Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií

Bakalářská práce

Porovnání kryptografických metod

Filipp Podriadchikov

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

Filipp Podriadchikov

Informatika

Název práce

Porování kryptografických metod

Název anglicky

Comparison of crypthographical methods

Cíle práce

Hlavním cílem práce je porovnat dostupné kryptografické metody pro oblast ochrany souborů doporučit vhodnou metodu, nebo skupinu metod v závislosti na jejich výkonnostích charakteristikách. Vedlejším cílem práce je vytvořit přehled aktuálně používaných kryptografických metod.

Metodika

Při studiu odborných pramenů bude využita metoda analýzy a syntézy. Pro výběr vhodné metody bude použita metoda párového porovnávání. Měření výkonosti bude realizováno testováním šifrovací funkce, rychlosti šifrování, délky klíče, rycholsti generování klíčů, a typy klíčů. Na základě zjištěných dat budou konkretizována doporučení pro výběr kryptografické metody vhodné pro ochranu souborů.

Doporučený rozsah práce

40

Klíčová slova

Cryptography, Information security, Algorithms , IT standards

---

Doporučené zdroje informací

’ Bezpečnost informací ’ Krysin, SPARRK Kyjev: 2003; ISBN: 5-7315-0144-0.

DOSEDĚL, Tomáš. Počítačová bezpečnost a ochrana dat. 1. vydání. Brno: Computer Press, 2004. s.190. ISBN 80-251-0106-1

DOSEDĚL, Tomáš. 21 základních pravidel počítačové bezpečnosti. 1. vydání. Brno: CP Books, 2005. 52s. ISBN 80-251-0574-1

MLÝNEK, Jaroslav. Zabezpečení obchodních informací. 1. vydání. Brno: Computer Press, 2007. s.154.ISBN 978-80-251-1511-4

PECINOVSKÝ, Josef. Metody a možnosti zálohování počítačových dat. 1. vydání. Praha: Grada publishing, 2003. 116 s. ISBN 80-247-0659-8.

Tarasyk M. V. Chráněné informační technologie. Projektování a aplikace. – ”SOLÓN-Press, 2004. – 192 s. ISBN: 5-98003-143 ..

Předběžný termín obhajoby

2017/18 LS – PEF

Vedoucí práce

Ing. Tomáš Vokoun

Garantující pracoviště

Katedra informačních technologií

Elektronicky schváleno dne 20. 2. 2018

Ing. Jiří Vaněk, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 21. 2. 2018

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 15. 03. 2018

Čestné prohlášení

Prohlašuji, že svou bakalářskou práci "Porovnání kryptografických metod" jsem vypracoval samostatně pod vedením vedoucího bakalářské práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené bakalářské práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne 15.3.2018 _____

Poděkování

Rád bych touto cestou poděkoval panu Ing. Tomáši Vokounovi za cenné rady, ochotu a vedení práce.

Porovnání kryptografických metod

Souhrn

Bakalářská práce se podrobněji věnuje kryptografii, konkrétné se zabývá popisem metody ochrany dat. Nejčastěji používané metody kryptografie jsou následně podrobněji popsány. V této částí také najdeme shrnutí všech jejich výhod a nevýhod. Po podrobnější analýze jednotlivých metod, následuje výběr nejvhodnější metody pomocí párového porovnání. Na závěr jsou shrnuty, výsledky porovnání a předložena nejvhodnější metoda pro bezpečnost šifrovaných souborů.

Klíčová slova: symetrická kryptografie, asymetrická kryptografie, algoritmy šifrování, šifrování dat.

Comparison of crypthographical methods

Summary

Bachelor's thesis is discussed in more detail in cryptography, particular deals with the description of the method of data protection. the Most commonly used methods of cryptography are subsequently described in more detail. In this part we also find a summary of all their advantages and disadvantages. After a more detailed analysis of each method, followed by selection of the most appropriate methods using pairwise comparison. In conclusion are summarized the results of the comparison and submitted to the most appropriate method for the security of the encrypted files.

Keywords: symmetric cryptography, asymmetric cryptography, encryption algorithm, data encryption.

Obsah

Seznam obrázků

1 Úvod

Široké využití počítačové techniky a neustálé zvyšování objemu informačních toků způsobuje neustálý nárůst zájmu o kryptografii. V poslední době roste potřeba na ochranu informaci za pomoci využití algoritmů. Nástroje využívajicí algoritmy jsou jednoduše modernizovány a nevyžadují velké finanční náklady ve srovnání s hardwarovými kryptosystémy. Moderní šifrovací metody zaručují téměř absolutní ochranu dat, ale stále existuje problém spolehlivosti jejich implementace.

V současné době došlo k tomu, že zhodnocení již používaných kryptografických algoritmů se stalo aktualní praktikou. Samostaný vývoj úloh zabývající se učinností ochranných prostředků je často jednodušší než, samostatná práce s nimi. A to převážn vyžaduje přítomnost specializovaných znalostí a zpravidla vyšší kvalifikace než úloha samotného rozvoje. Tyto okolnosti vedou k tomu, že na trhu se objevuje spousta prostředků kryptografické ochrany informací, o nichž nikdo nemůže říct nic určitého. Současně vývojáři uchovávají kryptoalgoritmy jak praxe ukazuje, často nestabilní) v tájemství. Nicméně úloha přesného určení kryptografického algoritmu nemůže být příliš složitou, a to proto, že každý vývojář jí musí znát. Kromě toho, pokud by porušovatel nalezl způsob, jak tuto ochranu překonat, tak není v jeho zájmu tuto zjištění zveřejňovat. Z toho důvodu by společnost mohla velmí benefitovat z otevřčené diskuze o bezpečností system, což by podle mého názoru mělo být kompletně nepřijatelné.

2 Cíl práce a metodika

2.1 Cíl práce

Hlavním cílem práce je porovnat dostupné kryptografické metody pro oblast ochrany souborů doporučit vhodnou metodu, nebo skupinu metod v závislosti na jejich výkonnostích charakteristikách. Vedlejším cílem práce je vytvořit přehled aktuálně používaných kryptografických metod.

2.2 Metodika

Při studiu odborných pramenů bude využita metoda analýzy a syntézy. Pro výběr vhodné metody bude použita metoda párového porovnání. Měření výkonnosti bude realizováno testováním šifrovací funkce, rychlosti šifrování, délky klíče, rycholsti generování klíčů, a typy klíčů. Na základě zjištěných dat budou konkretizována doporučení pro výběr kryptografické metody vhodné pro ochranu souborů.

3 Teoretická východiska

3.1. Problematika šifrování

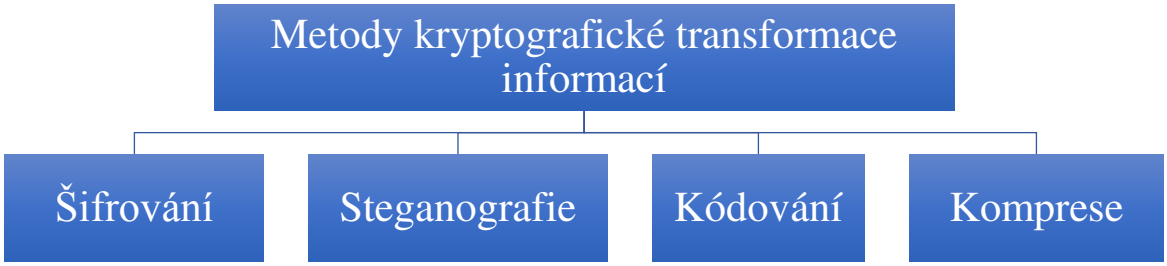
Zabezpečení úniku informací patří v dnešní době mezi jedno z nejvíce diskutovaných témat z oblasti informatiky. Co se týče oblasti zabezpečení, řeší se převážnč dva případy Zaprvé snaha zabránit útočníkovi, aby se k datům vůbec dosta. Zadruhé, v případě že se útočník k datům dostane, aby tyto data nerozluštil. Než se budeme zabývat podrobněji problematikou zašifrování dat, vysvětlíme si některé základní pojmy související z bezpečností informací a informačních technologii.

3.1.1 Základní pojmy a definice kryptografie

Kryptografie soubor metod transformace dat zaměřených na to, aby tato data byla pro útočníka nepoužitelná. Takové transformace umožňují řešení dvou hlavních otázek souvisejících s bezpečností informací: ochrana důvěrnosti a ochrana integrity.

Problémy ochrany důvěrnosti a integrity informací jsou úzce spjaty, takže metody řešení jednoho z nich jsou často použitelné pro účely řešení druhého.

Existují různé přístupy ke klasifikaci metod kryptografické transformace informací. Podle typu dopadu na počáteční informaci lze metody kryptografické transformace informací rozdělit do čtyř skupin. (1)



Obrázek 1 - Klasifikace metod kryptografické transformace informací. (1)

3.1.2 Šifrování

Proces šifrování spočívá v provádění vratných matematických, logických, kombinatorických a dalších transformací původních informací, v důsledku, kterých šifrované

informace vypadají jako chaotická sada písmen, čísel, dalších symbolů a binárních kódů.

Pro účely šifrování informací se používá algoritmus transformace a klíč. Zpravidla algoritmus pro konkrétní způsob šifrování zůstává nezměněn. Původní data šifrovacího algoritmu jsou informace, které mají být zašifrovány. Šifrovací klíč obsahuje řídící informaci, která určuje výběr transformace v určitých krocích algoritmu a velikost operandů používaných při provádění šifrovacího algoritmu. Operand je konstanta, proměnná, funkce, výraz a další objekt programovacího jazyka, s kterým jsou prováděny operace. (2) (1)

3.1.3 Steganografie

Na rozdíl od jiných metod kryptografické transformace informací, metody steganografie umožňují skrýt nejen význam uchovávaných či přenášených dat, ale také samotnou skutečnost ukládání nebo přenášení utajovaných informací. Základem všech metod steganografie je maskování soukromých informací mezi otevřenými soubory, tj. tajná data jsou skryta, zatímco jsou vytvářena realistická data, která nelze odlišit od skutečných. Zpracování multimediálních souborů v informačních systémech otevřelo téměř neomezené možnosti pro steganografii.

Grafické a zvukové informace jsou přadstavovány v číselné podobě. Dífky toho v grafických objektech může být nejmenší prvek obrazu zakódován v jednom bajtu. Do dolních bitů určitých bajtů obrazu se podle algoritmu kryptografické transformace umístí bity skrytého souboru. Pokud bude vybrán ten správný algoritmus transformace a obraz, proti kterému bude umístěn skrytý soubor, pro lidské oko bude téměř nemožné rozlišit výsledný obraz od původního. Prostřednictvím steganografie lze maskovat text, obraz, řeč, digitální podpis i šifrované zprávy.

Skrytý soubor může být také šifrován. Pokud někdo náhodou najde skrytý soubor, pak budou šifrované informace vnímány jako porucha systému. Komplexní využití steganografie a šifrování opakovaně zvyšuje složitost řešení problému zjišťování a zveřejňování důvěrných informací. (2) (1)

3.1.4 Kódování

Obsahem procesu kódování informací je nahrazení původního významu zprávy (slov, vět) kódy. Jako kódy mohou být použity kombinace písmen, čísel a znaků. Při kódování a zpětné transformaci se používají speciální tabulky nebo slovníky. V informačních sítích se kódování původní zprávy (nebo signálu) pomocí softwaru a hardwaru používá ke zvýšení

spolehlivosti přenášených informací.

Často kódování a šifrování jsou chybně zaměňováni za stejnou věc, přičemž se zapomíná, že k obnovení kódované zprávy, stačí znát pravidlo nahrazení, zatímco pro dešifrování zprávy, kromě znalosti šifrovacích pravidel, je nutný šifrovací klíč. (2) (1)

3.1.5 Komprese

Kompresi informací lze přiřadit k metodám kryptografické transformace informací jen s určitými výhradami. Cílem komprese je snížení obsahu informací. Současně nelze komprimované informace číst nebo používat bez zpětné transformace. Vzhledem k dostupnosti nástrojů pro kompresi a vratné transformace nelze tyto metody považovat za spolehlivé prostředky pro transformaci kryptografických informací. Dokonce i když skryjí algoritmy, mohou být poměrně snadno řešeny metodami statistického zpracování. Komprimované soubory důvěrných informací proto podléhají následnému šifrování. Pro účely zkrácení doby přenosu dat je doporučeně kombinovat procesy komprese a šifrování informací. (2) (1)

3.1.6 Jaké jsou šifrovací metody?

Po staletí staré historie používání šifrování informací lidstvem byly vynalezeny mnohé metody šifrování nebo šifr. Šifrovací metoda (šifra) je sada reverzibilních transformací otevřených informací do soukromých informací v souladu se šifrovacím algoritmem. Většina šifrovacích metod nekončí test času, ale některé jsou stále používány. Objevení počítačů a počítačových sítí zahájilo proces vývoje nových kódů, s ohledem na možnosti použití počítačové technologie pro šifrování a dešifrování informací. (3)

Moderní metody šifrování musí splňovat následující požadavky:

1. Pevnost šifry, která vydrží kryptoanalýzu (kryptografická stabilita), by měla být taková, aby mohla být otevřena pouze řešením problému úplného hledání klíčů
2. Kryptografická odolnost není zajištěna tajností šifrovacího algoritmu, ale tajností klíče
3. Šifrový text nesmí podstatně překročit původní informace z hlediska objemu
4. Chyby, ke kterým dochází během šifrování, by neměly vést k zkreslení a ztrátě informací

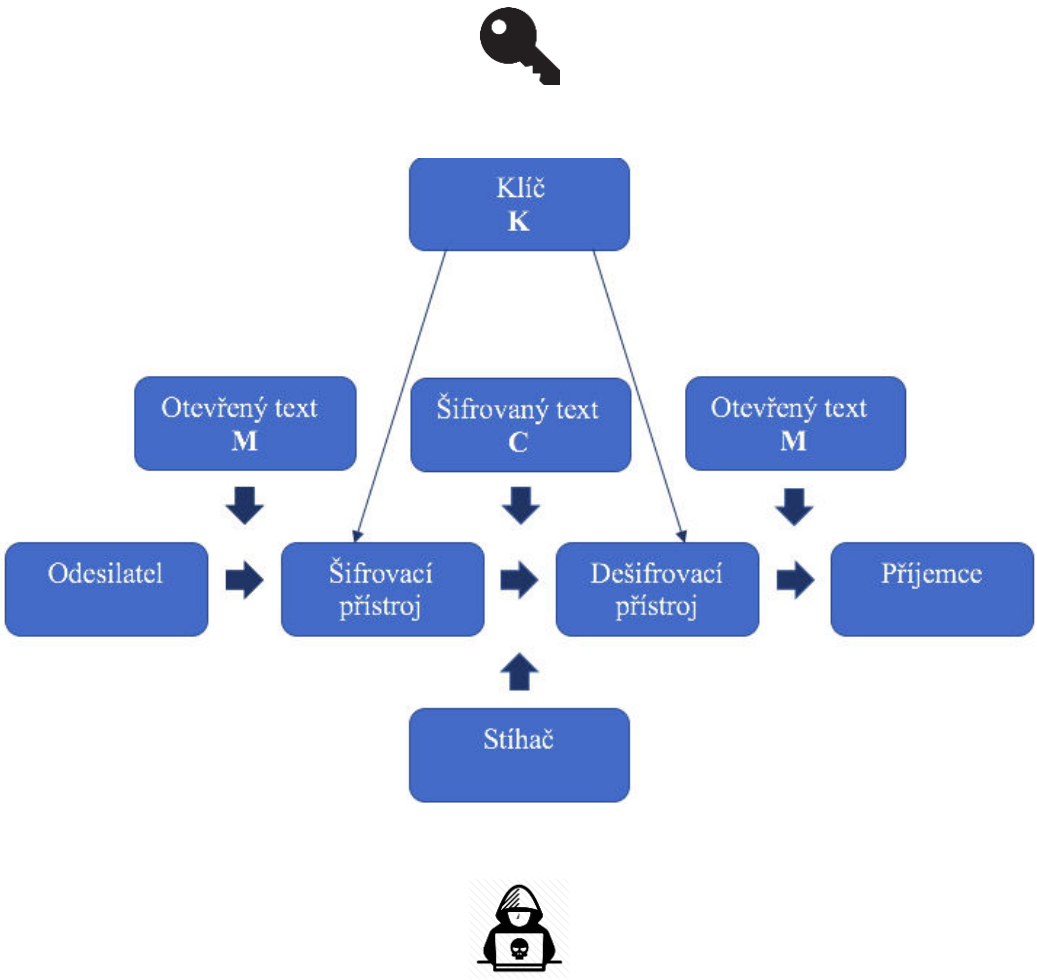
5. Doba šifrování by neměla být velká
6. Náklady na šifrování musí být v souladu s hodnotou skryté informace (4)

3.1.7 Kryptografická odolnost šifry

Kryptografická odolnost šifry je hlavním ukazatelem její účinnosti. To se měří podle času anebo finančních prostředků potřebných kryptografickému analytikovi, aby získat základní informace o šifrotextu, za předpokladu, že mu není znám daný klíč.

Zachovat tajemství šifrovací algoritmu, který se široce používá, je téměř nemožné. Proto nesmí algoritmus mít skryté slabiny, které by kryptanalyti používali. Pokud je tato podmínka splněna, kryptografická odolnost šifry je určena délkou klíče. A to z důvodu, že jediným způsobem otevření šifrované informace je hledání kombinací klíče a implementace dešifrovacího algoritmu. Čas a prostředky vynaložené na kryptoanalýzu tedy závisí na délce klíče a složitosti šifrovacího algoritmu. (3)

Fungování jednoduchého kryptosystému je znázorněno na obrázku 2.



Obrázek 2 - Obecné schéma kryptografického systému. (3)

Odesilatel generuje otevřený text původní zprávy M, který musí být vysílány na zamýšleného příjemce nezabezpečeným kanálem. Kanál je sledován stíhačem, který zachycuje a otevírá vysílanou zprávu. Aby stíhač nenalezl obsah zprávy M, odesílatel ji zašifruje pomocí reverzibilní transformace Eκ a dostává šifrovaný text (nebo kryptogramu) C=Eκ(M), který pošle příjemci.

Oprávněný příjemce, který dostává šifrovaný text C, dešifruje jej pomocí inverzní transformace Dκ(C), a přijímá počáteční zprávu v podobě otevřeného textu M.

Transformace Eκ se vybírá z rodiny kryptografických transformací, nazývaných kryptografické algoritmy. Parametr, podle kterého je vybrána taková samostatná transformace, se nazývá kryptografický klíč K.

Kryptografický systém má různé možnosti implementace: soubor instrukcí, hardware, sadu programů, které umožní zašifrovat otevřený text a dešifrovat šifrovaný text různými způsoby, z nichž jeden je vybrán pomocí konkrétního klíče K. (3)

Šifrovací transformace může být symetrická a asymetrická vzhledem k dešifrovací transformaci. Tato důležitá vlastnost definuje dvě třídy kryptosystémů:

1. Symetrické kryptosystémy (s jedním klíčem)
2. Asymetrické (dvouklíčové) kryptosystémy (s primárním klíčem) (3)

3.2 Symetrické metody

Symetrické kryptografické systémy to je metod šifrování, který používá stejný kryptografický klíč pro šifrování a dešifrování. Před vynálezem schématu asymetrického šifrování jedinou metodou, která existovala, bylo symetrické šifrování. Klíč algoritmu musí být zachován v tajnosti na obou stranách. Šifrovací algoritmus zvolí strany před zahájením odesílání zpráv.

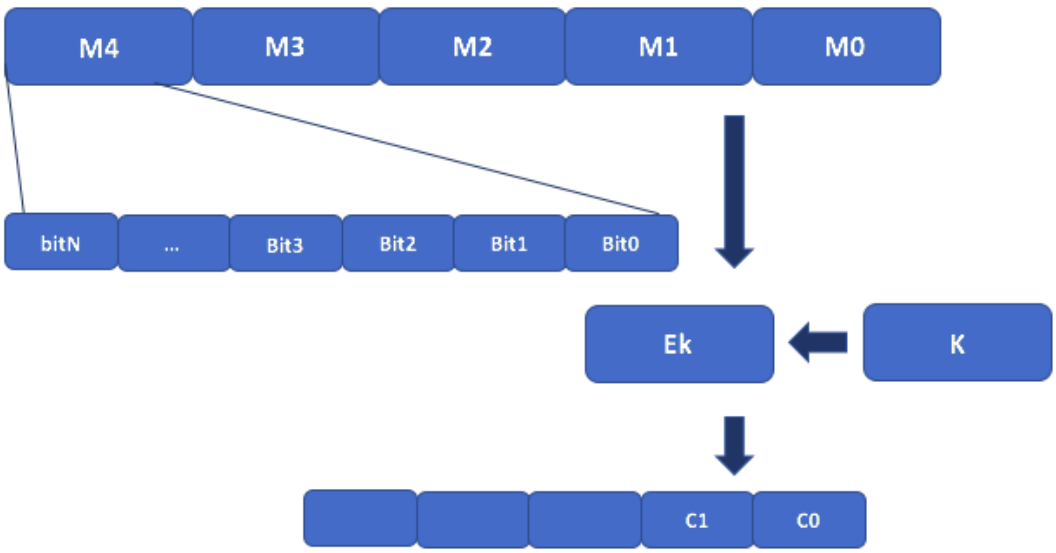
Algoritmy šifrování dat jsou široce používány v počítačové technologii v systémech pro skrytí důvěrných a komerčních informací od škodlivého použití třetími stranami. Hlavním principem v nich je podmínka, že odesilatel a přijímač předem vědí šifrovací algoritmus a klíč ke zprávě, bez kterého se informace je jen soubor symbolů, které nemají žádný význam. (6) (7) (8)

3.2.1 Blokové šifry

Bloková šifra – je druh symetrické šifry, který operuje skupinami bitů s pevnou délkou – „bloky“, jejichž charakteristický rozměr se mění v rozmezí 64 až 256 bitů. Pokud je původní text (nebo jeho zbytek) menší než velikost bloku, pak bude před šifrováním doplněn. Ve skutečnosti bloková šifra je substituce bloků podle abecedy, která, v důsledku toho, může být mono-nebo polyalfabetická. Bloková šifra je důležitou součástí mnoha kryptografických protokolů a je široce používána k ochraně dat přenášených v síti.

Na rozdíl od šifrovacího schéma jednorázové bloky (one-time pad), kde je délka klíče rovna délce zprávy, bloková šifra je schopna zašifrovat jednu nebo více zpráv o celkové délce delší, než je délka klíče. Přenos malého klíče ve srovnání se zprávou přes šifrovaný kanál je mnohem jednodušší a rychlejší než odesílání samotné zprávy nebo klíče o stejné délce, což umožňuje jeho každodenní použití. Šifra však přestává být nerozbitná. Od proudových šifrovacích jednotek se fungování blokové šifry liší zpracováním bitů ve skupinách, nikoliv proudem. Současně jsou blokové šifry spolehlivější, ale pomalejší než proudové šifry.

Mezi výhody blokových šifer patří podobnost šifrovacích a dešifrovacích postupů, které se zpravidla liší pouze podle pořadí akcí. To zjednodušuje vytváření šifrovacích zařízení, protože umožňuje používat stejné bloky v řetězcích šifrování a dešifrování. Flexibilita blokových šifrovacích jednotek umožňuje jejich použití pro vytváření dalších kryptografických primitiv: generátoru pseudonáhodných sekvencí, proudové šifry, MAC funkce a kryptografických hašů. (8)



Obrázek 3 - Obecné schéma fungování blokové šifry. (8)

Režim fungování blokové šifry

Bloková šifra sama o sobě umožňuje šifrovat pouze jednotlivé bloky dat s předem stanovenou délkou. Je-li délka zprávy menší než délka bloku, pak musí být doplněna o požadovanou délku. Je-li však délka zprávy větší, je nutné ji rozdělit do jednotlivých bloků. Existuje několik způsobů šifrování takových zpráv, nazývaných režimy fungování blokové šifry. (9) (8)

3.2.2 Proudové šifry

Proudové šifry jsou symetrické šifry, ve kterých je šifrování prováděno na každém bitu nebo bajtu zdrojového (otevřeného) textu pomocí gamování. Proudovou šifru lze snadno vytvořit na základě blokové šifry, spuštěného ve speciálním režimu. (10) (11)

3.2.3 Srovnání symetrické metody s asymetrickými metodami

Výhody:

1. Rychlost
2. Jednoduchost implementace (díky jednodušším operacím)
3. Menší požadovaná délka klíče pro srovnatelnou odolnost, osvojenost (z důvodu vyššího věku)

Nevýhody:

1. Složitost správy klíčů v rozsáhlé síti
2. Složitost výměny klíčů. Pro účely implementace je nutné vyřešit problém spolehlivého přenosu klíčů ke každému účastníkovi, protože je nezbytný tajný kanál pro přenášení jednotlivých klíčů na obě strany
3. Aby se kompenzovaly nevýhody symetrického šifrování, je v současné době široce používáno kombinované (hybridní) šifrovací schéma, kde se pomocí asymetrického šifrování přenáší klíč relace, který používají strany k výměně dat pomocí symetrického šifrování
4. Důležitou nevýhodou symetrických šifer je neschopnost používat je pro mechanismy tvorby elektronického digitálního podpisu a certifikátů, protože klíč je znám každé straně (7) (12)

3.3. Asymetrické metody

Asymetrické metody – to je šifrovací systém, v němž je primární klíč přenášen přes primární (tedy nechráněný, viditelný pro sledování) kanál a slouží k ověření elektronického podpisu a k zašifrování zprávy. Pro generování elektronického podpisu a dešifrování zprávy je použit soukromý klíč.

Myšlenka na asymetrický kryptosystém je velice úzce spojena s myšlenkou jednosměrných funkcí f(x), podle které je poměrně snadné najít hodnotu f(x) od známé x, zatímco definice x z f(x) je nemožná v přiměřené době.

Samotná jednosměrná funkce je však v aplikaci zbytečná: může šifrovat zprávu, ale nemůže ji dešifrovat. Proto kryptografie s veřejným klíčem používá jednosměrné funkce s mezerou. Mezerou je některé tajemství, které pomáhá dešifrovat. To znamená, že existuje takový ukazatel y, že s vědomím f(x) a y, lze vypočítat x.

Například pokud rozebíráte hodiny do mnoha komponent, je velmi obtížné sestavit nově pracující hodiny. Pokud však existuje montážní instrukce (mezera), může být tento problém snadno vyřešen. (12)

3.3.1 Výhody asymetrických algoritmů nad symetrickými

Výhody:

1. Nepotřebujete předem předávat soukromý klíč bezpečným kanálem
2. Pouze jedna strana zná dešifrovací klíč, který musí být uchován v tajnosti (v symetrické kryptografii je tento klíč známý oběma stranám a musí být tajně chráněn)
3. Ve velkých sítích je počet klíče v asymetrickém kryptosystému výrazně menší než v symetrickém

Neýhody:

1. Je obtížnější provést změny algoritmu
2. Relativně delší klíče
3. Šifrování a dešifrování pomocí dvojice klíčů se provádí o dva nebo tři řády pomalejší než šifrování a dešifrování stejného textu s použitím symetrického algoritmu (7) (12)

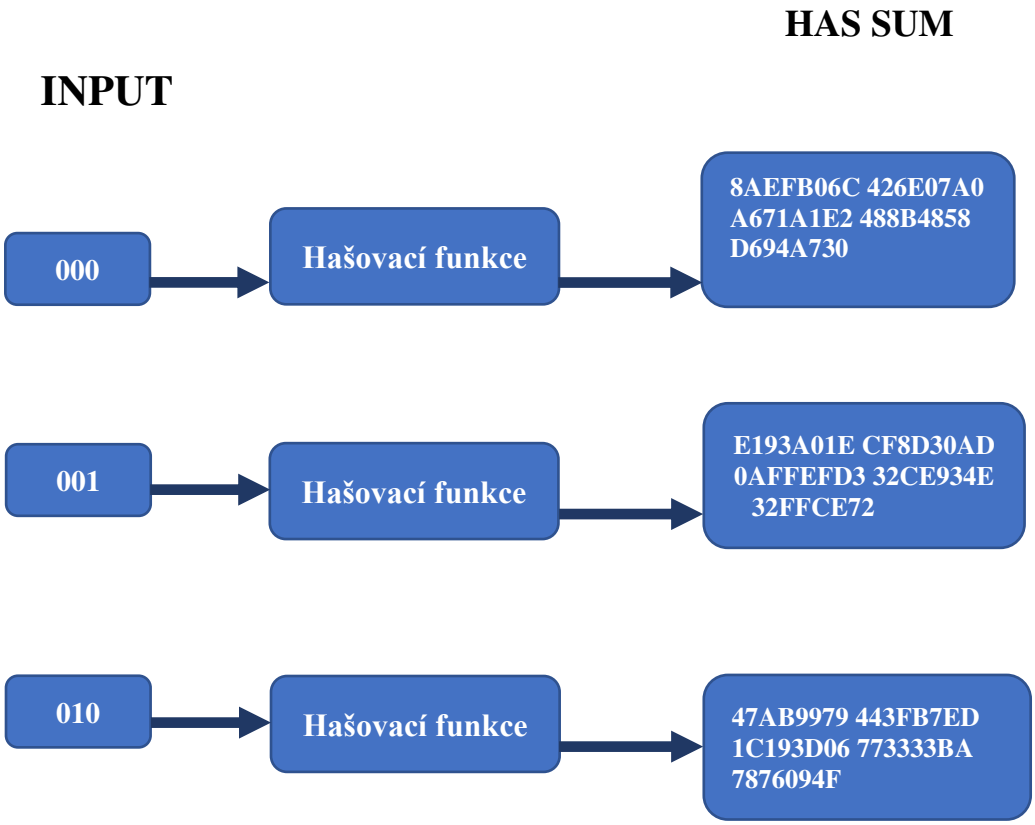
3.4 Hašovací funkce

Hašovací funkce je transformace pole vstupních dat libovolné délky do bitového řetězce s pevnou délkou, prováděná určitým algoritmem.

Ideální hašovací funkce je šifrovací funkce, na kterou lze přiřadit pět základních vlastností:

1. Definovanost. Se stejnými vstupními daty bude výsledek hašovací funkce stejný (stejná zpráva vždy vede ke stejnému hash)
2. Vysokorychlostní výpočet hodnoty hash pro každou danou zprávu
3. Neschopnost generovat zprávu z její hodnoty hash, s výjimkou pokusů o vytvoření všech možných zpráv

4. Přítomnost lavinového efektu. Dokonce i malá změna zpráv by měla měnit hodnoty hash, a to natolik široce, že nové hodnoty hash neodpovídají starým hodnotám hash
5. Neschopnost najít dvě různé zprávy se stejnými hodnotami hash (13)



Obrázek 4 - Hašovací funkce. (13)

3.5 Moderní kryptografie

Pro moderní kryptografii je charakteristické použití otevřených šifrovacích algoritmů, které zahrnují nezbytné použití výpočetních nástrojů. Existuje více než deset osvědčených šifrovacích algoritmů, které v případě použití klíčů dostatečné délky a správné implementace algoritmu jsou kryptograficky stabilní.

Společné algoritmy:

1. Symetrické AES, Blowfish, Twofish, RC4, Camellia, Serpent, Kuzynechik
2. Asymetrické RSA
3. Hašovací funkce MD4, MD5, SHA-1, SHA-2 (14)

3.5.1 Moderní symetrické algoritmy

AES

Advanced Encryption Standard (AES), také známý jako Rijndael, je symetrický blokový kódovací algoritmus (velikost bloku 128 bitů, klíč 128/192/256 bitů). Tento algoritmus je dobře analyzován a je nyní široce používán, stejně jako jeho předchůdce DES. (23)

AES – výhody i nevýhody

Největší výhodou algoritmu AES je splnění přísných kritérií zadaných úřady spojených států NIST pro jejich výběrové řízení. Mezi tyto kritéria spadá rychlost, šifrování i dešifrování, implementace, cena a bezpečnost a další. AES nejen všechna tato kritéria splnil, ale byt také bez konkurenčně nejlepší šifrou v kritériu rychlost šifrování a dešifrování. Na druhou stranu největší nevýhodou tohoto algoritmu jsou nároky na paměť, pokud si přejeme šifrovat i dešifrovat ve stejnou dobu. Navíc, v době, kdy se AES stal bez konkurenčně nejlepší šifrou, na ni nebyli známé žádné útoky. V přítomnosti tomu tak již samozřejmě není. AES ale nadále odolává a doposud není znám žádný případ prolomení této šifry. (23)

Blowfish

Blowfish je kryptografický algoritmus, který implementuje blokové symetrické šifrování s proměnnou délkou klíče. Provádí se na jednoduché a rychlé operace: XOR, substituce, přidávání. Je nekomerční a volně distribuovaný.

Tento algoritmus se skládá ze dvou částí: rozšiřování klíče a šifrování dat. Ve fázi rozšiřování klíče je původní klíč (o délce až 448 bitů) převeden do 18 32 - bitových podklíčů a do 4 32 - bitových S-bloků obsahujících 256 prvků. Celkový objem přijatých klíčů se rovná (18 + 256 * 4) * 32 = 33344 bitů nebo 4168 bajtů. (16)

Blowfish – výhody i nevýhody

Hlavním rozdílem mezi algoritmem Blowfish a ostatními algoritmy je ten, že s-box je generován z klíče poměrné délky (32-448 bitů) složitým a netriviálním způsobem. I když toto se může zdát jako nevýhoda, opak je pravdou. Díky tohoto kroku, Blowfish je jeden z nejbezpečnějších algoritmů. Napadnutí tohoto algoritmu je časově velice složitý a zdlouhavý proces, jelikož hacker musí spočítat bezpočet ‘s-boxů’ aby odzkoušel jeden klíč. Přitom kombinací k těmto klíčům je nekonečno. Další výhodou tohoto algoritmu je fakt, že byl navržen s ohledem na softwarovou implementaci na aktuální softwary a také to že není potřeba vlastnit žádnou licenci na tento algoritmus.

Nehodí se pro šifrování velkých databází. (16)

Twofish

Twofish je symetrický blokový šifrovací algoritmus s velikostí bloku 128 bitů a délkou klíče až 256 bitů. Počet cyklů je 16.

Významnými rysy algoritmu jsou použití předem kompatibilních a klíčových závislých náhradních uzlů a komplexní schéma vývoje šifrovacích podklíčů. Polovina n-bitového šifrovacího klíče je použita jako samotný šifrovací klíč, druhá polovina slouží k modifikaci algoritmu (závisí na ní náhradní uzly).

Twofish rozděluje vstupní 128 - bitový datový blok na čtyři 32 - bitové podbloky, s kterými po vstupním bělení (input whitening) se provádí 16 cyklů transformací. Po posledním cyklu se provádí výstupní bělení (output whitening).

Tento algoritmus je vytvořen na základě algoritmů Blowfish. (15)

RC4

RC4 je proudová šifra, která se v počítačových sítích běžně používá v různýchsystémech informační bezpečnosti.

Algoritmus RC4, stejně jako jakákoli proudová šifra, je vytvořen na základě generátoru pseudonáhodných bitů. Na vstup generátoru je zapsán klíč, a na jeho výstupu jsou čitelné pseudonáhodné bitové kódy. Délka klíče může být od 40 do 2048 bitů. Generované bity mají jednotnou distribuci. Jádro algoritmu proudových šifer se skládá z funkce -

generátoru pseudonáhodných bitů (gama), který produkuje proud klíčových bitů (klíčový proud, gama, sekvence pseudonáhodných bitů).

RC4 – výhody i nevýhody

Výhodou šifry RC4 je rozhodně rychlost při zpracování dat. Tato výhoda se ale také stala hlavní nevýhodou této šifry. A to převážné díky výzkumu, který dokázal identifikovat vliv některých bitů soukromého klíče na několik prvních bitů pseudonáhodného proudového klíče. Proto musíme doporučit zvážit, jak tuto slabinu eliminujete hned na začátku, pokud se rozhodnete využít tuto šifru. (16)

Camellia

Camellia je symetrický blokový šifrovací algoritmus (velikost bloku 128 bitů, klíč 128, 192, 256 bitů).

Struktura algoritmu je založena na klasickém Feistelovém řetězci s předběžným a konečným bělením. Cyklická funkce využívá nelineární transformaci (S-bloky), blok lineární disperze každých 16 cyklů (operace XOR pro každý bbajt) a permutaci bajtů.

V závislosti na délce klíče má 18 cyklů (klíč 128 bitů) nebo 24 cyklů (klíč 192 a 256 bitů). Podpora algoritmu Camellia byla zavedena v roce 2008 v prohlížeči Mozilla Firefox 3, ale v roce 2014 byla v Mozilla Firefoxu 33 zakázána. Tento algoritmus je patentován, ale je distribuován pod řadou licencí bezplatně, zejména je součástí projektu OpenSSL. (17)

Serpent

Serpent je symetrický blokový šifrovací algoritmus. Serpent má velikost bloku 128 bitů a možné délky klíče 128, 192 nebo 256 bitů. Algoritmus Serpent je síť SP, ve které je celý datový blok 128 bitů v každém kole rozdělen do čtyř slov o délce 32 bitů. Všechny hodnoty použité v šifrování jsou reprezentovány bitovými proudy. Bitové indexy se pohybují od 0 do 31 pro 32 - bitová slova, 0 až 127 pro 128 - bitové bloky, 0 až 255 pro 256 - bitové klíče a tak dále. Pro interní výpočty jsou všechny bity hodnot zobrazeny v přímém pořadí (little-endian).

Serpent šifruje otevřený text P o délce 128 bitů do šifrovaného textu C o stejné délce 128 bitů během 32 kol s použitím 33 podklíčů K0, ..., K32 o délce 128 bitů. Délka použitého klíče může mít různé hodnoty, ale pro účely specifikace fixujeme jejich délku na 128, 192

nebo 256 bitů. Krátké klíče s délkou menší než 256 bitů jsou doplněny do celkové délky 256 bitů. (18)

Kuznyechik

Kuznyechik – to je symetrický blokový šifrovací algoritmus s velikostí bloku 128 bitů a délkou klíče 256 bitů, který používá Feistelovou síť pro vytváření kruhových klíčů.

Šifrování je založeno na systematickém uplatňování několika kolech, z nichž každý obsahuje tři konverze: sčítání s klíčem, převod blokem vyhledáváním a lineární transformace. (19)

3.5.2 Moderní asymetrické algoritmy

RSA

Algoritmus RSA (podle prvních písmen jmén jeho tvůrců Rivest-Shamir-Adleman) je založen na vlastnostech prvočísel (velmi velkých). Kryptografický algoritmus RSA patří dnes mezi nejpoužívanější algoritmy asymetrické kryptografie. Využívá se jak k šifrování objemově menších zpráv, tak i pro digitální podpis.

Kryptografická odolnost RSA je založena na předpokladu, že je extrémně obtížné, pokud vůbec realistické, určit tajný klíč na základě otevřeného klíče. K tomu je třeby vyřešit problém existence deliterů obrovského celkového čísla. Doposud tento problém nikdo nerozhodl pomocí analytických metod, a algoritmus RSA může být popraskán pouze úplným vyhledáváním. (22)

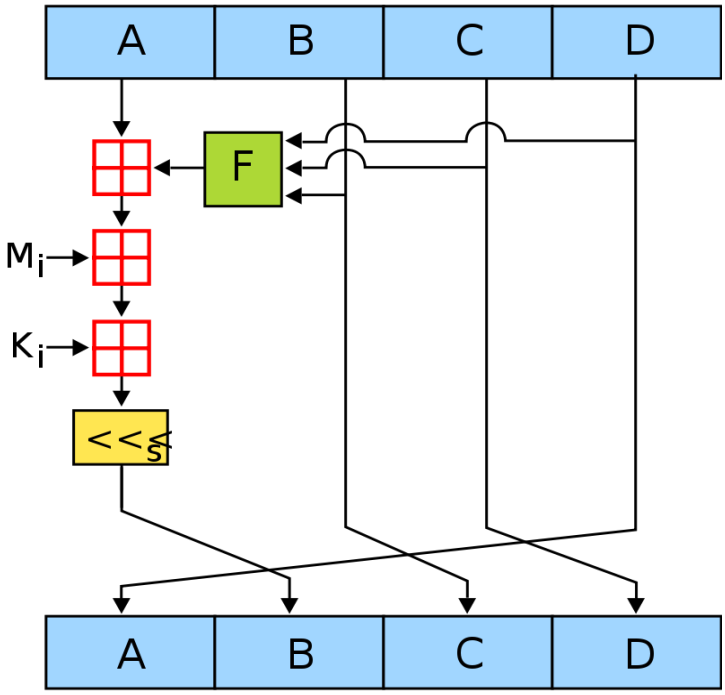
Nevýhody RSA

Asymetrický šifrovací algorimus RSA se prokázal být velmí pomalý z hledika procesů šifrování a dešifrování. Z toho důvodu se tento algoritmus používá převážně pro šifrování soukromých klíčů nebo při digitálním podpisu. Oba tyto úkony totiž využívájí funkce dekomprese. (22)

3.5.3 Moderní hašovací funkce

MD4

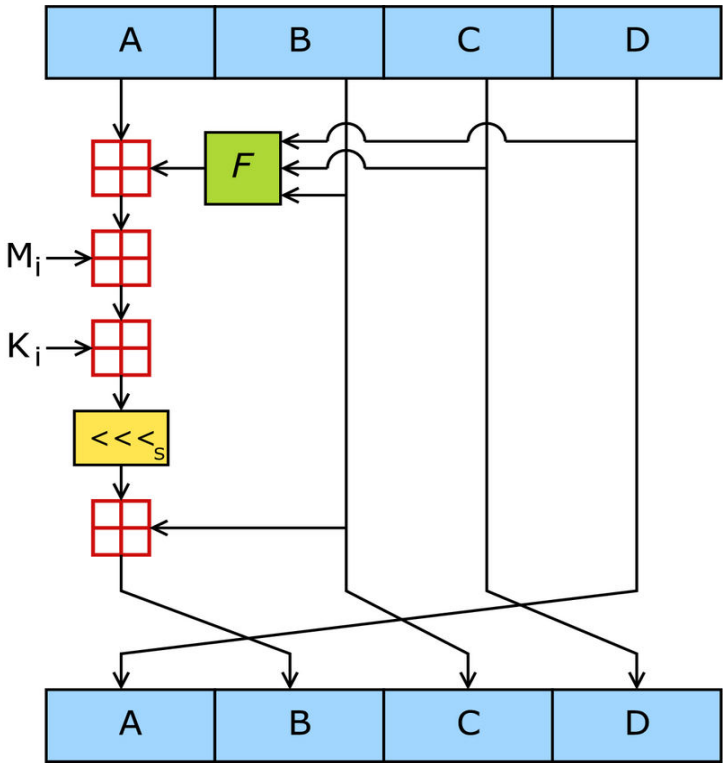
MD4 je kryptografická hašovací funkce. Pro libovolnou vstupní zprávu funkce generuje 128 - bitovou hodnotu hash, která se nazývá digest zprávy. Tento algoritmus se používá v autentizačním protokolu MS-CHAP vyvinutém společností Microsoft pro účely provádění ověřovacích postupů pro vzdálené pracovní stanice Windows. Je to předchůdce MD5. (13) (21)



Obrázek 5 - Schéma fungování MD4. (21)

MD5

MD5 je 128 - algoritmus hašovací funkce, vyvinutý v roce 1991. Je určen k vytváření "výtisků", nebo digestů zprávy libovolné délky a následné kontroly jejich pravosti. Široce se používá k ověření integrity informací a ukládání hesel. (13) (21)



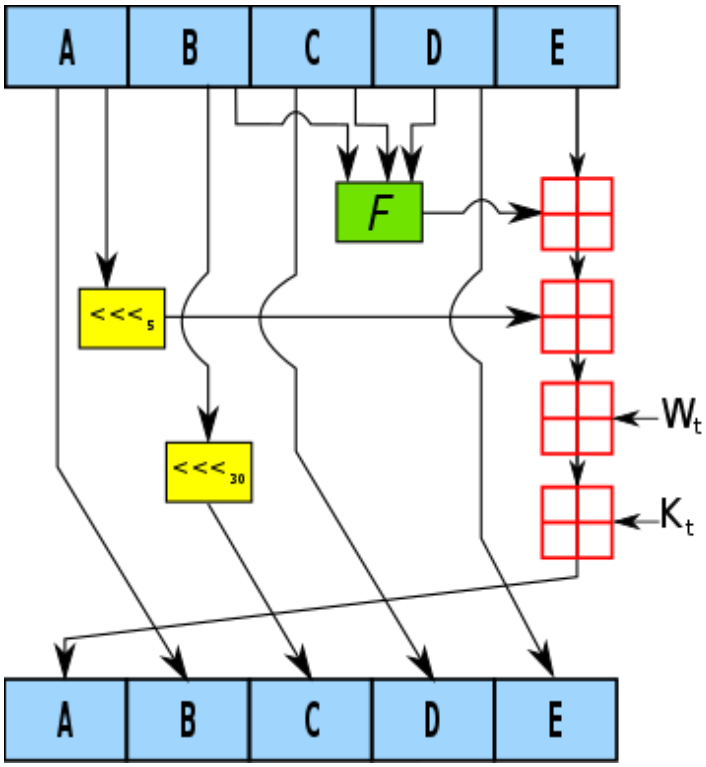
Obrázek 6 - Schéma fungování MD5. (21)

SHA-1

SHA-1 je kryptografický hašovací algoritmus.

Pro vstupní zprávu libovolné délky (maximálně 2^64-1 bit, což je přibližně 2 exabajty), algoritmus generuje 160 - bitovou hodnotu hash, nazývanou také digest zpráv. Používá se v mnoha kryptografických aplikacích a protokolech. Principy, které jsou základem SHA-1, jsou podobné těm, které Ronald Rivest používal při navrhování MD4.

SHA-1 implementuje hašovací funkci postavené na myšlence funkce komprese. Vstupy funkce komprese jsou 512 - bitový blok zprávy a výstup předchozího bloku zprávy. Výstupem je hodnota všech hashových bloků až do tohoto bodu. Jinými slovy, hash bloku Mi se rovná hi = f (Mi, hi-1). Hashová hodnota celé zprávy je výstup posledního bloku. (13) (22)



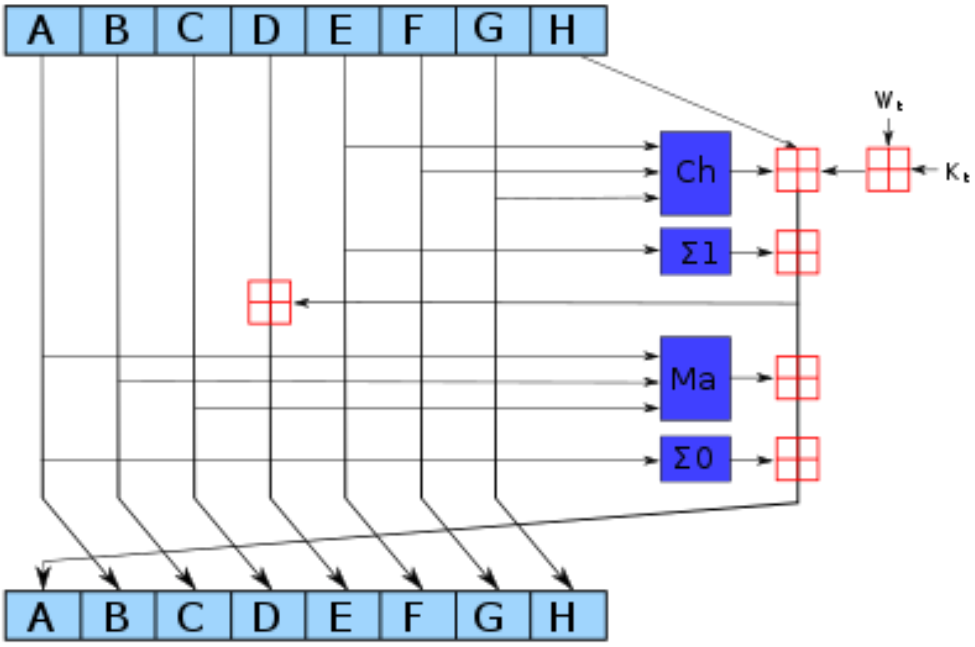
Obrázek 7 - Schéma fungování SHA-1. (22)

SHA-2

SHA-2 je bezpečný hašovací algoritmus, verze 2. Rodina kryptografických algoritmů - jednosměrných hašovacích funkcí.

Hašovací funkce rodiny SHA-2 jsou postaveny na základě Merkleové-Damgårdové konstrukce. Výchozí zpráva po přidání je rozdělena na bloky, každý blok o délce 16 slov. Algoritmus předává každý blok zprávy smyčkou se 64 nebo 80 iterací (kol). Při každé iteraci se převedou dvě slova, zbývající slova jsou nastavena na funkci transformace. Výsledky zpracování každého bloku se spočítají, jejich součet je hodnotou hašovací funkce. Inicializace interního stavu je však výsledkem zpracování předchozího bloku. Proto není možné zpracovávat bloky samostatně a přidávat výsledky.

Hašovací funkce se používají k vytváření "otisků" nebo "digestů" pro zprávy libovolné délky. Používají se v různých aplikacích nebo komponentech spojených s ochranou informací. (13) (22)



Obrázek 8 - Schéma fungování SHA-2. (22)

3.6 Představení vybraných programů pro šifrování souborů

Existuje hodně různých programů, které umožňují šifrovat data. Tyto programy můžou být placené i bezplatné. Existují navíc i řešení, které jsou zabudovány do operačního systému. Jako například BitLocker ve Windows a FireVault v macOSu.

Bohužel v softwaru takové třídy jsou často uváděny záložky pro snadné hackování. Na vývojáře těchto softwaru jsou vždy kladené maximální požadavky na bezpečnost a ochranu jejich produktu.

Dříve byl nejlepším vedoucím programem TrueCrypt. Bohužel vývojáři zastavili vývoj tohoto programu. Případ s programem TrueCrypt je vůbec celkové komplikovaný a nejasný. Mnozí věří, že za zastavení vývoje tohoto programu stojí speciální služby, pro které tento program mohl představovat hrozbu.

Ale kvůli tomu, že původní kód TrueCrypt byl otevřený, jiný tým se zavázal k dalšímu vývoji a podpoře. Odstranily několik zranitelných míst a vydaly novou verzi nazvanou VeraCrypt, kterou jsem si vybral za účelem studia kryptografických metod.

Tento program je neustále kontrolován a prochází bezpečnostní audity. (24)

Aktuální verze	1.21
Licence	Freeware
Česká lokalizace	ANO
Autor	IDRIX
Web programu	www.veracrypt.fr
Nutnost instalace	ANO
Velikost programu	22,1 MB
Kompatibilita s OS	Windows, MacOS, Linux
Poslední aktualizace	9.6.2017

Tabulka 1 - Parametry programu Veracrypt, zdroj: (autor)

4 Vlastní práce

4.1 Porovnání kryptografických metod

Jak již bylo řečeno v úvodu, cílem této práce je porovnat dostupné kryptografické metody pro oblast ochrany souborů a doporučit vhodnou metodu, nebo skupinu metod v závislosti na jejich výkonnostních charakteristikách. Přičemž nejprve je nutné vybrat aplikace pro porovnáni kryptografických metod pro oblast ochrany souborů.

Výběr vhodné aplikace pro porovnáni kryptografických metod

Pro porovnáni kryptografických metod, rozhodl jsem se použít aplikaci VeraCrypt Verze 1.21.

Přehled Hardwaru

Výsledky porovnání algoritmů závisí přímo na typu hardwaru počítače, proto ukazuju hardware, na kterém budu provádět testování.

Komponenty	Parametry
Procesor	Intel Core i5
Rychlost Procesoru	1,7 GHz
Core Boost Frekvence	2,9 GHz (2 900 MHz)
Počet Procesorů	1
Celkový Počet Jader	2
Cache procesoru	3 MB
Čip grafické karty	Intel HD Graphics 6000
Operační paměť	4 GB
SSD	256 GB

Tabulka 2 - Charakteristika hardwaru, zdroj: (autor)

Výběr kryptografických metod

Pro porovnání kryptografických metod pro ochranu souborů používám následující šifrovací algoritmy: AES, Serpent, Twofish, Camellia, Kuznechyk, a také kombinace těchto algoritmů:AES(Serpent(Twofish)),Twofish(Serpent),AES(Twofish), Serpent(Twofish(AES)), Serpent(AES).

Výběr metodiky pro porovnání kryptografických metod

Pro porovnání kryptografických metod používam metodu párového porovnání.

Výběr souborů pro šifrovaní

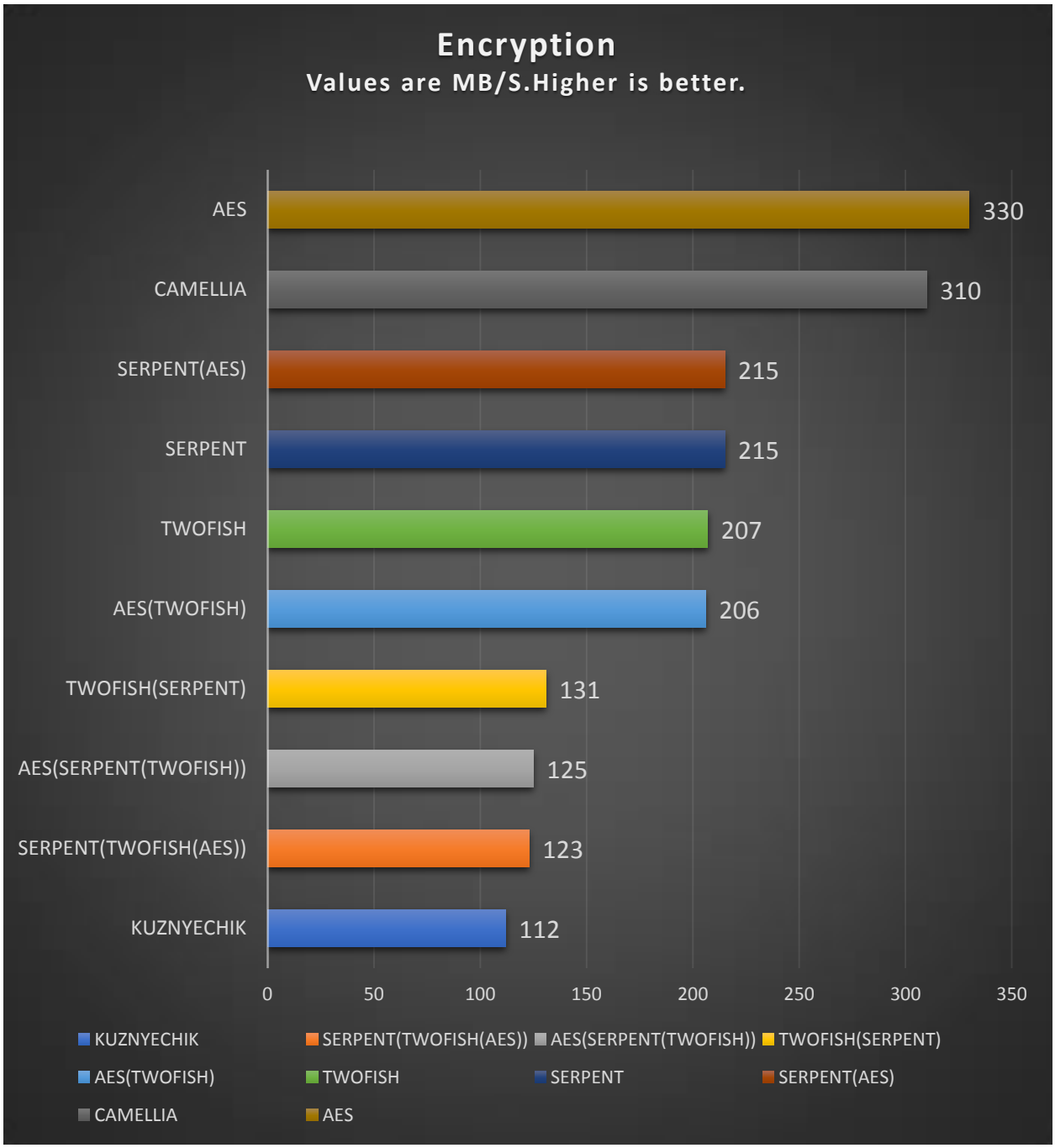
Testovat algoritmy šifrování budu na souborech o velikosti 1024 MB. Údaje, které jsou předložený ve formátu .JPEG.

4.1.1 Výběr nejvhodnějšího algoritmu

Všechny představené algoritmy je teď nyní potřeba vzájemně porovnat.

Test 1

Rychlost šifrovaní souborů

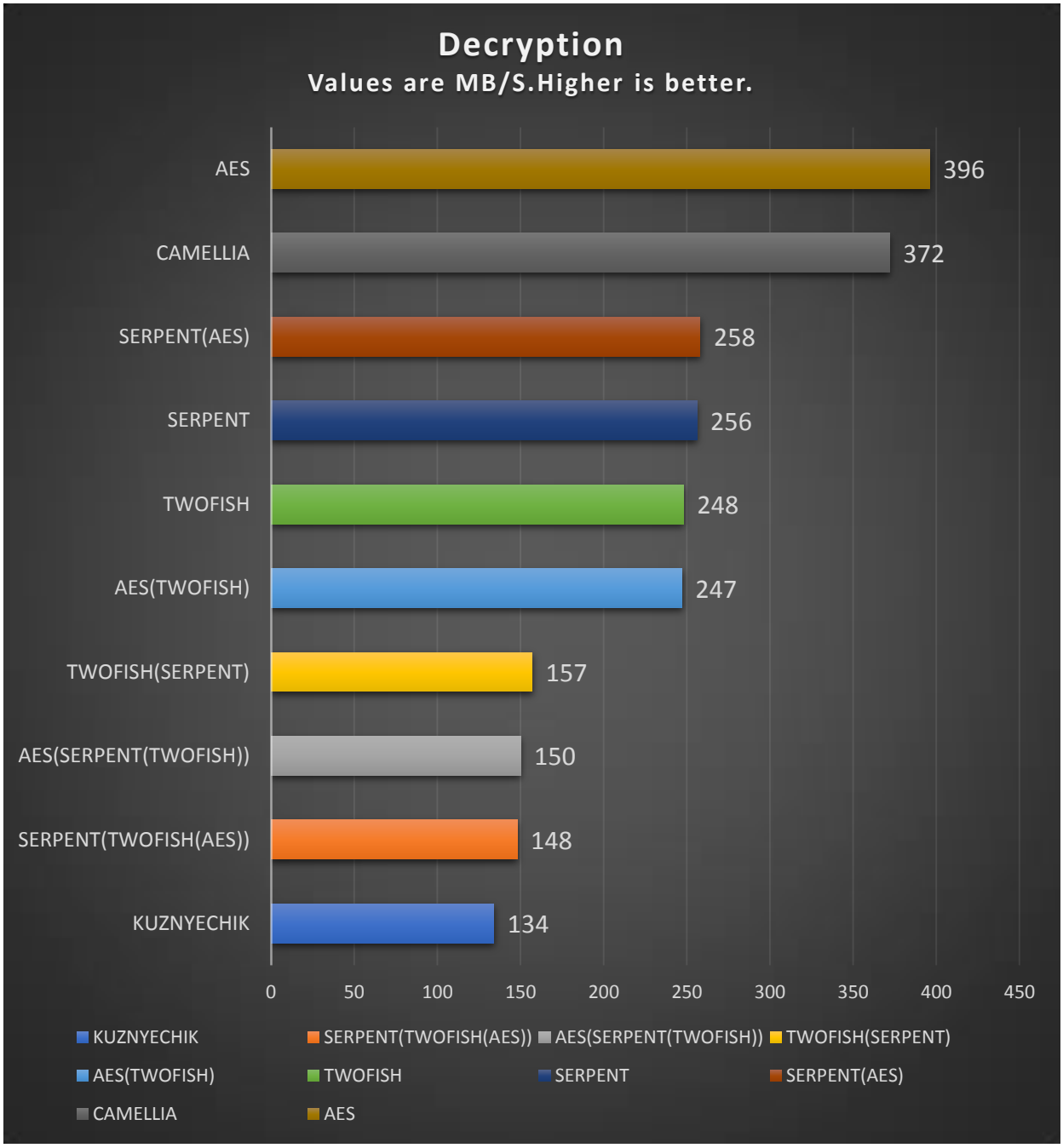


Tabulka 3 - Rychlost šifrování souborů, zdroj: (autor)

Výsledky ukazují, že nejvyšší rychlost šifrování souborů má algoritmus AES a algoritmus Camellia, s rychlostí 330 Mb/s a 310 Mb/s. Přesuňme se k druhému testu.

Test 2

Rychlost dešifrovaní souborů

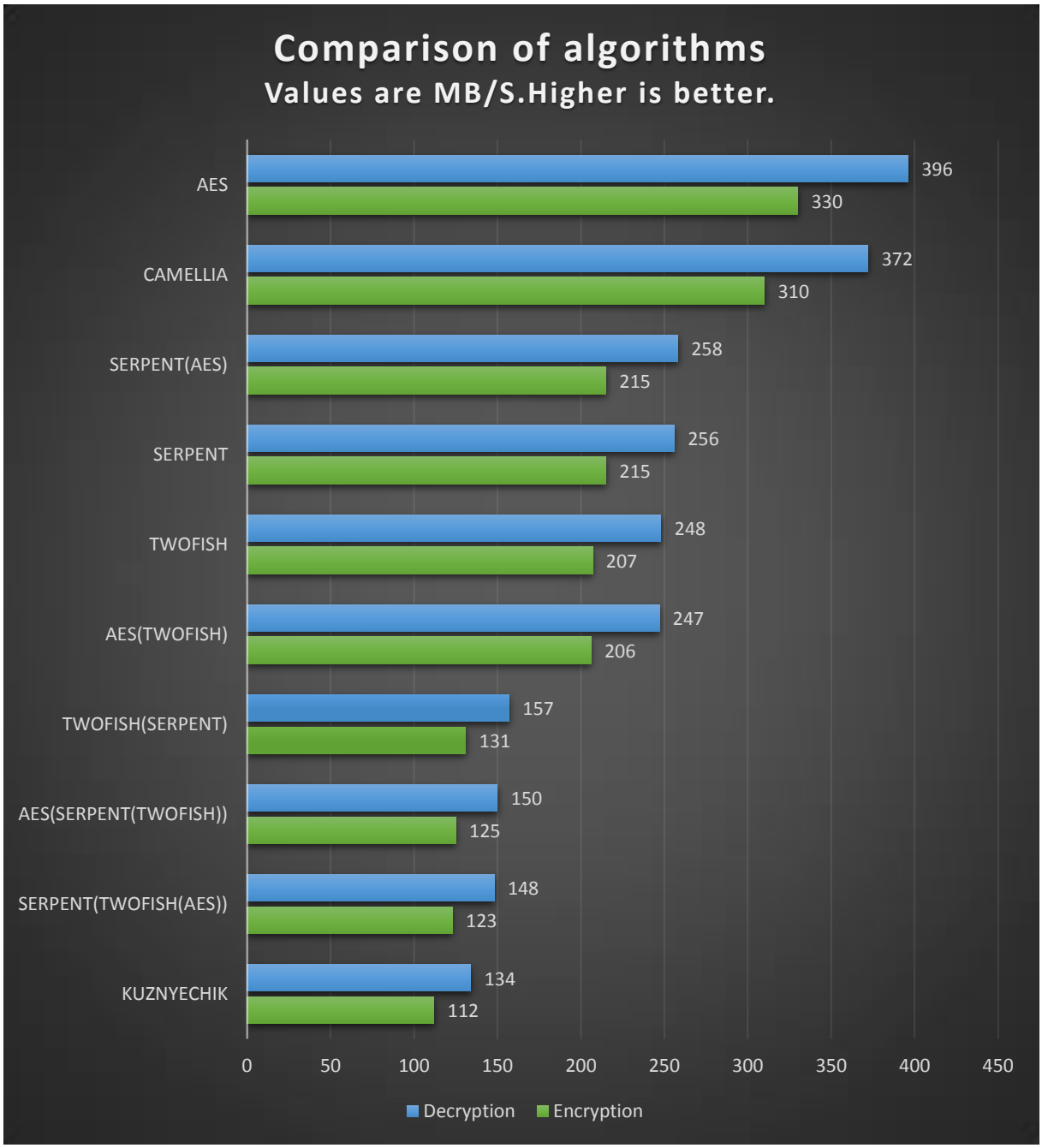


Tabulka 4 - Rychlost dešifrování souborů, zdroj: (autor)

Výsledky opět ukazují, že nejvyšší rychlost souborů má algoritmus AES a algoritmus Camellia, s rychlostí dešifrování souborů 396 Mb/s a 372 Mb/s. S rozdílem v dešifrování jen 24 Mb/s.

Výsledky porovnání šifrování a dešifrování souborů

Po provedení testů jsem se rozhodl, že bude vhodné vypsat data do jedné tabulky.

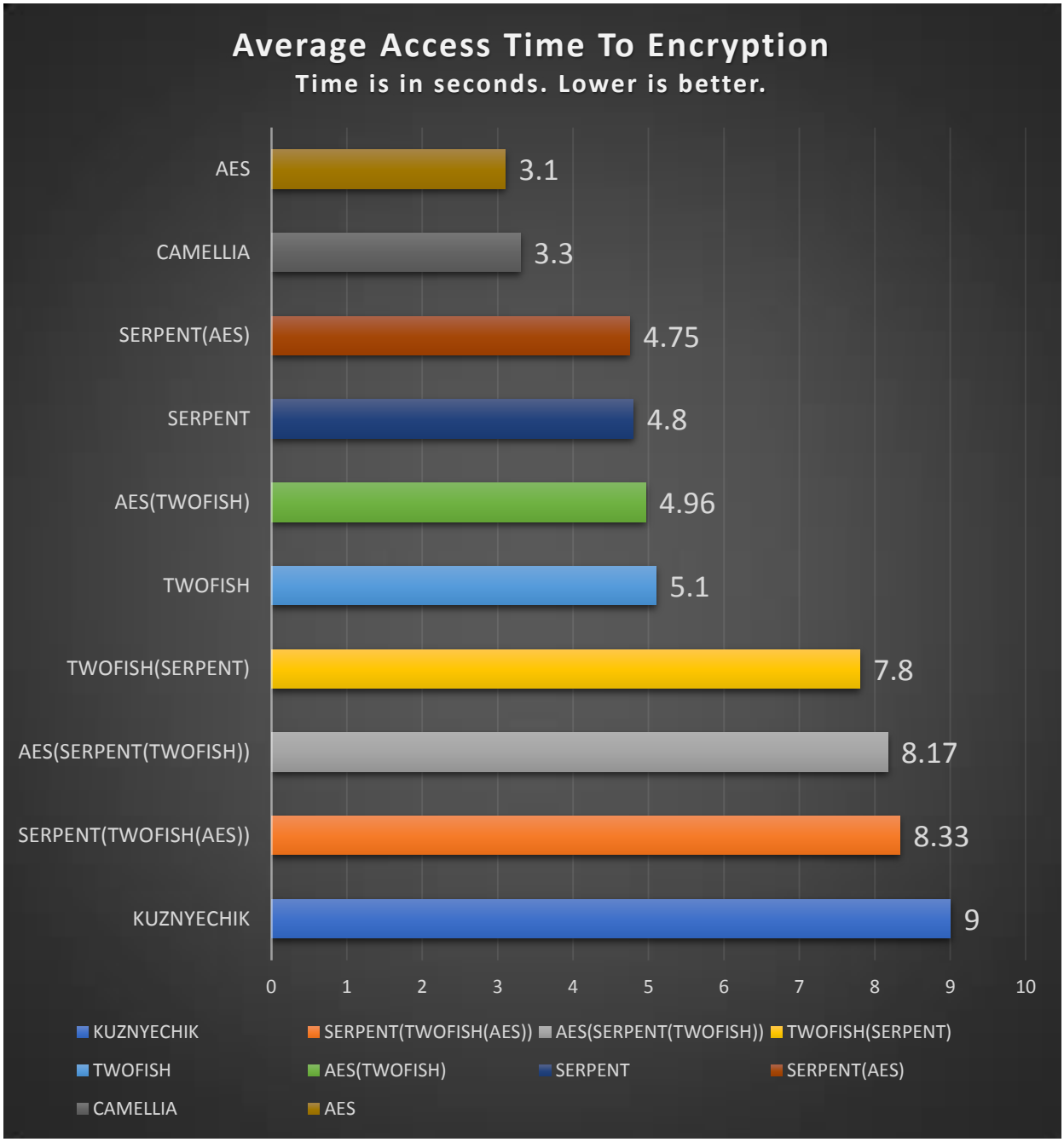


Tabulka 5 - Výsledky porovnání šifrování a dešifrování souborů, zdroj: (autor)

Porovnání algoritmů pro šifrování jsme zjistili, že nejideálnějším algoritmem pro šifrování našich souborů, je AES, který získal 330 MB/s šifrování a 396 MB/s dešifrování. Na druhém místě skončil l algoritmus Camellia.

Test 3

Průměrná doba šifrování souborů



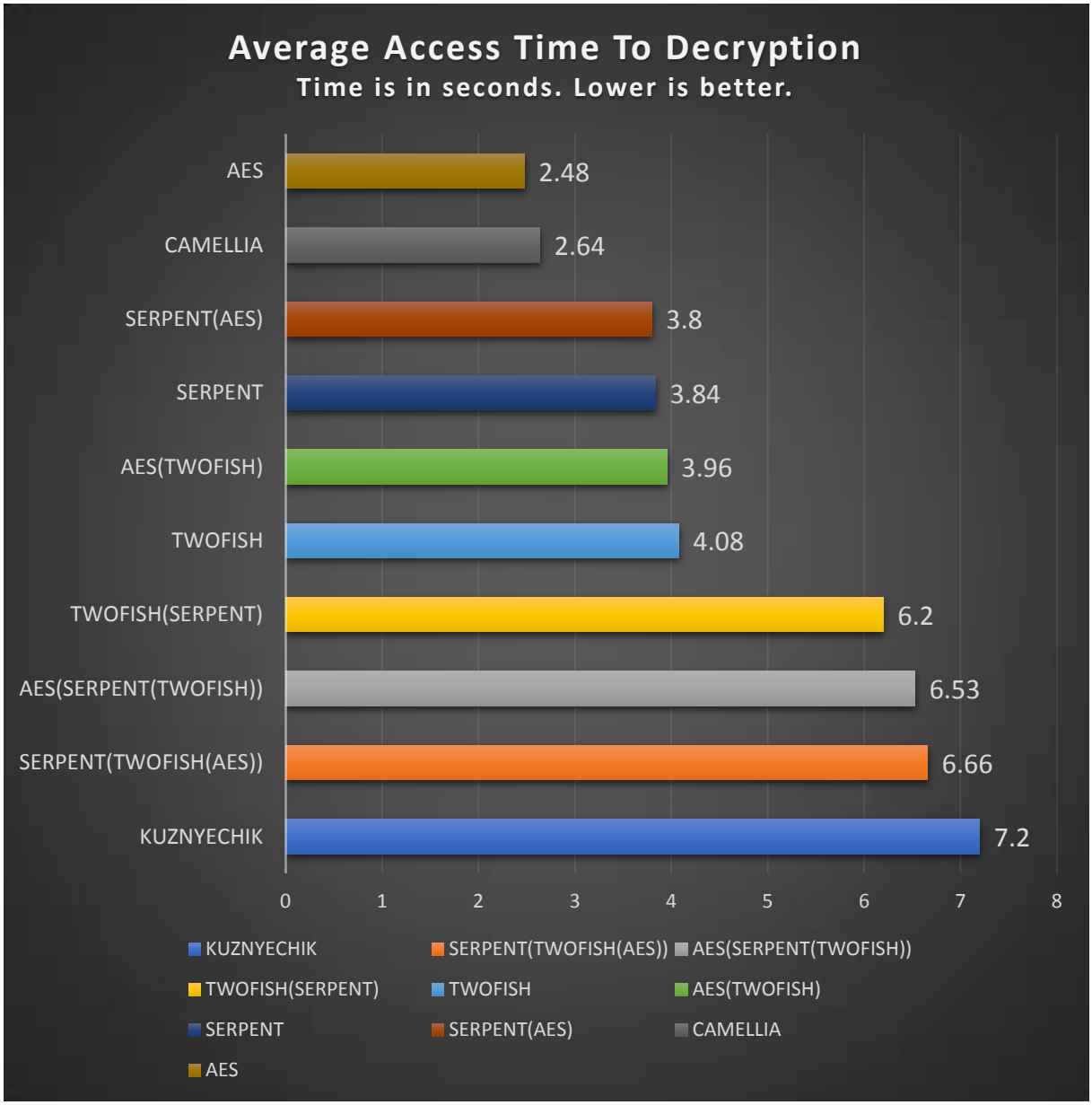
Tabulka 6 - Průměrná doba šifrování souborů, zdroj: (autor)

Vynikající výkon znovu prokázal má algoritmus AES s dobou šifrování souborů 3,1 vteřiny a poté algoritmus Camellia dobou šifrování souborů 3,3 vteřiny.

V tomto testu se prokázalo, že algoritmus AES(TwoFish), který ve všech ostatních testech skončil na 6 místě si zlepšil pozici o jednu příčku. Tím pádem se algoritmus AES(Twofish) dostal na 5 místo v žebříčku.

Test 4

Průměrná doba dešifrování souborů



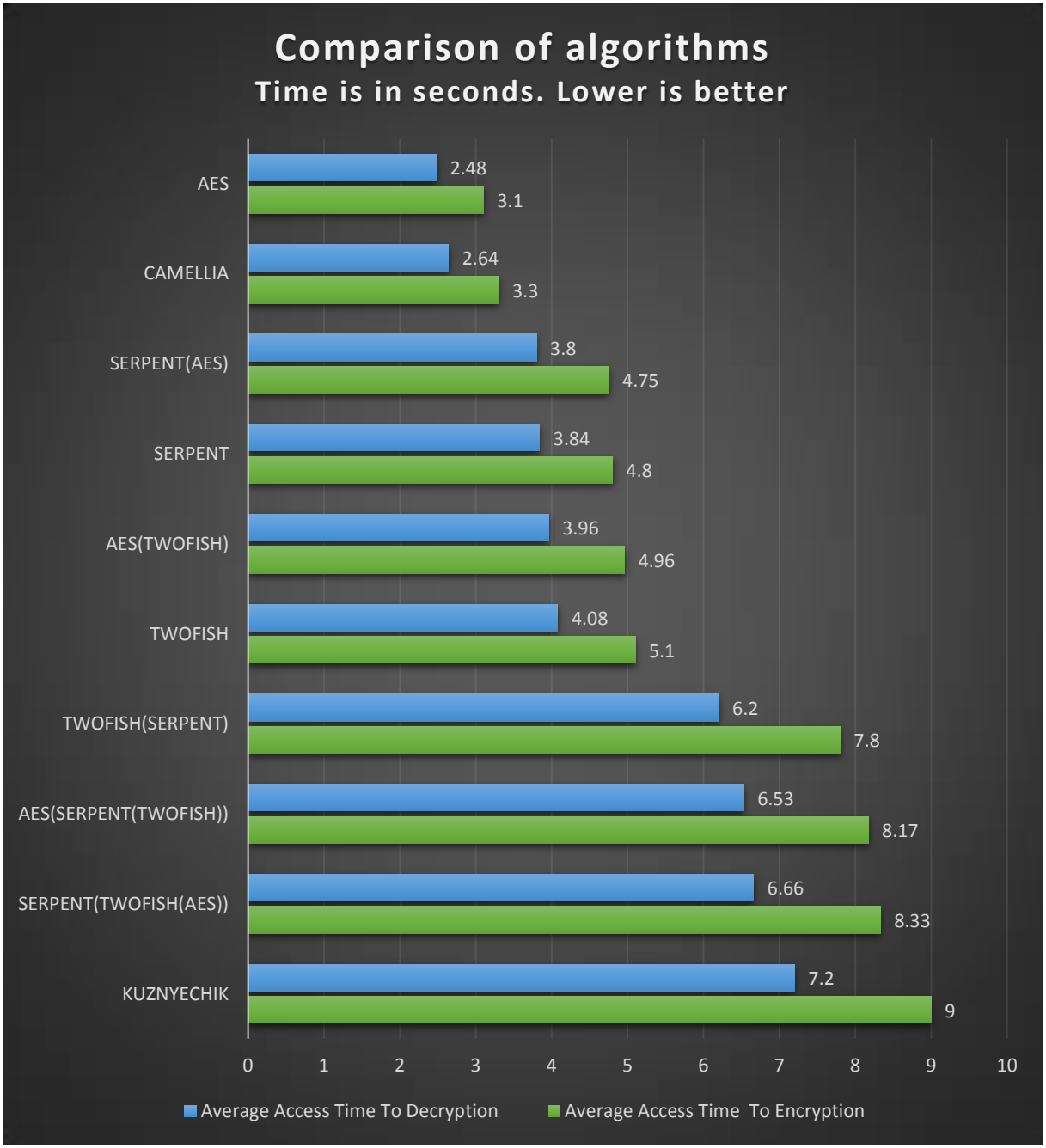
Tabulka 7 - Průměrná doba dešifrování souborů, zdroj: (autor)

Na základě provedených testů můžeme vidět, že nejlepší ukazatele mají algoritmus AES s dobou dešifrování souborů 2,48 vteřiny

a algoritmus Camellia s dobou dešifrování souborů 2,64 vteřiny.

Také si můžete všimnout, že algoritmus Serpent(AES) má mnohem lepší výsledky než zbytek testovaných algoritmů a získává 3 místo s dobou šifrování souborů 3,1 vteřiny.

Výsledky porovnání průměrné doby šifrování a dešifrování souborů



Tabulka 8 - Výsledky porovnání průměrné doby šifrování a dešifrování souborů, zdroj: (autor)

Nejrychlejší se ukázal být algoritmus AES. Druhé a třetí místo obsadili Camellia a Serpent(AES). Algoritmus Kuznyechik se podle výsledků všech testů zařadil na poslední místo.

4.1.2 Vyhodnocení porovnávaných algoritmů pro ochranu souborů

Při porovnání kryptografických metod jsme zjistili, že nejlépe si vede AES, který má rychlost šifrování souborů 330 MB/s, dešifrování souborů 396 MB/s a průměrnou dobu šifrování souborů 3,1 vteřiny a dešifrování 2,48 vteřiny. Na druhém místě se umístil algoritmus Camellia s rychlostí šifrování souborů 310 MB/s dešifrování souborů 372 MB/s a s průměrnou dobou šifrování souborů 3,3 vteřiny, dešifrování 2,64 vteřiny. Tyto dva typy algoritmů při porovnávání vyšli jako nejvhodnější metody pro ochranu souboru.

S nejmenší rychlosti šifrování a dešifrování souborů a zároveň s nejmenší průměrnou dobou šifrování souborů se stal ruský algoritmus Kuznyechik. Tento Algoritmus bych rozhodně nedoporučoval.

5 Výsledky a diskuze

5.1 Hodnocení konečného porovnání

Výsledkem této práce je porovnání algoritmů, kteří můžé pomoct čtenáři při výběru algoritmu pro šifrování souborů.

5.1.1 Přínosy

Velkým kladem tohoto porovnání, je ukáka, že s pomocí již existujících programů a základní znalostí kryptografie lze poměrně snadno zabezpečit soubory. V dnešní době, kdy jsou stále intenzivnější útoky hackerů, za účelem získání citlivých souborů, je toto zjištění velmi přínosné pro společnost.

6 Závěr

Prezentované výsledky porovnání ukázaly, že na základě výše uvedených testů, AES má lepší výkonnost než ostatní běžně používané šifrovací algoritmy. Vzhledem k tomu, že AES dosud nemá žádné známé bezpečnostní chyby, stává se vynikajícím kandidátem při zvážení pozice jako standardní šifrovací algoritmus. Algoritmus Kuznyechik vykázal nevyhovující výsledky ve srovnání s ostatními algoritmy, proto se stáva nejméně vhodným algoritmem. Toto tvrzení je podloženo nejhoršími výsledky v daných testech, které dokázali že kuznyechik vyžaduje nejvyšší výpočetní výkon.

Na základě výše uvedených testů jsem vyvodil závěr, že existuje přímý vztah mezi počtem kombinací algoritmů a výkonností vašeho procesoru. Tím pádem, pokud by jsté se chtěli vyhnout přetížení vašeho procesu, musíte zvolit nejjednodušší algoritmus.

Na základě těchto faktů, bych vám doporučil zvážit silný procesor, v případč že plánujete šifrovat velké soubory.

7 Seznam použitých zdrojů

(1) PARTYKA, Tatiana A., POPOV, Igor I. Information security. Publishing Infa-M, c2005. ISBN: 5-8199-0060-X.

(2) BRUEN, Aiden A., Mario FORCINITO. Cryptography, information theory. Hoboken, N.J.: Wiley-Interscience, c2005. ISBN 04-716-5317-9.

(3) TARASYUK, M.V. Secure information technology. Publishing "Solon-Press", c2004. ISBN 5-98003-143-X.

(4) RYABKO, B.Y., N. FIONOV. Basics of modern cryptography for specialists in information technologies. Publishing "Scientific world ", c2004.
ISBN 978-5-89176-233-6.

(5) IVANOV, Vladimir. Cryptography and encryption. [Online], 23.4.2017. [cit. 2017-12-11] Dostupné z: < https://habrahabr.ru/company/yandex/blog/324866/ >

(6) JESSICA, J BENZ. PGP: A Hybrid Solution. [Online], SANS Institute, 28. 6 2001. [cit. 2017-01-25.] Dostupné z: < https://clck.ru/CuXej > .

(7) SCHNEIER, B. FERGUSON, N. Practical cryptography. New York: Wiley, 2003. ISBN 047122894.

(8) MENEZES, A. J., Paul C. VAN OORSCHOT a Scott A. VANSTONE. Handbook of applied cryptography. Boca Raton: CRC Press, c1997. ISBN 08-493-8523-7.

(9) SCHNEIER, Bruce. Applied cryptography. Protocols, algorithms.
Publishing "Solon-Press", c2004. ISBN 5-98003-143-X.

(10) GATCHIN, Y., KOROBEINIKOV A. G. Bases of cryptographic algorithms. [Online] Spbgitmo(TU), 2002 [cit. 2017-12-12.]
Dostupné z: < http://www.ict.edu.ru/ft/001707/oka_2.pdf >

(11) MATT, J., B. ROBSHAW. Stream Ciphers [Online] Laboratories, 1995 [cit. 2017-11-12.] Dostupné z: < http://qoo.by/43IR >

(12) SALOMAA, A. The public key cryptography. Publishing "Scientific world ", c1995. ISBN 5-03-001991-X.

(13) WIRTH, Niklaus. Algorithms and data structures. Publishing "World ", c1989. ISBN 5-03-001045-9.

(14) MAO, A. Modern cryptography. Publishing "Williams ", c2005. ISBN 978-5-8459-0847-6.

(15) SCHNEIER, Bruce. N., FERGUSON., J. KELSEY., D. WHITNING. « A Twofish Retreat: Related-Key Attacks Against Reduced-Round Twofish » [Online] Twofish Technical Report #6, 2000 [cit. 2017-11-12.] Dostupné z: < https://www.schneier.com/academic/paperfiles/paper-twofish-related.pdfR >

(16) YUEN, P.K. A flexible and adabtive block cipher: Blowfish. Publishing " Pearson Education Canada ", c2005. ISBN 978-0-321-26333-9.

(17) MATSUI, M. A Description of the Camellia Encryption Algorithm . [Online] Network Working Group, 2004 [cit. 2017-11- 9.] Dostupné z: < https://tools.ietf.org/html/rfc3713 >

(18) ANDERSON, R., BIHAM, E., KNUDSEN, L. Serpent: A Proposal for the Advanced Encryption Standard. [Online] University of Bergen, 2012 [cit. 2017-11- 9.] Dostupné z: < http://www.cl.cam.ac.uk/~rja14/Papers/serpent.pdf >

(19) BIRYKOV, A., PERRIN, L., UDOVENKO, A. Reverse-Engineering the S-Box of Streebog, Kuznyechik and STRIBOBr1. [Online] University of Luxembourg, 2016 [cit. 2017-11- 9.] Dostupné z: < https://eprint.iacr.org/2016/071.pdf >

(20) KLEIN, A. Attacks on the RC4 stream cipher. [Online] Andreas Klein, 2006 [cit. 2017-11- 9.] Dostupné z: < http://cage.ugent.be/~klein/RC4/RC4-en.ps >

(21) RIVEST, R. RFC 1321, The MD5 Message-Digest Algorithm. [Online] MIT Laboratory for Computer Science and RSA Data Security, 1992 [cit. 2017-11- 9.]
Dostupné z : < https://tools.ietf.org/html/rfc1321 >

(22) SCHNEIER, Bruce. N. FERGUSON . Practical Cryptography: Designing and Implementing Secure Cryptographic Systems. Publishing " M. Dialectics,", c2004.
ISBN 5-8459-0733-0

(23) DERBEZ, Patrick a FOUQUE, Pierre Alain. Exhausting Demirci-Selçuk Meet-in-theMiddle Attacks against Reduced-Round AES. [Online] École Normale Supérieure, France, 13. 3 2013. [cit. 2017-12-25] Dostupné z: < http://www.di.ens.fr/~fouque/pub/fse13b.pdf >

(24) HALDERMAN Alex, Hidden Operating system. [Online], 4.1.2015. [cit. 2017-12-11]
Dostupné z: < https://www.veracrypt.fr/en/Home.html >