SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Evidenčné číslo: FEI-5382-91764

PASSWORD MANAGER

BAKALÁRSKA PRÁCA

2020 **Peter Čuřík**

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Evidenčné číslo: FEI-5382-91764

PASSWORD MANAGER
BAKALÁRSKA PRÁCA

Študijný program:	Aplikovaná informatika
Číslo študijného odboru:	2511
Názov študijného odboru:	9.2.9 Aplikovaná informatika
Školiace pracovisko:	Ústav informatiky a matematiky
Vedúci záverečnej práce:	prof. Ing. Pavol Zajac, PhD.
Konzultant:	unknown

Bratislava 2020 **Peter Čuřík**

SÚHRN

SLOVENSKÁ TECHNICKÁ UNIVERZITA V BRATISLAVE
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Študijný program:	Aplikovaná informatika
Autor:	Peter Čuřík
Bakalárska práca:	Password manager
Vedúci záverečnej práce:	prof. Ing. Pavol Zajac, PhD.
Konzultant:	unknown
Miesto a rok predloženia práce:	Bratislava 2020

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aenean et est a dui semper facilisis. Pellentesque placerat elit a nunc. Nullam tortor odio, rutrum quis, egestas ut, posuere sed, felis. Vestibulum placerat feugiat nisl. Suspendisse lacinia, odio non feugiat vestibulum, sem erat blandit metus, ac nonummy magna odio pharetra felis. Vivamus vehicula velit non metus faucibus auctor. Nam sed augue. Donec orci. Cras eget diam et dolor dapibus sollicitudin. In lacinia, tellus vitae laoreet ultrices, lectus ligula dictum dui, eget condimentum velit dui vitae ante. Nulla nonummy augue nec pede. Pellentesque ut nulla. Donec at libero. Pellentesque at nisl ac nisi fermentum viverra. Praesent odio. Phasellus tincidunt diam ut ipsum. Donec eget est. A skúška mäkčeňov a dĺžnov.

Kľúčové slová: tbd

ABSTRACT

SLOVAK UNIVERSITY OF TECHNOLOGY IN BRATISLAVA
FACULTY OF ELECTRICAL ENGINEERING AND INFORMATION TECHNOLOGY

Study Programme:	Applied Informatics
Author:	Peter Čuřík
Bachelor's thesis:	Password manager
Supervisor:	prof. Ing. Pavol Zajac, PhD.
Consultant:	unknown
Place and year of submission:	Bratislava 2020

On the other hand, we denounce with righteous indignation and dislike men who are so beguiled and demoralized by the charms of pleasure of the moment, so blinded by desire, that they cannot foresee the pain and trouble that are bound to ensue; and equal blame belongs to those who fail in their duty through weakness of will, which is the same as saying through shrinking from toil and pain. These cases are perfectly simple and easy to distinguish. In a free hour, when our power of choice is untrammelled and when nothing prevents our being able to do what we like best, every pleasure is to be welcomed and every pain avoided. But in certain circumstances and owing to the claims of duty or the obligations of business it will frequently occur that pleasures have to be repudiated and annoyances accepted. The wise man therefore always holds in these matters to this principle of selection: he rejects pleasures to secure other greater pleasures, or else he endures pains to avoid worse pains.

Keywords: tbd

Poďakovanie

Rád by som sa poďakoval vedúcemu tejto práce, prof. Ing. Pavlovi Zajacovi, Phd., za všetky rady, pripomienky a vedenie. Vážim si jeho trpezlivosť, každú ochotu navyše a taktiež jeho priateľský a úctivý prístup.

Obsah

Zoznam obrázkov a tabuliek

Zoznam algoritmov

Zoznam výpisov

Úvod

V dnešnej dobe je svet každým dňom stále väčšmi digitalizovaný. Neustále vznikajú nové produkty a služby. Systémy, ktoré nám pomáhajú mať všetko na jednom mieste. Bezpečnosť týchto systémov je rovnako dôležitá, ako jej funkčnosť. Keďže uchovávajú citlivé informácie používateľov, je absolútne kľúčové ich chrániť pred útokmi. Preto veľká väčšina aplikácií a systémov, ktoré pracujú s informáciami, používa účty. Používateľ si vytvorí svoj účet a dostane sa doň pomocou dvoch vstupov: používateľského mena a hesla. Heslo jeho účet chráni, keďže používateľské meno je verejné.

Na začiatku som uvádzal, že neustále vznikajú nové produkty, služby, či systémy. Môžeme teda očakávať, že bežný človek ich bude využívať viacero na dennej báze. Povedzme, že používa email, má účty v niekoľkých sociálnych sieťach, používa aplikáciu na elektronické bankovníctvo, je zaregistrovaný v niekoľkých internetových obchodoch, pravidelne pristupuje k svojim dátam na cloude (online úložisko) a podobne. Každá z týchto položiek pracuje s nejakým heslom, ktorá autentifikuje osobu, ktorá heslo zadala.

Ak hovoríme o hesle ako o reťazci, teda postupnosti znakov, používateľ si tento reťazec musí pamätať, aby mohol vstúpiť do systému. Tu vzniká problém. Problém pamätania si každého hesla pre každú aplikáciu. Existujú dve riešenia. Prvou možnosťou je nastavenie ľahko zapamätateľného, prípadne rovnakého hesla do všetkých účtov. Týmto sa dramaticky znižuje úroveň bezpečnosti. Zároveň sa ale zvyšuje level komfortu pri interakcii s aplikáciami.

Tou druhou možnosťou je použitie aplikácie typu password manager (správca hesiel). Môžeme o ňom uvažovať ako o trezore. Dovnútra môžeme uložiť všetky naše heslá a zamknúť ich pod jedným kľúčom. Situácia sa odrazu mení. Zrazu si nemusíme pamätať niekoľko hesiel, ale iba jedno. Úroveň komfortu pri interakcii s aplikáciami ostáva zachovaná, avšak rovnako je dosiahnutá vysoká úroveň bezpečnosti.

V praxi vytvára implementácia tohto managera určitý ekosystém. Teda, password manager dokáže poskytovať svoje služby len zariadeniu, na ktorom je nainštalovaný. Mimo tohto prostredia používateľ stráca znalosť o svojich údajoch. V súčasnosti považujeme oblasť ekosystému password managera za dostatočne rozvinutú. Preto sme sa rozhodli venovať oblasti mimo neho.

Cieľom práce je vyvinúť aplikáciu password manager a nájsť bezpečný, pohodlný a efektívny spôsob prístupu používateľa k heslám na cudzom zariadení. Teda na takom zariadení, kde password manager s citlivými údajmi používateľa nie je prítomný (mimo ekosystému). Snahou bude vymyslieť riešenie (riešenia), ktoré by toto umožňovali.

1 Základné pojmy

Pred samotným ením do problému, ktorým sa zaoberá táto práca je dôležité vysvetliť a definovať základné pojmy, ktoré sú spojené s danou problematikou a využívané v texte.

1.1 Heslo

Heslo je prostriedok, pomocou ktorého je overená totožnosť používateľa. [**1**] Pomocou neho vieme získať prístup k informáciam, dátam atď. ktoré sú pod ním uzamknuté. Teda iba ten, kto heslo pozná, môže pristupovať k týmto materiálom. Z tohto môžeme usúdiť, že heslo by malo byť dostatočne silné. Musí byť ťažko uhádnuteľné a komplexné. Jeho vlastník by ho mal ukryť pred odhalením alebo uhádnutím útočníka. Týmto nám vznikajú rôzne otázky: *Aké miesto je bezpečné na ukrytie hesla? Kedy môžeme prehlásiť, že heslo je „silné"?*

1.2 Autentizácia

Proces, pri ktorej je overená totožnosť osoby, sa nazýva autentizácia. Predchádza ju proces identifikácie, kedy sa osoba „predstaví" a povie, kto je. Systém ho ďalej v procese autentizácie „vyzve", aby dokázal, že dotyčná osoba je naozaj tou, za ktorú sa prehlásil. Tým dôkazom myslíme vyššie spomínané heslo. [**2**]

V praktickej rovine sú spôsoby autentizácie rôzne, ako napríklad: biometrický odtlačok, fráza vo forme hlasu, textového reťazca, číselný PIN a podobne [**3**].

1.3 Sila hesla

Uvažujme heslo ako textový reťazec. Sila hesla označuje stupeň obtiažnosti s akou ho neautorizovaná osoba dokáže uhádnuť [**4**]. Heslo môže byť silné alebo slabé, v závislosti od toho, ako ťažké ho je uhádnuť [**4**]. Slabé heslo je napríklad používanie iba malých písmen alebo iba číslic. Dôvod, prečo to tak je, je príliš malý priestor výberu znaku. Pri číselnom hesle hovoríme o priestore desiatich znakov. Uvažujme štandardnú telegrafnú abecedu s 26 písmenami. Potom je priestor pri použití hesla iba z malých písmen veľký 26 znakov. Útočník môže predpokladať, že používateľ má heslo zložené iba z číslic alebo iba z malých písmen[1].

Preto na druhej strane hovoríme, že silné heslo je také heslo, ktoré obsahuje kombináciu veľkých a malých písmen a číslic. Už len kombináciou veľkých a malých písmen sa nám priestor zdvojnásobí. Abeceda veľkých písmen aj malých písmen má 26 znakov, čo je spolu

[1]Možnosť použitia hesla iba z veľkých písmen nespomíname, pretože z matematického hľadiska náročnosti prelomenia hesla ide o rovnaký prípad ako pri malých písmenách.

52 znakov. Zrazu je pre útočníka pri každom písmene nutné uvažovať, či sa použilo ako veľké, alebo ako malé. Z matematického hľadiska, teda z hľadiska permutácií sa celkový počet možných usporiadaní exponenciálne zvýši. Permutácia znamená usporiadanie.

Tabuľka 1: Zložitosť prelomenia hesiel pomocou útoku brute-force podľa webovej stránky grc.com/haystack.htm

Typ Hesla	**Heslo**	**Priestor**	**Počet možností**
Číslice (ďalej len C)	01234	10	$1,11 * 10^5$
Malé písmená (ďalej MP)	heslo	26	$1,24 * 10^7$
MP + veľké písmená (VP)	hEsLo	52	$3,88 * 10^8$
MP + VP + C	h3sL0	62	$9,31 * 10^8$
MP + VP + C	h3sL0jeSiLn3	62	$3,28 * 10^{21}$
MP + VP + C + špec. znaky	h3sL0=%SiLn3	95	$5,46 * 10^{23}$

Heslo	**Čas (online útok) pri 1000 pokusoch/s**	**Čas (offline útok) pri miliarde pokusoch/s**
01234	1,85 min	0,00000111 s
heslo	3,43 hod	0,000124 s
hEsLo	4,49 dní	0,00388 s
h3sL0	1,54 týždňov	0,00931 s
h3sl0jeSiLn3	104 miliárd rokov	1043 rokov
h3sLo=%SiLn3	17,4 biliónov rokov	1740 rokov

Z tabuľky sme pozorovaním zistili, že rovnako ako bohatý priestor znakov je dôležitá aj dĺžka hesla. S použitím veľkých aj malých písmen pri dĺžke hesla 5 by bol útočník schopný zistiť naše heslo za veľmi krátky čas. Môžeme si z časových výsledkov všimnúť, že z praktického hľadiska skoro ani nezáleží, či použijeme C, MP, MP + VP alebo MP + VP + C, pokiaľ je heslo krátke. Najmä pri offline útoku zjavne vidieť, že vo všetkých prípadoch by stroj uhádol heslo doslova do sekundy.

Silu exponenciálneho rastu si všímame pri zmene dĺžky hesla na 12 znakov. Celá situácia sa dramaticky zmenila a kombinácie C, MP a VP už dávajú zmysel. Ďalší veľký skok spôsobilo pridanie špeciálnych znakov. Keďže zväčšili priestor rôznych znakov o viac ako polovicu, významná zmena je vidieť aj vo výsledkoch.

Predpokladajme, že útočník má informáciu, že používateľ vlastní heslo zložené iba z MP. Potom platí, že ak by používateľ zväčšil dĺžku hesla o jeden znak, útočník musí

vykonať v priemere o 26 viac pokusov pri každej permutácii. [**5**]

Ďalej predpokladajme, že používateľ vlastní heslo zložené z MP, VP a C. Takáto kombinácia je dnes pri registrácii vo veľkej miere povinnosťou na rôznych webových stránkach. Potom platí, že pri zväčšení dĺžky hesla o jeden znak by sa zložitosť hesla nezvýšila iba 26, ale až 62-násobne. Z toho vyplýva, že útočník by musel mať 62-násobne väčší výkon, aby mohol za rovnaký čas zlomiť heslo z pôvodnou dĺžkou. Ten sa zvyšuje každé dva roky dvojnásobne, podľa Moorovho zákona. [**5**]

Táto úvaha spolu s ďalšími typmi útokov a ochranou pred nimi je hlbšie obsiahnutá v práci [**5**].

1.4 Password manager

Password manager (alebo po sl. správca hesiel) je aplikácia, ktorá umožňuje vytváranie, uchovávanie a používanie rôznych hesiel [**6**]. Zhromažďuje ich na jednom mieste a vytvára pre používateľa prehľad jeho prístupových údajov. Tento „trezor" je chránený master heslom.

Master heslo (často sa s ním stretnete v angl. forme „master password") je heslo, ktoré je použité na sprístupnenie iných hesiel [**7**]. Táto skutočnosť nám odhalila jednu pozitívnu a jednu negatívnu stranu password managerov všeobecne. Pozítivom je, že namiesto n hesiel si stačí pamätať práve jedno heslo - master heslo. Týmto bolo rovno zodpovedané aj negatívum. Ak sa vieme jedným heslom dostať ku všetkým ostatným, stačí zlomiť master heslo a útočník získa všetky údaje v trezore, teda v password manageri.

1.5 Šifrovanie

Nasledujúci text vychádza zo zdroja [**8**].

Šiforvanie je prepis otvoreného (čitateľného) textu do zašifrovaného textu, ktorý nazývame šifra. Abeceda, z ktorého vychádza otvorený text budeme označovať ako $\mathcal{P}$ a abecedu, z ktorého bude vychádzať zašifrovaný text budeme označovať ako $\mathcal{C}$.

$$e_k = \mathcal{P} \rightarrow \mathcal{C} \tag{1}$$

Vidíme, že ide o zobrazenie. Toto zobrazenie je bijektívne, no nie vždy je tomu tak (napríklad pri znáhodnených šifrách). Je závislé na tajnom parametri k, ktorý nazývame kľúč. Ten patrí do množiny kľúčov K. V dnešných, moderných šifrách skoro vždy platí, že $\mathcal{P} \neq \mathcal{C}$, avšak v klasických šifrách bol opak úplne bežným úkazom (transpozičné, substitučné, homofónne šifry a podobne).

Aby vedel príjemca šifru prečítať, musí byť spomínané zobrazenie invertovateľné. To znamená, že musí byť použité inverzné zobrazenie

$$d_k = \mathcal{C} \rightarrow \mathcal{P} \tag{2}$$

také, aby platilo: $d_k(e_k(x)) = x$, kde x je nezašifrovaná správa. Z týchto vzťahov je zrejmé, že príjemca musí použiť totožný kľúč k s kľúčom odosielateľa, aby sa dostal k x. Útočník sa snaží nájsť tento kľúč. Preto čím väčšia je množina K, tým náročnejšie, niekedy až nemožné z hľadiska výpočtovej sily, je nájsť k.

V dobe klasických šifier (obdobie do roku 1945) väčšinou spočívala bezpečnosť niektorých algoritmov v ukrytí algoritmu samotného. To ale nie je správny prístup, pretože podľa Kerckhoffovho princípu [**9**] má bezpečnosť šifrovacieho algoritmu spočívať na utajení kľúča, nie algoritmu samotného. Týmto princípom sa riadia dnešné, moderné šifry dodnes.

2 Problém password managerov

Password manager ponúka používateľovi mnohé výhody a možnosti. Od generátora hesiel a ich automatického vyplnenia pri prihlasovaní do stránok, cez synchronizáciu medzi zariadeniami, zálohovanie, až po automatickú, pravidelnú zmenu jednotlivých hesiel. Napriek tomu existuje problém týchto managerov, ktorý ostal neriešený. Jemu sa bude táto kapitola venovať.

2.1 Súčasný stav na trhu

Ak hovoríme o heslách, priame (explicitné) používanie nižšie uvedených aplikácií pri dennom používaní je minimálne. A to vďaka automatickému vypĺňaniu hesla. Používateľ vstúpi na webovú stránku alebo do aplikácie. Pred vstupom sa musí prihlásiť do svojho účtu. Tam mu password manager výzvou ponúkne automatické vyplnenie (angl. výraz „autofill") prihlasovacieho mena a hesla. Toto poskytuje vysokú úroveň komfortu. Používateľ nielen že nemusí vypisovať svoje prihlasovacie údaje manuálne, ale aj ich bezpečnosť porástla na vyššiu úroveň. Vďaka password manageru.

Podobne to funguje pri registrácii nového konta. Niektoré password manager aplikácie ponúknu náhodne vygenerované silné heslo (iCloud Keychain, [**10**]). Používateľ sa môže rozhodnúť ho prijať. V takom prípade sa heslo pre vytvorený účet uloží do password manageru a ten ho pri každom ďalšom prihlásení vyplní.

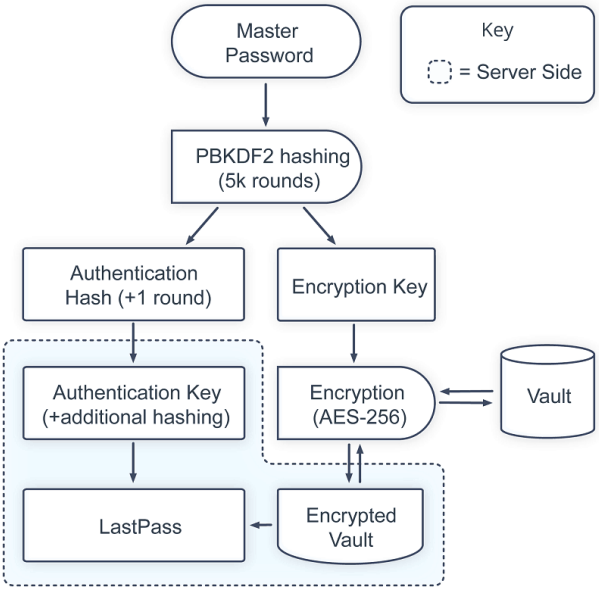
2.1.1 LastPass

LastPass patrí medzi jeden z najpopulárnejších, čo sa týka počtu používateľov [**11**]. Všetky heslá a ostatný obsah sú uzamknuté pod jedným master heslom. LastPass ale povoľuje aj vstup do jeho aplikácie pomocou biometrickej autentifikácie, ktorú väčšina smartfónov podporuje. Používateľ tak nemusí každý raz písať dlhé master heslo. Jediné, čo stačí, je priložiť prst, či pozrieť sa na smartfón (sken tváre).

LastPass je client-server aplikácia. To znamená, že niektoré operácie a procesy prebiehajú na strane kienta, teda priamo v danom zariadení, ktoré používateľ drží a niektoré prebiehajú na strane LastPass serverov. Samotné šifrovanie, aj dešifrovanie prebieha na strane zariadenia [**12**]. LastPass vytvorí kľúč z master hesla pomocou šifry AES (Advanced Encryption Standard) s hašovaním PBKDF2 (Key Derivation Function), SHA (Secure Hash Algorithm) s pridaným saltom [**13**]. Týmto algoritmom sa bližšie venujeme v X.X (TODO číslo sekcie kde budem vysvetľovať AES princíp a hašovanie).

Dáta sú synchronizované pomocou serverov, čo umožňuje zálohovanie a synchronizáciu medzi zariadeniami. Dátový prenos po sieti je chránený pomocou TLS/SSL ([**14**] sa

bližšie venuje tomuto protokolu). LastPass v bezpečnosti pokračuje v možnosti dvojfaktorovej autorizácie a overovania na základe lokácie: kedykoľvek sa užívateľ prihlasuje do aplikácie z inej lokality, je vyzvaný prostredníctvom emailu s linkom, ktorý po otvorení overí používateľa ako verifikovaného. LastPass má mnoho možností, ako napríklad zdieľanie hesiel s iným LastPass účtom, generovanie hesiel s používateľom zvolenou dĺžkou a podmienkami, hodnotenie sily hesiel (systém usúdi, či je heslo dostatočne bezpečné), či import hesiel pomocou CSV súboru alebo iného password manageru.



Obr. 1: Schéma šifrovania aplikácie LastPass.

2.1.2 Dashlane

Tento password manager je tiež vysoko využívaný [**15**]. Jedna z jeho jedinečných funkcionalít je automatická zmena hesla [**16**]. Používateľ si môže vybrať v zozname svojich hesiel, ktoré by sa mali automaticky meniť. Dashlane tak bude pravidelne generovať nové, komplexné heslá pre vybrané položky. Keďže medzi rôznymi webovými stránkami nie je jednotná architektúra a dizajn, nie všetky položky účtov vie Dashlane meniť automaticky. V takom prípade si užívateľ vie meniť heslo pre danú položku iba manuálne, v konkrétnej aplikácii alebo webovej stránke pre službu, ku ktorej prislúcha daný účet v Dashlane.

Tak ako LastPass, aj tento password manager dominuje silnou bezpečnosťou. Podporuje dvojfaktorovú autentifikáciu (dodatočné overenie po prvotnom prihlásení, viac v [**17**]), šifru AES a synchronizáciu medzi zariadeniami. Mnohé z funkcionalít a možností má spoločné s aplikáciou LastPass. Nebudeme ich opäť spomínať, nakoľko cieľom tejto

kapitoly je iba ukázať už existujúce vymoženosti rôznych password managerov.

2.1.3 iCloud Keychain

Za spomenutie určite stojí vstavaný password manager od spoločnosti Apple. Obsahuje určité funkcionality password managera, ktoré sú základnou súčasťou každého iOS, iPadOS, či MacOS zariadenia. Každý používateľ nejakého Apple zariadenia je identifikovaný pomocou AppleID konta. K nemu má pridelené cloud úložisko, kde sa mu všetky dáta zálohujú a synchronizujú so všetkými Apple zariadeniami.

Okrem fotiek, poznámok, kontaktov a iných dát tam sú uložené aj heslá z Keychainu. Keychain si pamätá nielen heslá, ale aj certifikáty, dôležité pri rôznych verifikáciach v rámci systému. Taktiež si pamätá kľúče. Sú verejné, ale aj súkromné, systém ich používa napríklad pri používaní iMessage (četovacia aplikácia). Sú šifrované pomocou šifry RSA a dĺžka kľúča je niekedy až 2048 bitov.

Z ostatných managerov ukážeme zopár ďalších funkcionalít, ktoré neboli spomenuté, respektíve ich LastPass, Dashlane, či Keychain neposkytujú.

1Password ponúka takzvaný „Emergancy kit". Po vytvorení 1Password účtu je používateľovi poskytnutý dokument. Následne je vyzvaný, aby si ho vytlačil, prípadne uložil na pamäťové médium. Dokument obsahuje prihlasovacie údaje a master heslo. Taktiež obsahuje QR kód, ktorý automaticky vyplní tieto dáta pri núdzovom prihlasovaní.

Obr. 2: Emergency kit od 1Password - ukážka pdf súboru, ktorý používateľ obdrží.

Password manager Remembear ponúka jednoduchú aktiváciu aplikácie na novom zariadení prostredníctvom QR kódu. Za spomenutie stojí aj menej známy správca, konkrétne Myki.

Myki ako jeden z mála nepoužíva svoje servery na zálohovanie a ukladanie hesiel. Namiesto toho využíva server iba ako sprostredkovateľa [**18**] spojenia keď nastáva synchronizácia medzi zariadeniami. Teda, používať Myki na viacerých zariadeniach je akousi formou „zálohy".

Vyššie spomenuté aplikácie fungujú na rôznych platformách: verzia pre smartfóny (iOS, Android), počítače (Windows, MacOS, Linux), inteligentné hodinky a podobne.

2.2 Nízka popularita password managerov a nesprávne alternatívne spôsoby ukladania hesiel

Ľahko sme vedeli ukázať, že bohatosť a rôznorodosť trhu s aplikáciami na správu hesiel je naozaj veľká. Používateľ by teda nemal mať problém vybrať si takú aplikáciu, ktorá vyhovuje jeho požiadavkám. Napriek tomu je popularita password managerov nízka.

Už nadpis článku [**19**] začína slovami *„Hardly Anybody Uses a Password Manager"*. Teda v slovenskom znení - „Horko-ťažko niekto vôbec používa password manager". Píše o prieskume na túto tému a jeho výsledkoch. V Spojených štátoch a v Anglicku sa pýtali 1000 respondentov rôzne otázky. Týkali sa bezpečnosti pri používaní internetu. Išlo najmä o praktiky pri používaní hesiel; či pre každú stránku používajú iné heslo, či je heslo silné, dlhé alebo krátke, „náhodné" alebo ľahko zapamätateľné. Spomedzi opýtaných, 59% uviedlo, že používa 5 alebo menej rôznych hesiel, ktoré si pamätajú v hlave. Pritom 74% sa denne prihlási do 6 a viac účtov. Môžeme už teraz hovoriť o pravdepodobnosti faktu, že niektorí z týchto používateľov používajú rovnaké heslo pre rôzne účty.

Ďalšia negatívna skutočnosť, ktorá vyplýva z prieskumu je písanie hesiel na papier. Až 42% opýtaných používa túto metódu. Niekto by argumentoval, že rovnako ako papier s heslami vie niekto ukradnúť aj server, kde sú uložené heslá password manageru. Aj keď sa môže zdať, že princíp je ten istý, na rozdiel od papiera s ručne vypísanými heslami sú heslá na serveroch password managerov zašifrované. Dnešné moderné šifry, ktoré sa na ich zašifrovanie používajú sú matematicky silné a v súčasnosti neprelomiteľné. Tá ťažšia časť je ukryť kľúč tak, aby ho nikto nenašiel. Dôvodom úspešných útokov teda nie je nedokonalosť moderných šifier. Väčšinou ide o postranné kanály, zlé ukrytie kľúča, spiknutie zvnútra a podobne.

Keď máme hovoriť o popularite password managerov, tento prieskum silno podporuje nadpis tejto podkapitoly. Len 8% opýtaných používa na svoje heslá nejakého správcu.

Naopak, jemne menej než tri štvrtiny si vystačí s tým, že si ich heslá zapamätá prehliadač. Tu treba uviesť skutočnosť, že uklasdanie hesiel do prehliadača je náchylné na krádež pomocou malvéru (škodlivý softvér, ktorý sa snaží infikovať zariadenie, väčšinou využívaný na krádež citlivých údajov, najmä hesiel [**20**]).

Populárny prehliadač Google Chrome ponúka používateľovi zapamätať si heslá pri prihlasovaní na rôznych stránkach. Tieto heslá sú synchronizované do všetkých zariadení pomocou Google konta. Je ľahké sa dostať do tohto trezoru, stačí do URL text-boxu napísať „chrome://settings/passwords" [**21**]. Zobrazí sa zoznam hesiel pre dané stránky, ktoré si prehliadač ukladal. Jedným tlačidlom sa odokryje heslo ako otvorený text. Chrome nevyžaduje žiadnu autentizáciu. Je jednoduché pre útočníka pri fyzickom prístupe k zariadeniu tieto dáta získať pár klikmi. Chrome vývojár Justin Schuh avšak argumentuje [**21**]. Hovorí, že keď má niekto prístup do účtu operačného systému používateľa, vie, vidí a má prístup ku všetkému. Môžeme z tohto tvrdenia vyvodiť záver, že Chrome nevidí zmysel v chránení trezoru prehliadača (okrem hesla do Googlu účtu), lebo rozumie faktu, že ten, kto má prístup do operačného systému, má aj tak prístup ku všetkému.

Mozilla Firefox tiež nechráni svoj trezor hesiel, avšak narozdiel od Chromu ponúka aktiváciu master hesla. Spomeňme ešte Safari od Apple, ktorý chráni celý trezor heslom účtu operačného systému. Z výroku Schuha vyplýva, že takéto zabezpečenie je zbytočné. Apple pravdepodobne počíta s tým, že môže prísť k zneužitiu zariadenia už po prihlásení do systému, teda útočník nemusí poznať heslo. V takom prípade považujeme takéto zabezpečenie za múdre, avšak určite existujú bezpečnejšie spôsoby (za cenu komfortu).

Spomínaný prieskum z roku 2015 nepriniesol pozitívne výsledky. O tri roky neskôr sa uskutočnil ďalší [**22**]. Spomedzi 2500 Američanov si 35% nikdy heslá nemení. Robí tak iba po vyzvaní. Veľkým prekvapením bolo 11% používateľov, ktorí si ich menia každý deň. The National Institute of Standards and Technology tvrdí, aby si používatelia menili heslá nie pravidelne, ale až keď boli prelomené. Keď padla otázka, aký nástroj používajú na svoju ochranu na internete, víťazom bol antivírový software (53%), password manager získal 24%, čo je trojnásobný nárast za obdobie troch rokov[2].

Napriek pozitívnemu nárastu popularity password managerov považujeme 24% za malé číslo. Je pravdou, že tieto aplikácie sú na trhu nie tak dlho. Antivírový softvér má v tomto náskok. Tento softvér má avšak iný zámer a zameranie v bezpečnosti, než password manager. U niektorých čitateľov môže vznikať otázka typu: *Prečo používatelia nepoužívajú viac bezpečnostné nástroje na ochranu ich osobných údajov?* V druhom spomínanom pries-

[2]vychádzajúc zo vzorky opýtaných, teda určitá štatistická odchylka je pri týhto úvahách samozrejmosťou.

kume z roku 2018 sa opýtaných pýtali aj na otázku kedy boli poučení, respektíve vzdelaní na tému bezpečnosti na internete. Až 36% nedostalo na túto tému žiadne vzdelanie. Aj toto môže byť odpoveďou na vyššie spomínanú otázku.

Ďalším dôvodom môže byť nedôvera. Používatelia nemusia byť presvedčení o tom, že password manager naozaj ich heslá ochráni, nezverejní, prípadne nezneužije. Jedna práca [**23**] študovala, prečo je stále málo používateľov password manageru. Urobila prieskum, kde prizvala 137 respondentov používajúcich password manager a 111 takých, ktorí ho nepoužívajú. Vznikla štúdia, ktorá porovnáva odpovede na 6 otázok týchto dvoch skupín a snaží sa vyvodiť záver, prečo je popularita týchto aplikácií taká, aká je.

3 Recitácia

Citujem všetky zdroje v **bibliography.bib**, [**t00**, **t01**, **t02**, **t03**, **kniha**, **kniha2**, **kniha3**, **small**, **big**, **cs**, **koll**, **kap**, **tug**, **knuth**, **zbornik**, **prispevok**].
Good luck.

4 Možnosti anonymizácie

Anonymizácia znamená zmena alebo úprava údajov tak, aby sa podľa nich nedala jednoznačne určiť osoba, ktorej tieto údaje patria [**t01**]. Existuje niekoľko spôsobov, ktorými môžeme dosiahnuť rôznu úroveň anonymizácie na internete: od mazania cookies súborov po ukončení prehliadania webových stránok až po používanie operačných systémov, ktoré sú na anonymite založené; od bezplatných možností až po komerčné verzie.
Nasleduje priblíženie niektorých možnosti anonymizácie.

4.1 Súkromné prehliadanie

Najpoužívanejšie internetové prehliadače súčasnosti majú v sebe zabudovanú funkcionalitu, ktorá dokáže čiastočne anonymizovať prístup na internet. Táto funkcionalita blokuje ukladanie navštívených stránok do histórie a nezaznamenáva súbory, ktoré sa stiahnu z internetu. SW a Halo Wars sú skratky.

4.2 Anonymná sieť

Anonymná sieť je sieť serverov, medzi ktorými dáta prechádzajú šifrované. V anonymných sieťach dáta prechádzajú z počítača používateľa, odkiaľ bola požiadavka poslaná, cez viaceré proxy smerovače, z ktorých každý správu doplní o smerovanie a zašifruje vlastným kľúčom. Cesta od ...

4.3 Funkcionalita

Rozšírenie tiež okrem splnenia špecifikácie malo pre prehľadnosť a overenie funkčnosti zobrazovať údaje, ktoré boli na server odoslané. Zoznam údajov odoslaných na server, sa mal ukladať do krátkodobej histórie, aby nemal používateľ k dispozícií len najnovšie údaje, ale aj údaje odoslané v nejakom časovom období. Nejaky listing z priloh C.1.

4.3.1 Funkcionalita2

Samozrejmosťou bolo nastavenie zapnutia rozšírenia pri štarte, prípadne interval zmeny odosielaných údajov.

4.4 Vzhľad

Dôležitou požiadavkou kladenou na rozšírenie bolo príjemné používateľské rozhranie. Z tohto dôvodu malo rozšírenie obsahovať zoznam modifikovaných vlastností a tlačidlo pre prístup k nastaveniam rozšírenia v jednoduchej a praktickej forme. Predpokladaný vzhľad je zobrazený na obrázku č. 3. Dôležitou požiadavkou kladenou na rozšírenie bolo príjemné používateľské rozhranie.[**t00**] Z tohto dôvodu malo rozšírenie obsahovať zoznam

Tabuľka 2: Moduly a ich funkcie pri anonymizácii

	Funkcia													
						Modifikácia								
Modul	zobrazenie hlavičky	blokovanie skriptov	zmena IP	zmena lokalizácie	zmazanie/blokovanie cookies	blokovanie trackerov	popis	používateľský agent	kódové označenie prehliadača	názov prehliadača	verzia prehliadača	platforma	výrobca prehliadača	označenie výrobcu prehliadača
User agent switcher							X	X	X	X	X	X	X	X
Ghostery					X	X								
Better privacy					X									
Anonymox			X	X	X		X	X						
Modify headers					X			X						
Request policy						X								
Live HTTP headers	X													
User agent awitcher for chrome							X	X						
Header hacker							X	X	X	X	X	X	X	X
Mod header							X	X	X	X	X	X	X	X
Script no		X												
No script		X												
Proxify it			X	X										
I'm not here				X										
Get anonymous personal edition		X	X	X	X	X								
Anonymous browsing toolbar			X	X										
Easy hide your IP and surf anonymously			X	X				X	X	X	X			

Obr. 3: Predpokladaný vzhľad rozšírenia.

modifikovaných vlastností a tlačidlo pre prístup k nastaveniam rozšírenia v jednoduchej a praktickej forme. Predpokladaný vzhľad je zobrazený na obrázku č. 3.

```
/* Hello World program */

#include<stdio.h>

struct cpu_info {
    long unsigned utime, ntime, stime, itime;
    long unsigned iowtime, irqtime, sirqtime;
};

main()
{
    printf("Hello World");
}
```

Listing 1: Ukážka algoritmu

Algorithm 1 Ukážka príkazov pre algorithmic

```
  <text>
  if <condition> then
    <text>
  else
    <text>
  end if
  if <condition> then
    <text>
  else if <condition> then
    <text>
  end if
  for <condition> do
    <text>
  end for
  for <condition> to <condition> do
    <text>
  end for
  for all <condition> do
    <text>
  end for
  while <condition> do
    <text>
  end while
  repeat
    <text>
  until <condition>
  loop
    <text>
  end loop
Require: <text>
Ensure: <text>
  return <text>
  print <text> {<text>} and , or , xor , not , to , true, false
```

Záver

Conclusion is going to be where?
Here.

includes/bibliography.bib

Prílohy

A Štruktúra elektronického nosiča

/CHANGELOG.md

· file describing changes made to FEIstyle

/example.tex

· main example *.tex* file for diploma thesis

/example_paper.tex

· example *.tex* file for seminar paper

/Makefile

· simply Makefile – build system

/fei.sublime-project

· is project file with build in Build System for Sublime Text 3

/img

· folder with images

/includes

· files with content

/bibliography.bib

· bibliography file

/attachmentA.tex

· this very file

B Algoritmus

Algorithm B.1 Vypočítaj $y = x^n$

Require: $n \geq 0 \vee x \neq 0$

Ensure: $y = x^n$

```
y ⇐ 1
if n < 0 then
    X ⇐ 1/x
    N ⇐ −n
else
    X ⇐ x
    N ⇐ n
end if
while N ≠ 0 do
    if N is even then
        X ⇐ X × X
        N ⇐ N/2
    else {N is odd}
        y ⇐ y × X
        N ⇐ N − 1
    end if
end while
```

C Výpis sublime

```
{
    "folders":
    [
        {
            "path": ".",
            "folder_exclude_patterns": [".build", ".aux"],
            "follow_symlinks": true
        }
    ],
    "settings" : {
        "TEXroot": "example.tex",
        "tex_file_exts": [".tex"],
        "use_biblatex": true,
        "glossary_auto_trigger": true,
        "aux_directory": "./.aux",
        "output_directory": "./.build",
        "builder_settings": {
            "program": "pdflatex",
            "options": "--shell-escape"
        }
    },
    "build_systems":
    [
        {
            "name": "FEI - LaTeX",
            "working_dir": "${folder}",
            "shell_cmd": "make",
            "variants": [
                {
                    "name": "clean",
                    "shell_cmd": "make clean",
                }
            ]
        }
    ]
}
```

Listing C.1: Ukážka sublime-project