor, u něhož se bude ověřovat jeho nepozměnění, dále pak soubor s odpovědí od časové autority (soubor s časovým razítkem hashe souboru, který ověřujeme – avšak před jeho možným pozměněním) a dvojici certifikátu (certifikát časové autority a certifikát certifikační autority). V případě, že bylo využito při vytváření časového razítka autority, která je výchozím stavu zvolena v aplikaci, není zapotřebí vkládat žádné certifikáty (pro tuto autoritu jsou certifikáty automaticky vloženy a není nutné měnit jejich cesty, jedině v případě jestliže je aplikace spuštěna z jiného místa než ze složky, se kterou je distribuována).



Obrázek 2.13: Okno aplikace pro ověření časového razítka.

Jako poslední možnost celé aplikace je využití technologie blockchain (viz obrázek 2.14). Tato část aplikace funguje jen jako nastínění další možné varianty pro dlouhodobou ochranou autorských práv u digitálních děl. Zapotřebí je pouze požadovaný soubor k ochraně, a soubor s řetězcem bloků. Po vložení souboru se vytvoří jeho hash, který je třeba uložit pro budoucí ověření. Následně otisk souboru je nezbytné vložit do bloku, jenž bude uložen do řetězce bloků. Uvnitř bloku se nachází další hodnoty jako například čas, kdy byl blok vložen do řetězce (údaj data a času je získáván z NTP serveru provozovatele CZ.NIC) nebo hash předchozího bloku. Při ověřování je zkoumáno, zdali se otisk souboru nachází v nějakém bloku řetězce.



Obrázek 2.14: Okno aplikace využívající k ochraně technologii blockchain.

2.3 Implementace funkcí aplikace

Při tvorbě grafických částí aplikací bylo nutné soubory vytvořené právě programem Qt designer následně převést do souboru, který bude moci být dále editován za účelem doplnění funkcí aplikace. Pro převedení takového souboru je nutné v příkazovém řádku zadat tento příkaz:

```
pyuic5 -x -o NazevNovehoPYSouboru.py NazevSouboruZQTDesigneru.ui
```

Po vytvoření jednotlivých skriptovacích souborů, pro každé okno aplikace jeden, lze do těchto souborů již dopisovat jednotlivé funkce, které následně přiřadit k daným tlačítkům, a tak dotvořit zbylou část aplikace.

Vkládání vodoznaku do obrazového souboru

Pro vkládání vodoznaku do obrazového souboru ve formátu JPG byl využit modul pathlib, pomocí kterého bylo zajištěno, že při vkládání cest k souborům byly doopravdy vloženy cesty k souborům, které již existují. V případě, že jsou vyplněny správně veškeré cesty k souborům, je možné pokračovat otevřením editoru, což vyvolá další okno aplikace. Při tvorbě editoru je použita knihovna OpenCV a funkce imutils (součást OpenCV, která dokáže např. rotaci a změnu velikosti daného obrázku). Knihovna je zde využita pro načtení obrázku a vodoznaku a pro jejich následné sloučení a uložení.

V editoru se nachází několik posuvných jezdců, kdy dvojice umístěná vlevo a pod obrázkem slouží k přesnějšímu umístění vodoznaku v obrázku. Trojice jezdců slouží k doplňkovým vlastnostem vodoznaku, přesněji pak velikost, rotaci a průhlednosti. Všechny jezdce mění zobrazovanou dvojici obrazových souborů dynamicky. Jestliže je změněn vodoznak a ať už velikostně, natočením nebo jeho pouhým umístněním bude zasahovat mimo obrázek, zobrazí se varovná hláška o přesahu vodoznaku z obrázku.

Obrázek 2.15: Vývojový diagram části aplikace se vkládáním vodoznaku.

Skrytí zprávy do obrazového souboru

V tomto případě, kdy je také pracováno s obrazovým souborem je zde využita knihovna OpenCV a modul math, jelikož je nutné použít zaokrouhlení čísla a tato funkce není dostupná bez importu právě této knihovny. Technika metody steganografie je zde zvolena, dříve popsaná, metoda substituce. Při programování této části aplikace bylo vycházeno z kódu zveřejněného programem Section's Engineering Education (EngEd), který se zaměřuje na poskytování komunitních zkušeností mezi studenty vysokých škol se zaměřením na počítačovou vědu.[1] Kód je uvolněn pod licencí Apache License 2.0

Obrázek 2.16: Vývojový diagram skrývání zprávy do souboru s formátem JPG.

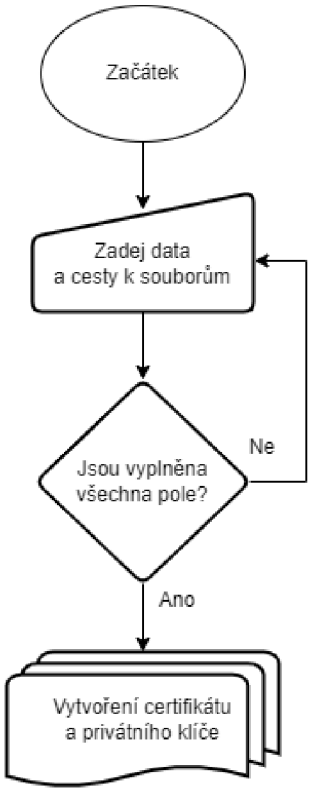
Vytvoření self-signed certifikátu

Vytvoření certifikátu podepsaného sebou samotným bylo zvoleno, neboť získat digitální certifikát, který je volně zdarma dostupný je takřka nemožné. Pro získání certifikátu je nutné prokázat fyzicky identitu uživatele při návštěvě pobočky poskytovatele certifikační služby a za následný certifikát zaplatit.

[1] Viz https://www.section.io/engineering-education/steganography-in-python/

Při vytváření self-signed certifikátu v aplikace je nutné zadat jméno, emailovou adresu, stát a jeho dvou písmennou zkratku. Nadále pak ještě celkovou dobu platnosti udávanou ve dnech, a místa, kam bude uložený certifikát a privátní klíč.

K vytvoření je využit modul pyOpenSSL, který dokáže vytvořit certifikát X.509 a modul random, který přidělí náhodné sériové číslo certifikátu. Nejdříve je vytvořena dvojice klíčů pomocí algoritmu RSA o délce klíče 4096 bitů. Následně se vytvoří certifikát z hodnot, které se v předchozím kroku zadali, a do certifikátu je přidán veřejný klíč a vydavatel (v tomto případě shodný se jménem, které bylo zadáno, jelikož se jedná o certifikát vydaný sám sebou). V posledním kroku se privátní klíč a certifikát uloží.



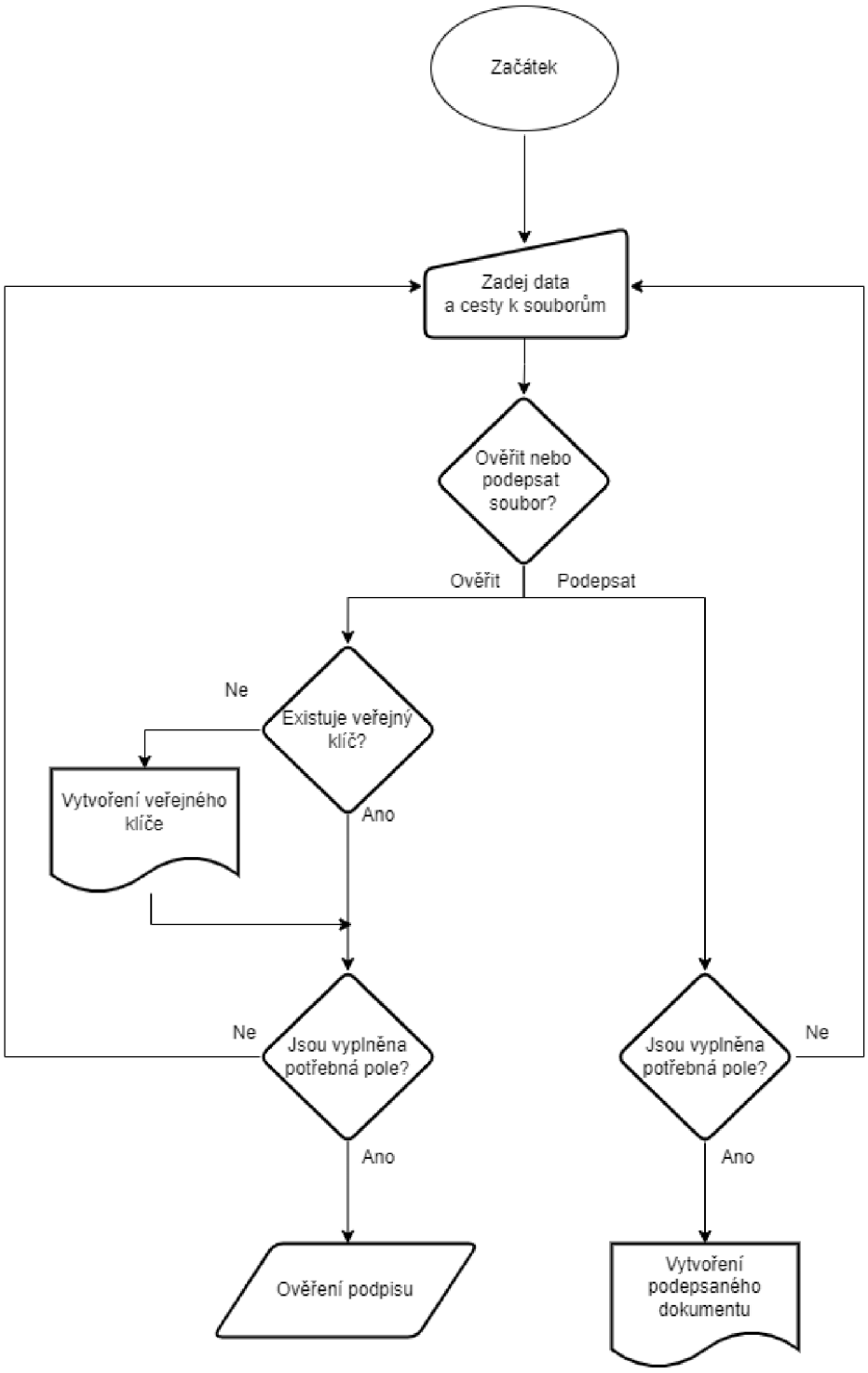
Obrázek 2.17: Vývojový diagram vytvoření self-signed certifikátu a privátního klíče.

Digitální podpis pomocí self-signed certifikátu

Certifikát utvořený v předchozím kroku lze využít při podepisování dokumentu. Jestliže cíl úkonu je podepsat soubor, postačí vlastnit soukromý klíč. Ověřování daného souboru s podepsaným souborem už avšak vyžaduje veřejný klíč. V případě, kdy není vlastněn nebo nebyl prozatím vypočítán a uložen veřejný klíč, je nutné zadat privátní klíč a veřejný tak dopočítat. Poté nic nebrání využití aplikace pro podepsání a ověření souboru pomocí digitální podpisu s využitím self-signed certifikátu.

Při programování této části aplikace je využito kryptografické knihovny OpenSSL (implementuje protokoly SSL, TLS a nástroje pro kryptografii). [42] Tato knihovna není

primárně v operačním systému Windows implementována, tudíž je nutné ji doinstalovat. Jelikož knihovna je spustitelná z příkazového řádku, tak je nutné naimportovat také knihovnu os, která otevře příkazový řádek a spustí daný příkaz.



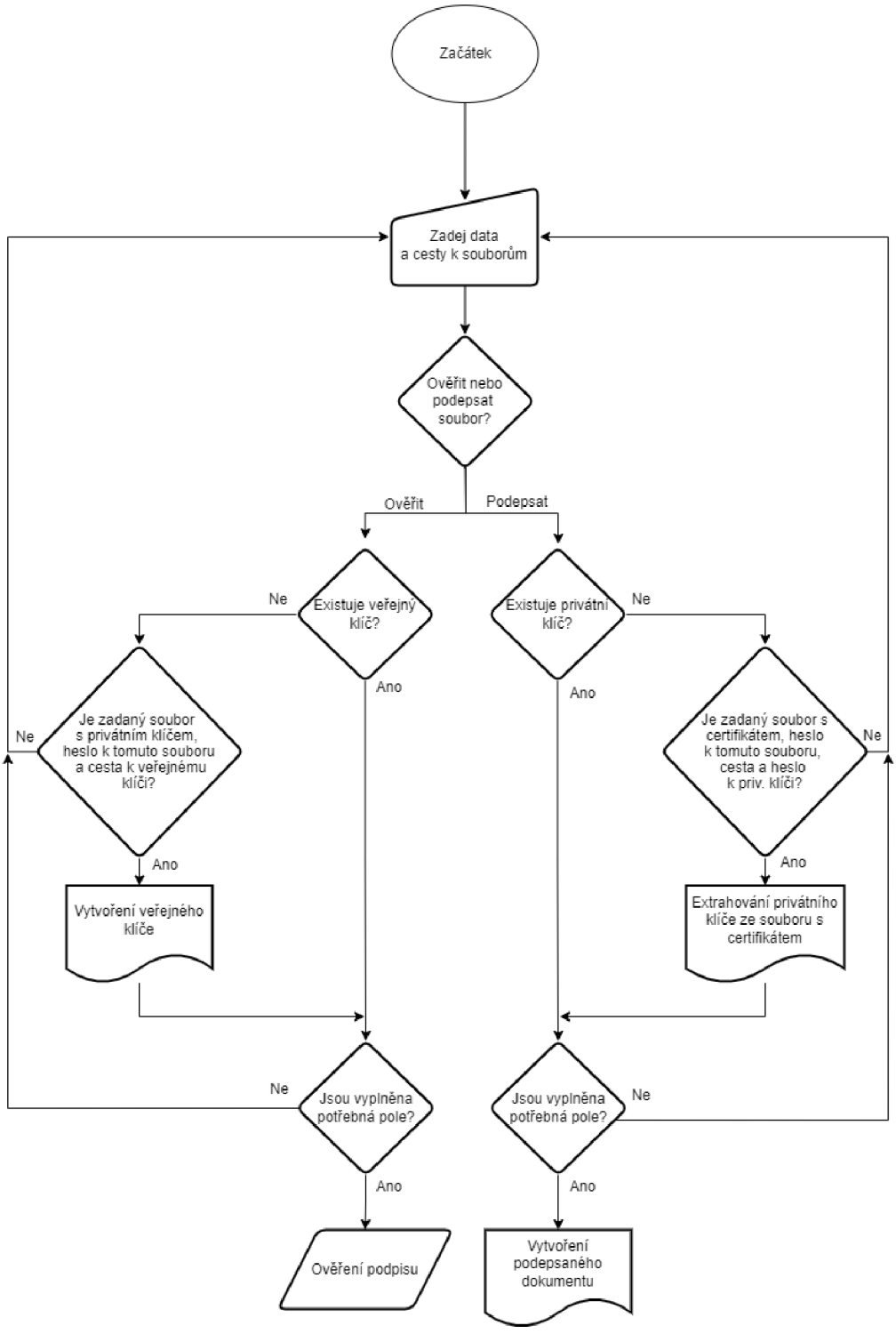
Obrázek 2.18: Vývojový diagram digitálního podpisu s využitím self-signed certifikátu a privátního klíče.

Digitální podpis pomocí certifikátu uloženého v souboru formátu PFX

Jedná se o digitální podpis, kde se využívá pro podpis, stejně jako v předchozí možnosti digitálního podpisu, asymetrická kryptografie. Avšak tentokrát s rozdílem, že certifikát spolu s privátním klíčem jsou distribuovány pospolu v kryptograficky zašifrovaném kontejneru formátu PFX. Daný kontejner může mít ještě příponu .pkcs12 nebo .p12. Všechny tyto formáty jsou formáty definovány v RFC 7292 a jedná se o standardy PKCS (Public Key Cryptography Standard). [43]

Tento formát se často používá při exportování certifikátu, tudíž aplikace by měla fungovat s jakýmkoliv certifikátem uložený právě v tomto formátu. Pro test aplikace byl vytvořen účet na stránce www.codegic.com, která poskytuje služby vytvoření zkušebního certifikátu na dobu určitou (přesněji pak na 2 měsíce). Při vytváření certifikátu je zadáváno heslo pro PFX kontejner, které je nutné zadat i do aplikace, neboť kontejner je heslem zabezpečený a bez správně zadaného hesla nebudou certifikát a soukromý klíč úspěšně extrahovány. Ještě před samotným extrahováním je nutné zadat heslo také pro soukromý klíč, nelze jej totiž z kontejneru extrahovat bez hesla.

Programování této části bylo opět využito knihovny OpenSSL a modulu os. V obou případech digitálních podpisů bylo využíváno hash funkce SHA-512.



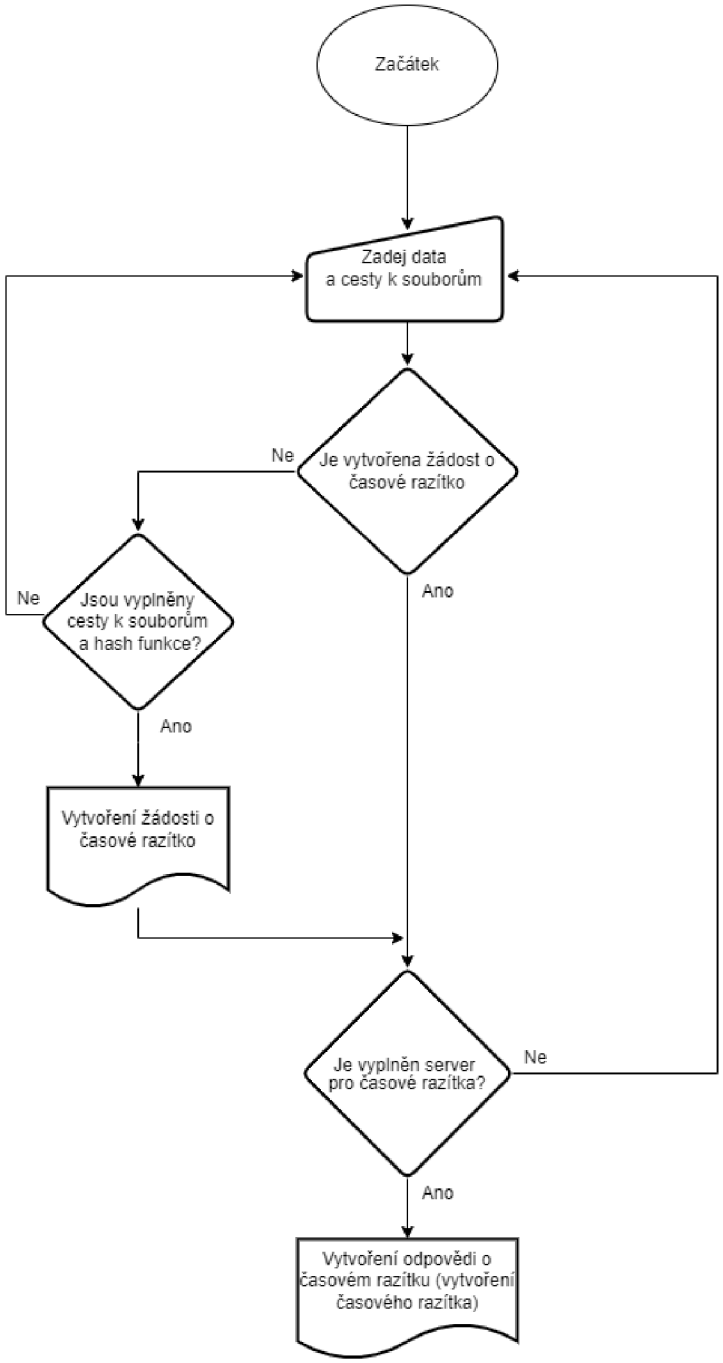
Obrázek 2.19: Vývojový diagram digitálního podpisu s využitím certifikátu a klíče uloženého v kontejneru formátu PFX.

Vytvoření časového razítka

U vytváření časového razítka je nutné znát adresu serveru časových razítek, přičemž v aplikaci je primárně vložen volně dostupný server autority FreeTSA, avšak lze také

použít jiné servery. Při tvorbě žádosti o časové razítko je třeba zvolit hash funkci. Po vytvoření žádosti o časové razítko je žádost odeslána na námi zadaný server. Nazpět ze serveru přijde časové razítko uložené v souboru s odpovědí.

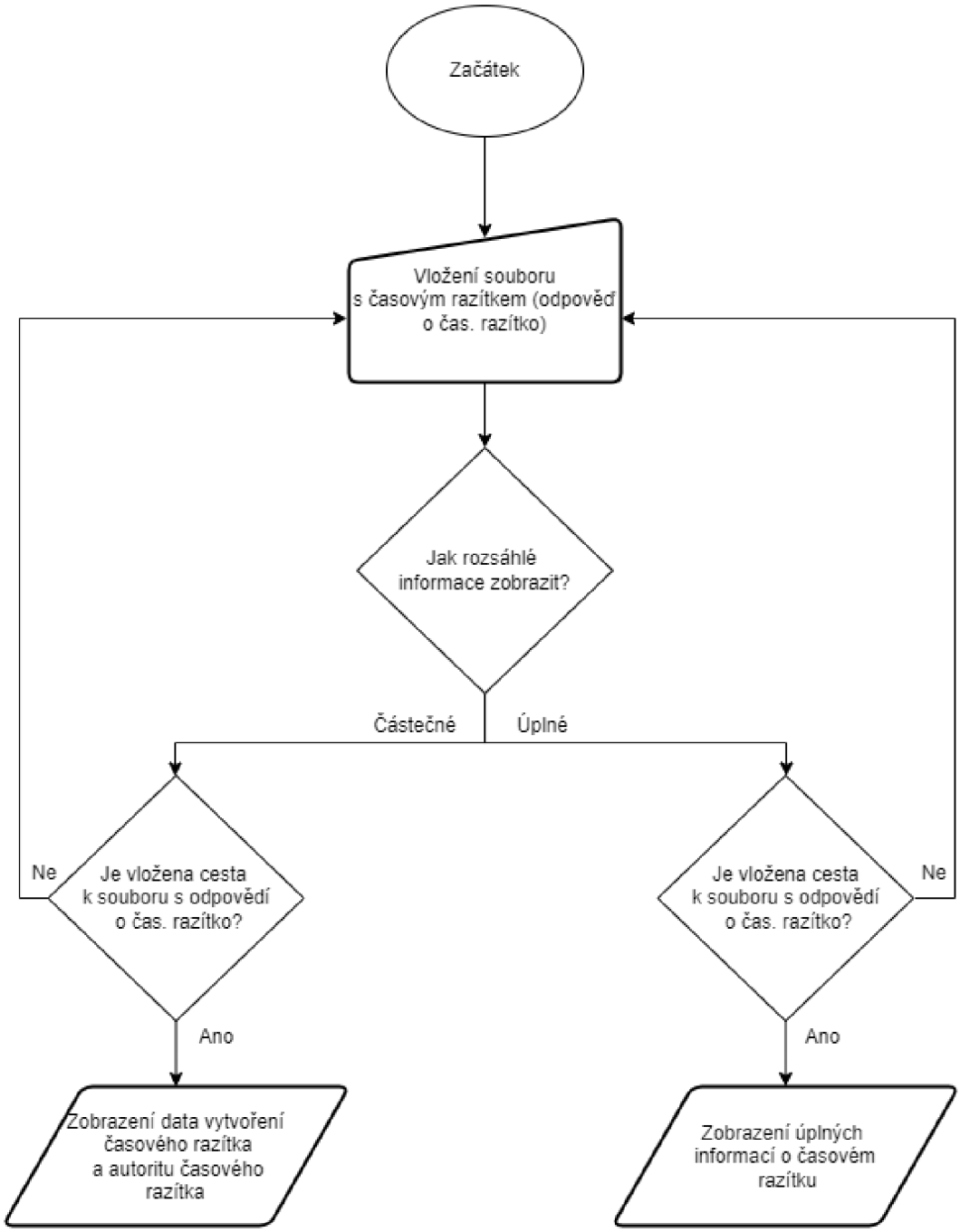
Stejně jako u digitálního podpisu, tak i při veškerých úkonech spojené s časovým razítkem pracuje kód s příkazovým řádkem, a tak je i zde implementován vždy modul os a využívána knihovna OpenSSL. V případě, kdy je posílána žádost o časové razítko na server časové autority je použit příkaz curl, který zajišťuje přenos dat.



Obrázek 2.20: Vývojový diagram vytvoření žádosti o časové razítko.

Zobrazení obsahu časového razítka

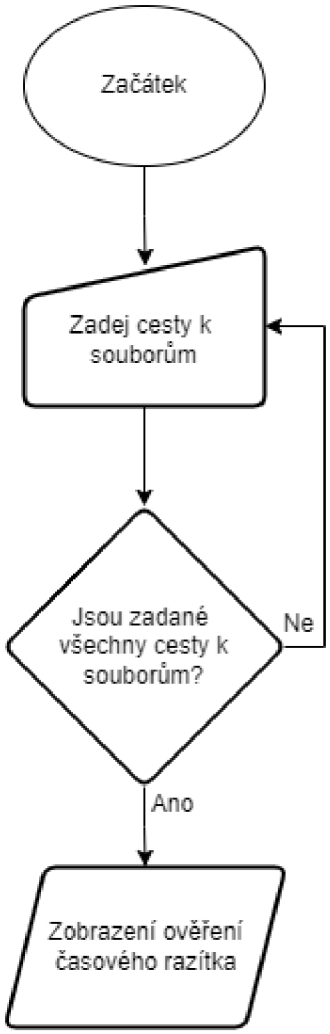
Jelikož časové razítko, respektive soubor s časovým razítkem neboli odpověď na žádost o časové razítko není v podobě, kterou dokáže člověk přečíst pouze pomocí textového editoru, je nutné zadat příkaz ze knihovny OpenSSL. Aplikace dokáže zobrazit výtah informací o časovém razítku nebo celý obsah daného razítka.



Obrázek 2.21: Vývojový diagram zobrazení informací o časovém razítku (odpovědi o časové razítko).

Ověření časového razítka

U ověření časového razítka se porovnává původní soubor, ze kterého byla vytvořena žádost a následně odpověď na žádost o časové razítko. Při tomto ověřování je nutné mít staženou dvojici certifikátu, a to certifikát certifikační autority a certifikát autority časového razítka s délkou klíčů 4096 bitů a podpisovým algoritmem SHA512WithRSAEncryption (RSA-SHA512). Pokud nebyl při vytváření časového razítka použit jiný server, než který byl předem vložený do aplikace, není nutné ani v tomto případě měnit dané certifikáty. Certifikáty, které jsou v aplikaci vložené jsou stažené a uložené ve složce a distribuované s aplikací.



Obrázek 2.22: Vývojový diagram ověření časového razítka.

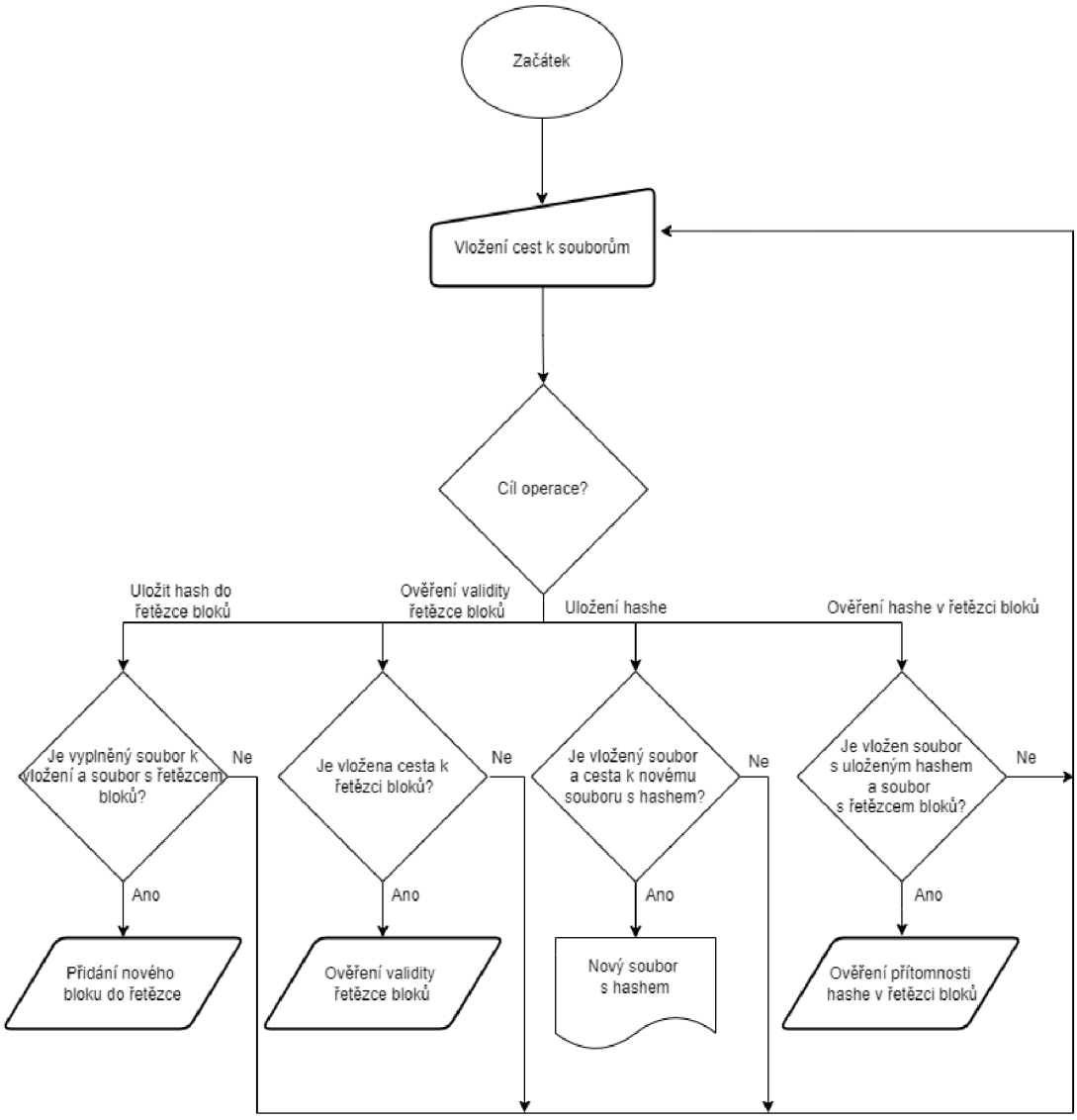
Využití technologie blockchain při ochraně autorských práv

V této části aplikace je poukázáno na možné využití technologie blockchain při dlouhodobé ochraně autorských práv. Aplikace v tomto bodě funguje jen jako nastínění možné ochrany. K plnému využití aplikace v praxi by bylo ještě vhodné doimplementovat

další části aplikace. Mezi ně lze zařadit decentralizované uložení řetězce, vyřešení situace přidání dvou bloků ve stejný moment apod. Jako první se do aplikace vkládá soubor, který je určený k uložení do řetězce bloků. Daný soubor se přečte v binární podobě a utvoří se z něho hash (SHA-256). Tento hash se poté ukládá do bloků spolu s dalšími informacemi (např. data vytvoření bloku – k vložení UTC data a času je využit NTP server provozovaný společností CZ.NIC, s indexem o kolikátý blok v řetězci se jedná apod.) a slouží ke zpětnému ověření, že soubor v daném tvaru (odpovídající otisku) byl v tento den vložen do řetězce bloků. Proto je dobré tento hash uložit. Soubor s řetězcem je předem vložen v aplikaci a v samotném souboru už se řetězec nachází, tudíž budeme stávající sekvenci bloku rozšiřovat o nové bloky. Při ověřování validity řetězce je zkoumáno, zda je shodná hodnota uložená v bloku jako hash předešlého bloku s hodnotou, která je vypočítávána jako hash předešlého bloku. A dále je zkoumáno, jestli není změněna hodnota „proof“ (tzv. hodnota „nonce“), tato hodnota je přidána k datům z důvodu těžení jednotlivých bloků řetězců. Podmínkou pro vytěžení bloků je, aby první čtyři znaky digitálního otisku byly nuly.

Tvorba kódu vycházela z kódu pro technologii blockchain zveřejněného programem Section's Engineering Education (EngEd)[2], který se zaměřuje na poskytování komunitních zkušeností mezi studenty vysokých škol se zaměřením na počítačovou vědu. Kód je uvolněn pod licencí Apache License 2.0. V kódu jsou implementovány moduly ast (Abstract Syntax Trees), hashlib, JSON (JavaScript Object Notation), pathlib, ntplib a datetime.

[2] Viz https://www.section.io/engineering-education/how-to-create-a-blockchain-in-python/



Obrázek: 2.23: Vývojový diagram ochrany autorských dat pomocí technologie blockchain.

2.4 Spuštění programu

Jestliže na počítači není stažený Python, je nutné ho stáhnout. Pokud už je na zařízení nainstalovaný Python, celá aplikace by měla jít spustit pouhým spuštěním souboru „Aplikace.pyw“. Avšak pro správné fungování celé aplikace je nutné zkontrolovat a případně doinstalovat některé knihovny pomocí těchto příkazů:

```
pip install pyopenssl
pip install opencv-python
pip install pyqt5
pip install imutils
pip install ntplib
```

Dále je nutné mít na zařízení nainstalovanou knihovnu OpenSSL, která lze stáhnout z webových stránek.[3] Po instalaci této knihovny je nutné zadat její cestu do systémových proměnných.

Aplikaci lze také spustit po vytvoření spustitelného souboru s příponou .exe. K tomu, aby byl vytvořen spustitelný soubor je zapotřebí mít nainstalovaný balíček PyInstaller (dostupný ve verzi 5.0.1). Ten sdružuje python skript a všechny jeho závislosti do jednoho balíčku, který lze následně spustit. [44] K instalaci balíčku PyInstaller je nutné zadat do příkazového řádku tento příkaz:

```
py -m pip pyinstaller
```

K převedení na spustitelný soubor je nutné po instalaci zadat do příkazového řádku vyvolaného ve složce, kde se python skript nachází, tento příkaz:

```
pyinstaller --onefile -w ***.py
```

kde *** znamená název souboru. Po provedení tohoto příkazu se v téže složce vytvoří nový soubor s příponou .spec a složka dist. Nový soubor s příponou je možné vymazat a uvnitř dist složky se bude nacházet spustitelný soubor.

K snazší distribuci aplikace lze z celé složky, ve které je aplikace uložena, vytvořit instalační soubor. Ten je v porovnání se složkou a všemi jejími soubory snazší sdílet, neboť se jedná pouze o jeden soubor. K vytvoření instalačního souboru je nutné celou složku zkomprimovat a následně využít NSIS (Nullsoft Scriptable Install System) – což je aplikace k vytváření spustitelných souborů na operačním systému Windows. [45]

2.5 Porovnání implementovaných ochran

Implementované ochrany autorských práv v aplikaci mají rozdílné doby platnosti. Jednotlivé doby platnosti jsou níže rozepsané, a nakonec je vytvořena tabulka pro přehlednější souhrn daných dob (viz Tabulka 2.1: Tabulka).

Elektronický podpis je platný do té doby, dokud je platný certifikát (do té doby je možné podpis ověřit, v případě že je certifikát propadlý nebo není vlastněn, dokazování platnosti je poté o něco složitější, neboť certifikát je revokován – zneplatněn). Z pravidla se certifikáty (u českých autorit, vydávající kvalifikované certifikáty pro elektronický podpis, např. Česká pošta, s.p.) vydávají na 1 nebo 3 roky.

Pro prodloužení elektronicky podepsaného dokumentu lze využít časová razítka, ty se vystavují na 3 až 5 let. Po uplynutí doby je možné dokument znovu orazítkovat. Důvod, proč nejsou razítka a ani elektronický podpis platné delší dobu je, neboť jsou založeny na matematických operacích, které jsou momentálně výpočetně složité, avšak to nemusí platit za pár let (některé kryptografické části metod mohou být prolomeny).

[3] Např. z https://slproweb.com/products/Win32OpenSSL.html

Nejdelší možnou ochranou autorských práv elektronického dokumentu je využití kryptoměn. Zde je platnost téměř časově neomezená. Úskalím může být ztráta původního souboru (čímž posléze nelze prokázat validitu mezi souborem a otiskem, který je uložený v řetězci) nebo například zaniknutí řetězce bloků (zaniknutím celé kryptoměny). Proti prvnímu zmíněnému úskalí některé společnosti zajišťují zálohu souboru na jejich serveru (např. Certoo zálohuje soubor na 10 let). Proti zaniknutí řetězce se nelze nijak chránit, avšak v případě použití známějších kryptoměn lze předpokládat jejich podstatně delší existenci než v případě použití těch méně známých.

Tabulka 2.1: Tabulka doby platnosti jednotlivých metod ochrany.

Název metody	**Doba platnosti v letech**
Elektronický podpis	1 nebo 3
Časové razítko	3 až 5
Využití kryptoměn (technologie blockchain)	Není omezená[4]

2.6 Systémové požadavky aplikace

Celá aplikace byla vytvořena na platformě Microsoft Windows 10 verze 21H1 (build 19044.1645) a je plně funkční na dané verzi MS Windows. Jelikož se jedná o podstatně jednoduchou python aplikaci, v její zpětné kompatibilitě na starší verze Windows by neměl být problém. Staršími verzemi je myšleno Windows 8 a novější, avšak Windows 7 a starší již nelze použít, neboť není na těchto verzích Windows podporován Python 3.10.3 v němž je aplikace napsána. Pro správné fungování celé aplikace je dále nutné být připojen k internetu. Alespoň v případě, kdy bude aplikace použita pro práci s časovými razítky (nutná komunikace se serverem autority časových razítek) a práci s využitím technologie blockchain (je získáván přesný čas ze serveru).

Z hardwarové stránky, aplikace byla sepsána na notebooku, který měl následovnou konfiguraci:

- Procesor: AMD Ryzen 5 4600H
- Fyzická paměť (RAM): 16 GB
- Pevný disk: Samsung SSD 512 GB
- Grafická karta:
  - Dedikovaná: NVIDIA GeForce RTX 2060 6 GB
  - Integrovaná: AMD Radeon (TM) Graphics

[4] Za splnění podmínek, že autor disponuje certifikátem o vlastnictví a případně ještě nepozměněným souborem, který byl vložen do řetězce bloků nebo že kryptoměna není zaniklá.

ZÁVĚR

Bakalářská práce s názvem Ochrana autorských práv elektronických dokumentů pojednává především o zachovávání informací o tom, kdo je vlastníkem vytvořeného dokumentu a jak lze ochrany takových informací dosáhnout.

Z počátku teoretické části byl definován elektronický dokument a jaký je rozdíl mezi elektronickým a listinným dokumentem. Dále se věnovala rozdělením souborů a jejich typům. Neboť elektronickým dokumentem, jak již bylo řečeno v práci, je jakákoliv obrazová, textová či audio informace a nejčastěji jsou právě tyto informace uchovávány v podobě různých souborů. Následně jsou v práci definovány autorská práva. Poslední kapitola teoretické části práce je věnována analýze ochran elektronického dokumentu. Jsou zde popsány nejen různé typy kryptografických možností ochran, ale je zde i jedna podkapitola věnována steganografii či technologii blockchain.

V praktické části byla vytvořena aplikace s grafickým rozhranním, ve které jsou implementovány některé z ochran zmíněných v teoretické části. Jmenovitě pak steganografie, digitální vodoznak, digitální podpis, časové razítko a využití technologie blockchain. Při vkládání digitálního vodoznaku do obrázku je využito metody, kdy slučujeme dva obrazové soubory. Technikou steganografie byla zvolena metoda, která ukládá zprávu do posledního bitu pixelu. V případě digitálního podpisu se využívá asymetrické kryptografie (algoritmu RSA). Pro časová razítka byla zvolena volně dostupná autorita FreeTSA, avšak lze použít i jiné. Část aplikace s technologií blockchain je utvořena jako koncept, který dokáže přiblížit, jak je možné takovou technologii zhruba použít v případě ochrany autorských práv. Avšak zde není žádná ochrana proti pozměnění souboru s řetězcem bloků nebo případ, kdy dva uživatelé přidají blok v jeden moment a další možná bezpečností úskalí.

Při zpětné analýze vytvořeného programu jsem došel k závěru, že ohledně vkládání vodoznaku existují aplikace, které odstraní vložený vodoznak. Při odstraňování vodoznaku cestou online aplikací jsem se setkal s aplikací, ve které je možné označit vodoznak a vyznačenou část poté zamaže. Taková část, kde se vodoznak nacházel, může být značně zdeformována a obrázek být tak dále nepoužitelný. Ochranou proti takovému odstraňování je vkládat dostatečně velký vodoznak (což by při odstraňování způsobilo deformaci velké plochy obrázku) a nejlépe vícebarevný (to ztíží mazání vodoznaku). K ověření steganografie by bylo nutné zhotovit dekodér. Zpráva samotná ukrytá v obrázku však je, nicméně bude smazána či přepsána v případě, kdy by bylo využito stejné metody pro skrytí další zprávy či by byl obrázek pozměněn.

Jako budoucí možné vylepšení aplikace by bylo vhodné přidání dekodéru pro steganografii, dopracovat část aplikace s technologii blockchain tak, aby byla bezpečnější, řetězec decentralizovaný a řešila problém přidání dvou bloků v jeden stejný moment či vygenerování certifikátu s údaji vkladatele. Dále lze do aplikace přidat další metody ochran, např. elektronickou pečeť.

LITERATURA

[1] ČESKO. Zákon č. 499/2004 Sb., zákon o archivnictví a spisové službě a o změně některých zákonů. In: *Zákon pro lidi* [online]. © AION CS 2010-2021 [cit. 2021-11-1]. Dostupné z: https://www.zakonyprolidi.cz/cs/2004-499

[2] LECHNER, Tomáš. *Elektronické dokumenty v právní praxi*. Praha: Leges, 2013. Praktik (Leges). ISBN 978-80-87576-41-0.

[3] POLČÁK, Radim. *Právo informačních technologií*. Praha: Wolters Kluwer, 2018. S. 215. Právní monografie (Wolters Kluwer ČR). ISBN 978-80-7598-045-8.

[4] Nařízení Evropského parlamentu a Rady (EU) č. 910/2014 ze dne 23. července 2014 o elektronické identifikaci a službách vytvářejících důvěru pro elektronické transakce na vnitřním trhu a o zrušení směrnice 1999/93/ES. In: Úřední věstník Evropské unie [online]. Kapitola I, článek 3, odst. 35 [cit. 2021-11-12]. Dostupné z: https://eur-lex.europa.eu/legal-content/CS/TXT/PDF/?uri=CELEX:32014R0910&from=CS

[5] Elektronický dokument. *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2021-11-27]. Dostupné z: https://cs.wikipedia.org/wiki/Elektronick%C3%BD_dokument

[6] Nařízení Evropského parlamentu a Rady (EU) č. 910/2014 ze dne 23. července 2014 o elektronické identifikaci a službách vytvářejících důvěru pro elektronické transakce na vnitřním trhu a o zrušení směrnice 1999/93/ES. In: Úřední věstník Evropské unie [online]. Kapitola III, oddíl 4, článek 25 [cit. 2021-12-4]. Dostupné z: https://eur-lex.europa.eu/legal-content/CS/TXT/PDF/?uri=CELEX:32014R0910&from=CS

[7] CZECHPOINT. Konverze. Ministerstvo vnitra České republiky [online]. MVČR, © 2021 [cit. 2021-11-25]. Dostupné z: https://www.czechpoint.cz/public/verejnost/autorizovana-konverze/

[8] ČESKO. Zákon č. 300/2008 Sb., Zákon o elektronických úkonech a autorizované konverzi dokumentů. In: *Zákon pro lidi* [online]. © AION CS 2010-2021 [cit. 2021-12-1]. Dostupné z: https://www.zakonyprolidi.cz/cs/2008-300

[9] Co je to soubor? [online]. © BoBr 1995 [cit. 2021-11-18]. Dostupné z: http://it.pedf.cuni.cz/~bobr/ucspoc/ossoub.htm

[10] TECHLIB. Formát souborů. [online]. [cit. 2021-11-18]. Dostupné z: https://tech-lib.eu/definition/file_format.html

[11] SHARPENED PRODUCTION. What is difference between Binary and text files? *Fileinfo.com* [online]. June 3, 2010 – Updated December 21, 2011 [cit. 2021-11-18]. Dostupné z: https://fileinfo.com/help/binary_vs_text_files

[12] HARSH, Kumar. What is Binary File? In: *CAREERKARMA*. [online]. Feb 5, 2021. [cit. 2021-11-21]. Dostupné z: https://careerkarma.com/blog/what-is-binary-file/

[13] CZ.NIC. Typy Souborů [online]. © 2021 [cit. 2021-11-22]. Dostupné z: https://www.jaknainternet.cz/page/1720/typy-souboru/

[14] CRIDER, Michael. What Is a .DOCX File and How It Is Diffent From a .DOC File in Microsoft Word? In: *How-To Geek* [online]. MAY 2, 2017, 10:24 AM [cit. 2021-11-22]. Dostupné z: https://www.howtogeek.com/304622/what-is-a-.docx-file-and-how-is-it-different-from-a-.doc-file-in-microsoft-word/

[15] Portable Document Format *Wikipedia: the free encyclopedia* [online]. San Francisco (CA): Wikimedia Foundation, 2001- [cit. 2021-11-27]. Dostupné z: https://cs.wikipedia.org/wiki/Portable_Document_Format

[16] SEGGERN, Dietrich von. PDF standards explained: with focus on the newest [online]. September 15, 2021 [cit. 2021-11-19]. Dostupné z: https://www.pdfa.org/pdf-standards-explained-with-a-focus-on-the-newest/

[17] RATHOUZ, Vítězslav. Formáty obrazových souborů [online]. 16. 4. 2012 v 09:59 [cit. 2021-11-28]. Dostupné z: https://wiki.knihovna.cz/index.php?title=Form%C3%A1ty_obrazov%C3%BDch_soubor%C5%AF

[18] SILITONGA, Parasian a Irene Sri MORINA. Compression and Decompression of Audio Files Using the Arithmetic Coding Method [online]. May 2019 [cit. 2021-12-05]. Dostupné z: https://www.researchgate.net/publication/340573649_Compression_and_Decompression_of_Audio_Files_Using_the_Arithmetic_Coding_Method

[19] FILESTACK. The Complete List of Video File Formats and Codecs for Developers [online]. October 5, 2021 [cit. 2021-12-05]. Dostupné z: https://blog.filestack.com/thoughts-and-knowledge/complete-list-audio-video-file-formats/]

[20] SHARPENED PRODUCTION. .RAR File Extension. *Fileinfo.com* [online]. [cit. 2021-12-04]. Dostupné z: https://fileinfo.com/extension/rar

[21] HAVLOVÁ, Jaroslava. Správa digitálních práv. In: *KTD: Česká terminologická databáze knihovnictví a informační vědy (TDKIV)* [online]. Praha: Národní knihovna ČR, 2003- [cit. 2021-12-02]. Dostupné z: https://aleph.nkp.cz/F/?func=direct&doc_number=000015940&local_base=KTD

[22] About CC Licenses. Creativecommons.com [online]. [cit. 2021-12-02]. Dostupné z: https://creativecommons.org/about/cclicenses/

[23] BRETT, Daniel. Symmetric vs. Asymmetric Encryption: What's the Difference? In: *Trentonsystems* [online]. May 4, 2021 9:30:00 AM [cit. 2021-11-30]. Dostupné z: https://www.trentonsystems.com/blog/symmetric-vs-asymmetric-encryption

[24] ZEMAN, V.: *Digitální podpis* [přednáška]. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Aplikovaná kryptografie. [cit. 2021-12-05]

[25] GLUTTONY777. Difference between Direct and Arbitrated Digital Signature. In: *GeeksforGeeks* [online]. 15 Jan, 2020 [cit. 2021-12-04]. Dostupné z: https://www.geeksforgeeks.org/difference-between-direct-and-arbitrated-digital-signature

[26] HRBOTICKÝ, Lukáš. Elektronická identifikace a služby vytvářející důvěru pro elektronické transakce. Brno, 2018. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. [online]. [cit. 2021-12-05]. Dostupné z: https://www.vut.cz/www_base/zav_prace_soubor_verejne.php?file_id=175177

[27] Nařízení Evropského parlamentu a Rady (EU) č. 910/2014 ze dne 23. července 2014 o elektronické identifikaci a službách vytvářejících důvěru pro elektronické transakce na vnitřním trhu a o zrušení směrnice 1999/93/ES. In: Úřední věstník Evropské unie [online]. Kapitola I, článek 3, odst. 10 [cit. 2021-12-4]. Dostupné z: https://eur-lex.europa.eu/legal-content/CS/TXT/HTML/?uri=CELEX:32014R0910&from=cs#d1e787-73-1

[28] Nařízení Evropského parlamentu a Rady (EU) č. 910/2014 ze dne 23. července 2014 o elektronické identifikaci a službách vytvářejících důvěru pro elektronické transakce na vnitřním trhu a o zrušení směrnice 1999/93/ES. In: Úřední věstník Evropské unie [online]. Kapitola III, oddíl 4, článek 26 [cit. 2021-12-4]. Dostupné z: https://eur-lex.europa.eu/legal-content/CS/TXT/HTML/?uri=CELEX:32014R0910&from=cs#d1e787-73-1

[29] Nařízení Evropského parlamentu a Rady (EU) č. 910/2014 ze dne 23. července 2014 o elektronické identifikaci a službách vytvářejících důvěru pro elektronické transakce na vnitřním trhu a o zrušení směrnice 1999/93/ES. In: Úřední věstník Evropské unie [online]. Kapitola I, článek 3, odst. 12 [cit. 2021-12-4]. Dostupné z: https://eur-lex.europa.eu/legal-content/CS/TXT/HTML/?uri=CELEX:32014R0910&from=cs#d1e787-73-1

[30] Časové razítko. Earchivace.cz [online]. © 2014 [cit. 2021-12-04]. Dostupné z: http://www.earchivace.cz/technologie/casove-razitko/

[31] What Is Blockchan-Based Timestamping and Who Needs It? In: *Originstamp* [online]. © 2021 [cit. 2021-12-03]. Dostupné z: https://originstamp.com/blog/what-is-blockchain-based-timestamping/#a-step-by-step-guide-for-using-originstamp

[32] Certoo. Certoo.eu [online]. © 2021 [cit. 2021-12-03]. Dostupné z: https://certoo.eu/

[33] A timestamping proof standard. Opentimestamp.org [online]. [cit. 2021-12-04]. Dostupné z: https://opentimestamps.org/#

[34] GIPP, B., MEUSCHKE, N. a GERNANDT, A. Decentralized Trusted Timestamping using the Crypto Currency Bitcoin [online]. February 2015 [cit. 2021-12-05]. Dostupné z: https://www.researchgate.net/publication/272359313_Decentralized_Trusted_Timestamping_using_the_Crypto_Currency_Bitcoin

[35] PODHORSKÝ, Jiří. Steganografie. Brno 2010/2011. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií, Ústav počítačové grafiky a multimédií. [online]. [cit. 2021-12-05]. Dostupné z: https://www.vut.cz/www_base/zav_prace_soubor_verejne.php?file_id=117633

[36] SINGH, A. K., SINGH, J a SINGH, H. V. Steganography in Images Using LSB Technique. *Inflow: International Journal of Latest Trends in Engineering and Technology (IJLTET)* [online]. [cit. 2021-12-05]. Dostupné z: https://www.ijltet.org/wp-content/uploads/2015/02/60.pdf

[37] KOCIÁNOVÁ, Helena. Digitální steganografie. České Budějovice 2009. Diplomová práce. Jihočeská univerzita v Českých Budějovicích, Pedagogická fakulta, Katedra informatiky. [online]. [cit. 2021-12-05]. Dostupné z: https://theses.cz/id/5y5kip/downloadPraceContent_adipIdno_11124

[38] TAYAN, O., KABIR, M. N. a ALGINAHI Y. M. A Hybrid Digital-Signature and Zero-Watermarking Approach for Authentication and Protection of Sensitive Electronic Documents. *Inflow: The Scientific World Journal* [online]. 28 August 2014 [cit. 2021-12-05]. Dostupné z: https://downloads.hindawi.com/journals/tswj/2014/514652.pdf

[39] SAHOO, B. M., BEHERA, J. a ROUT, R. K. A Robust Fragile Watermarking Technique for Digital Image. *Inflow: International Journal of Latest Trends in Engineering and Technology (IJLTET)* [online]. [cit. 2021-12-05]. ISSN: 2278-0181. Dostupné online z: https://www.ijert.org/research/a-robust-fragile-watermarking-technique-for-digital-image-IJERTCONV3IS25024.pdf

[40] BALVÍNOVÁ, Alena. WYSIWYG. In: KTD: Česká terminologická databáze knihovnictví a informační vědy (TDKIV) [online]. Praha: Národní knihovna ČR, 2003- [cit. 2021-11-18]. Dostupné z: https://aleph.nkp.cz/F/?func=direct&doc_number=000000275&local_base=KTD

[41] RIVERBANK COMPUTING. What is PyQt? [online]. [cit. 2021-11-18]. Dostupné z: https://riverbankcomputing.com/software/pyqt/intro

[42] THE OPENSSL PROJECT AUTHORS. Welcome to OpenSSL! [online]. [cit. 2022-05-05]. Dostupné z: https://www.openssl.org/

[43] INTERNET ENGINEERING TASK FORCE. RFC 7292: *PKCS #12: Personal Information Exchange Syntax v1.1* [online]. Edited by: K. Moriarty, Ed., M. Nystrom, S. Parkinson, A.Rusch, M. Scott. July 2014 [cit. 2022-05-05]. Dostupné z: https://datatracker.ietf.org/doc/html/rfc7292

[44] PYINSTALLER. PyInstaller Manual [online]. [cit. 2021-11-28]. Dostupné z: https://pyinstaller.readthedocs.io/en/stable/#

[45] NULLSOFT SCRIPTABLE INSTALL SYSTEM. Main Page [online]. 12 August 2016, at 17:22 [cit. 2021-11-25]. Dostupné z: https://nsis.sourceforge.io/Main_Page

SEZNAM SYMBOLŮ A ZKRATEK

3DES	Triple Data Encryption Standard
AES	Advanced Encryption Standard
AMD	Advanced Micro Devices
ASCII	American Standard Code for Information Interchange
AVI	Audio Video Interleave
CC	Creative Commons
CD	Compact Disk
Czech POINT	Český Podací Ověřovací Informační Národní Terminál
DRM	Digital Rights Management
DSA	Digital Signature Algorithm
ECC	Elliptic Curve Cryptography
eIDAS	nařízení Evropského parlamentu a Rady (EU) č. 910/2014 ze dne 23. července 2014 o elektronické identifikaci a službách vytvářejících důvěru pro elektronické transakce na vnitřním trhu a o zrušení směrnice 1999/93/ES
FLAC	Free Lossless Audio Codec
GB	Gigabyte
GUI	Graphical User Interface
H.264	MPEG-4 Part 10
H.265	MPEG-H Part 2
HEVC	High Efficiency Video Coding

IDE	Integrated Development Environment
ISO	International Organization for Standardization
JFIF	JPEG File Intercharge Format
JPEG	Joint Photographic Experts Group
JSON	JavaScript Object Notation
LSB	Least Significant Bit
MKV	Matroska Video File
MP3	MPEG-2 Layer 3
MP4	MPEG-4 Part 14
MPEG-2	Moving Picture Experts Group Phase2
MPEG-4	Moving Picture Experts Group Phase 4
MS	Microsoft
NAS	Network Attached Strorage
NSIS	Nullsoft Scriptable Install System
NTP	Network Time Protocol
OS	Operating systém
PDF	Portable Document Format
PDF/A	Portable Document Format for Archive
PDF/E	Portable Document Format for Engineering

PDF/UA	Portable Document Format/ Universal Accessibility
PDF/X	Portable Document Format for Exchange
PFX	Personal Exchange Format
PKCS	Public Key Cryptographic Standards
PNG	Portable Network Graphics
RAM	Random Access Memory
RAR	Roshal Archive
RC2	Ron's Code or Rivest Cipher
RFC	Request for Comments
RSA	Rivest Shamir Adleman
RTF	Rich Text File
RTX	Ray Tracing Texel eXtreme
SHA	Secure Hash Algorithm
SQL	Structured Query Language
SSD	Solid-State Drive
SSL	Secure Sockets Layer
TIFF	Tag Image File Format
TLS	Transport Layer Security
TM	Trademark
TSA	Trusted Stamping Authority

TSQ	Time-Stamp Request
TSR	Time-Stamp Response
TXT	Plain Text File
UTC	Coordinated Universal Time
UTF	Unicode Text Format
WAV	Waveform audio format
WYSIWYG	What You See What You Get
XML	Extensible Markup Language
ZIP	Zipped File

SEZNAM PŘÍLOH

Příloha A - Obsah elektronické přílohy

V elektronické příloze jsou obsaženy dva adresáře, jeden textový soubor a několik souborů s kódy aplikace. V kořenovém adresáři se nachází složky *blockchainFiles* (obsahuje soubor s řetězcem bloků) a *timestampFiles* (obsahuje certifikáty potřebné k ověření časového razítka). Dále se zde nachází také textový dokument s názvem *Návod.txt* v němž je napsán návod na spuštění aplikace.

```
/ ..... kořenový adresář přílohy
├── blockchainFiles/ ..... adresář s řetězcem bloků
│       └── Blockchain.txt
├── timestampFiles/ ..... adresář s certifikáty
│       ├── cacert.pem
│       └── tsa.crt
├── Aplikace.pyw ..... hlavní soubor pro spuštění aplikace
├── Blockchain.py
├── BlockchainScreen.py
├── CertificateCreate.py
├── DigitalSignChoice.py
├── DigitalSignWithCert.py
├── DigitalSignWithCertFromPFX.py
├── EditorScreen.py
├── MainScreen.py
├── Návod.txt ..... soubor s návodem na spuštění aplikace
├── Steganography.py
├── TimestampChoice.py
├── TimestampCreate.py
├── TimestampInfo.py
├── TimestampVerify.py
└── WatermarkScreen.py
```