UNIVERZITA PARDUBICE
Fakulta elektrotechniky a informatiky

Informační bezpečnost a kryptologie

Lukáš Marian

Bakalářská práce
2012

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: **2011/2012**

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Lukáš Marian**
Osobní číslo:	**I09189**
Studijní program:	**B2646 Informační technologie**
Studijní obor:	**Informační technologie**
Název tématu:	**Informační bezpečnost a kryptologie**
Zadávající katedra:	**Katedra informačních technologií**

Zásady pro vypracování:

Cílem práce je popsat kryptologii jako jednu z metod informační bezpečnosti. V teoretické části autor vyjde z prostudované odborné literatury a na základě získaných znalostí souhrně popíše požadavky na model konvenčního kryptosystému včetně charakteristiky tohoto systému. V práci bude mimojiné podrobně popsán princip veřejného klíče a možnosti jeho použití. Dále budou popsány sysmetrické a asymetrické šifry a hashovací funkce a jejich vazby. Praktickým výstupem bakalářské práce budou laboratorní úlohy pro demonstraci využití vybranných kryptografických technik.

Předpokádaný rozsah práce 35-45 stran.

Rozsah grafických prací:

Rozsah pracovní zprávy:

Forma zpracování bakalářské práce: **tištěná/elektronická**

Seznam odborné literatury:

Menezes A, Vanstone S, van Oorschot P., Handbook of Applied Cryptography, CRC Press, 1996, volně ke stažení na http: //www.cacr.math.uwaterloo.ca/hac/
Levický D., Kryptografia v informačnej bezpečnosti, elfa, 2005,ISBN:80-8086-022-X
Mao W., Modern Cryptography - Theory & Practice, Prentice-Hall, 2004, ISBN: 0-13-066943-1

Vedoucí bakalářské práce: **Ing. Soňa Neradová**
Katedra softwarových technologií

Datum zadání bakalářské práce: **16. prosince 2011**
Termín odevzdání bakalářské práce: **11. května 2012**

Univerzita Pardubice Česká republika

L.S.

prof. Ing. Simeon Karamazov, Dr.
děkan

Ing. Lukáš Čegan, Ph.D.
vedoucí katedry

V Pardubicích dne 30. března 2012

Prohlášení autora

Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 08. 05. 2012 Lukáš Marian

Poděkování

Chtěl bych vyjádřit poděkování Ing. Soně Neradové za poskytování studijních materiálů a připomínky v průběhu vypracování této práce. Dále bych chtěl poděkovat své rodině za finanční a psychickou podporu.

Anotace

Bakalářská práce se zabývá studiem kryptologie. Představen je model konvenčního kryptosystému, jeho charakteristika a základní metody šifrování. Popsány jsou principy symetrického a asymetrického šifrování, hašovacích funkcí, digitálního podpisu a proces hodnocení moderního kryptosystému. Nastíněny jsou základní možnosti kryptoanalytických útoků a možnosti obrany proti těmto útokům. Praktická část je rozdělena do dvou laboratorních úloh. První se věnuje programu PGP, který se používá pro šifrování emailové komunikace, je zde popsána a předvedena instalace a konfigurace na platformách Windows a Linux. V druhé laboratorní úloze je předvedena a popsána konfigurace autentizační metody CHAP na sériové lince mezi dvěma směrovači.

Klíčová slova

Kryptologie, Kryptografie, Kryptoanalýza Šifrování, Kryptosystém, PGP, PPP, CHAP

Title

Information security and kryptology

Annotation

The theme of this bachelor study thesis is study of cryptology. There is introduced model of conventional cryptographic system, its characteristics, and basic methods of encryption in this thesis. Principles of symmetric and asymmetric encryption, hash functions, digital signature and the process of evaluation modern cryptographic systems are described int this thesis. The practical part is divided into two laboratory tasks. The first task describes program PGP, which is program for encryption of email communication and in this thesis is presented installation and configuration on Windows and Linux platforms. Configuration of authentication method CHAP on the serial line between two routers is presented in the second task.

Keywords

Cryptology, cryptography, cryptanalysis, encryption, PGP, PPP, CHAP

Obsah

Seznam zkratek

AES	Advanced Encryption Standard
CAST	Carlisle Adams and Stafford Taveres algorithm
CHAP	Challenge-Handshake Authentication
CLI	Command-Line Interface
DCE	Data Communications Equipment
DES	Data Encryption Standard
DSA	Digital Signature Algorithm
DTE	Data Terminal Equipment
D-H	Diffie-Hellman algoritmus
HDLC	High-level Data Link Control
HP	Hewlett Packard
IDEA	International Data Encryption Algorithm
IP	Internet Protocol
IPX	Internet Protocol Exchange
IS	Informační systém
ISDN	Integrated Services Digital Network
ISO	International Organization for Standardization
IT	Informační technologie
MD5	Message-Digest algoritmus verze 5
NIC	Network Interface Controller
N/A	Not Applicable
OSI	Open Systems Interconnection
PC	Personal Computer
PGP	Pretty Good Privacy
PPP	Point-to-Point Protocol
RSA	Rivest-Shamir-Adleman algorithm

Seznam Obrázků

Seznam tabulek

Úvod

Po přečtení této práce získá čtenář základní představu o problematice utajování citlivých informací v prostředí moderních informačních systémů. Představen je konvenční kryptografický model, který velmi dlouho ovlivňoval a doposud stále ovlivňuje vývoj kryptologie. Podrobná charakteristika popisuje také požadavky, které musí splňovat moderní kryptosystém. Dále jsou popsány metody, na kterých jsou založené dnešní šifrovací systémy a také způsoby, jakými je možné na kryptosystémy útočit. Zmíněny jsou základní metody obrany proti kryptoanalytickým útokům. V rámci popisu symetrického šifrování je dán důraz na metody sdílení tajného klíče. V kapitole Advanced Encryption Standard se lze dozvědět, jak probíhal výběr nejnovějšího standardního kryptosystému, který byl následně adoptován pod označením AES a jaká kritéria byla hodnocena. V závěru teoretické části je popsán princip hašovacích funkcí a šifrování prostřednictvím veřejného klíče. Jako možnost využití algoritmů s veřejným klíčem je uveden princip digitálního podpisu. V závěru je představen algoritmus RSA a jeho praktické předvedení.

Praktická část je rozdělena do dvou laboratorních prací. V první práci je představen populární program Pretty Good Privacy, který je široce používán pro šifrování a dešifrování emailové komunikace. Podporována je i technologie elektronického podpisu a jeho ověření. Předvedena je instalace a konfigurace programu PGP na platformách Windows i Linux. Jednotlivé kroky konfigurace jsou přehledně popsány a vysvětleny. Předvedena je tvorba soukromého a veřejného klíče, zanesení veřejného klíče na jeden z klíčových serverů a postup ověřování podepsané zprávy.

První laboratorní práce se věnuje možnostem kryptografických technik, které může využít koncový uživatel informačních sítí. Oproti tomu se druhá část věnuje samotnému srdci informačních sítí, a tím je komunikace mezi jednotlivými směrovači. Směrovače, totiž kromě uživatelských dat, také přeposílají směrovací tabulky, informace o stavu linky, informace potřebné k autentizaci a mnohé další. Tyto informace jsou označovány jako režie a prozrazení takových informací lze stejně efektivně zneužít jako data koncových uživatelů. Proto i směrovače využívají kryptografických technik. Pro demonstraci těchto technik je předvedena konfigurace sériové linky mezi dvěma směrovači, využívající standardní Point-to-Point Protokol spolu s autentizační metodou Challenge-Handshake Autentication Protokol. Tato autentizační metoda využívá hašovací funkci MD5, její využití v tomto procesu je před samotným předvedením konfigurace podrobně popsáno.

1 Informační bezpečnost a kryptologie

1.1 Terminologie

- **Entita:** Komunikační prvek, může jím být konkrétní osoba nebo jakýkoliv síťový uzel (tiskárna, směrovač, …).
- **Zdroj:** Může se jednat o paměťové místo, přidělený čas procesoru, vzdálené zařízení poskytující služby nebo se může jednat i o zdroje finanční.
- **Informační technologie:** Veškerá technika zabývající se zpracováním informací. Zejména se jedná o výpočetní a komunikační techniku a její programové vybavení.
- **Informační systém:** Identifikovatelný funkční celek, zabezpečující cílevědomé a systematické shromažďování, zpracovávání a zpřístupňování informací.
- **Bezpečnost informačního systému:** Stav informačního systému, kdy rizika, jímž je vystaven, jsou snížena na přijatelnou úroveň na základě vhodných bezpečnostních opatření.
- **Analýza rizik:** Činnost prováděná v souladu s normou ČSN EN 19011, zaměřená na odhad ztrát, které mohou vzniknout působením hrozeb na IS, získání přehledu o závažnosti jednotlivých hrozeb a zranitelných míst hodnoceného IS.
- **Bezpečnostní audit:** Činnost provedení kontroly IS se zaměřením na jeho bezpečnost. Audit může být proveden pomocí interních pracovníků nebo pracovníků specializované certifikační organizace.
- **Bezpečnostní incident:** Událost, která má nebo může mít negativní dopad na bezpečnost IS. Příčinou je velmi často úmyslná nebo neúmyslná činnost člověka.
- **Citlivá informace:** Informace, jejíž ztráta, chybné použití, neoprávněná modifikace nebo zneužití neoprávněnou osobou může způsobit škodu.
- **Sociální inženýrství:** Neoprávněné získávání důvěrných informací (například technické parametry vyvíjených zařízení, adresy osob, přístupová hesla, obchodní záměry a podobně) na základě komunikace s lidmi. Jedná se například o telefonování pod cizím jménem.
- **Aktivum:** Nehmotný nebo hmotný majetek mající v IS určitou hodnotu.
- **Hodnota aktiva:** Vychází z objektivního vyjádření vnímané ceny nebo subjektivním ohodnocením důležitosti aktiva, případně je kombinací obou procesů.

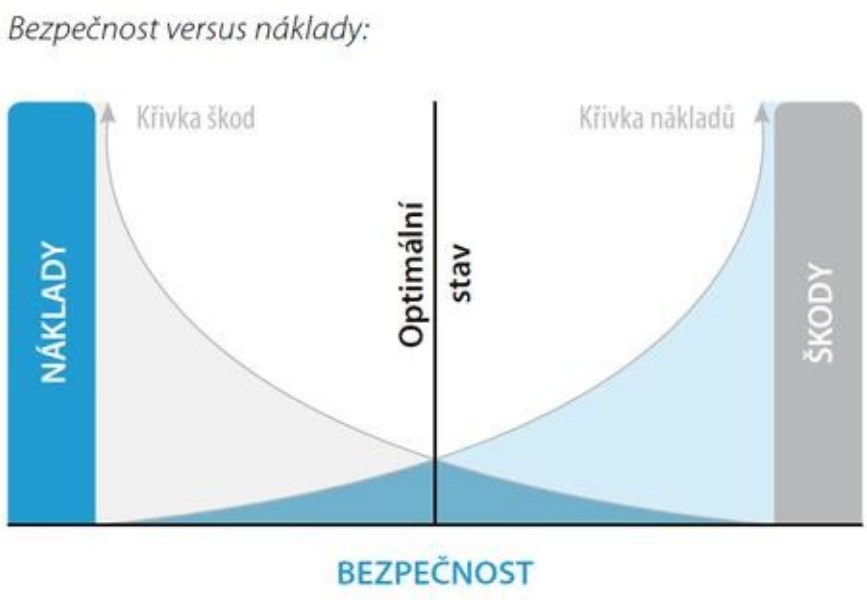
Části tohoto seznamu jsou čerpány z článku nezávislého odborného on-line magazínu (7,Terminologie).

1.2 Informační bezpečnost

Požadavky na informační bezpečnost se vyvíjejí spolu se síťovými technologiemi. Ty byly nejprve určeny pro výzkumné účely. Tehdy se jednalo prakticky o sdílení tiskáren a posílání emailů. Zvýšený zájem o informační bezpečnost nastal až později, kdy se začalo uvažovat o nasazení síťových technologií pro firmy a veřejnost. Dnes využívá síťové technologie více než milióny lidí pro účely bankovní, obchodní nebo vyplnění daňových přiznání (4, s. 721).

Pojem **informační bezpečnost** je chápán jako komplexní sada pravidel a cílů, které je třeba dodržet nebo splnit. Je třeba si uvědomit, že pro každou situaci může být tato sada značně odlišná. Ačkoliv se název tohoto pojmu může zdát příliš obecný, ve skutečnosti jde o velmi komplexní, specializovaný systém pro ochranu dat.

Často nelze bezpečnostní cíle splnit pouze vývojem dokonalejších algoritmů nebo bezpečnostních protokolů. Někdy je potřeba nastavit i okolní systém, ve kterém se budou tyto bezpečnostní cíle plnit. Jde například o zákony státu, které musí jasně označit činnost, kterou lze považovat za kriminální čin (8, s. 2).



Obr. 1 - Optimální míra bezpečnosti, zdroj: (14)

Je třeba si uvědomit, že bezpečnost není produkt, který lze jednorázově zakoupit. Jde o trvalý proces, na kterém se musí neustále pracovat. Řešení musí být komplexní, odolné a efektivní. Je také potřeba zahrnout ekonomickou situaci. Jak je vidět na Obr. 1, příliš nízké náklady na bezpečnost mohou vést k vysokému finančnímu vyčíslení způsobených škod. Stejně tak příliš vysoké náklady nezajistí absolutní bezpečí a míra investice nemusí značně odpovídat potenciálním škodám. Je třeba hledat optimální stav. Mnohdy se však bezpečnost bagatelizuje, protože nepřináší viditelný finanční zisk (14, Informační Bezpečnost).

1.2.1 Cíle informační bezpečnosti

Na ochraně dat se podílí několik entit a bezpečnostních prvků. Každý z nich může pracovat na odlišném stupni procesu ochrany dat. Za jakékoliv situace by měl mít každý jistotu, že byly naplněny všechny kroky ke splnění bezpečnostních cílů (8, s. 2). Některé z bezpečnostních cílů a mechanismů jsou uvedeny v následujícím seznamu[1].

- **Soukromí a důvěryhodnost:** Zaměřuje se na ochranu obsahu předávaných zpráv. Přístup k informacím mají pouze oprávněné osoby (12, CCNA4, 6.3.4).
- **Integrita dat:** Zabezpečuje, že příjemce obdržel data v nezměněné podobě. Jinými slovy, v průběhu přenosu nebylo s daty neoprávněně manipulováno. Konkrétně se jedná o vkládání, mazání a změnu obsahu zprávy.
- **Ověřování entit:** Zajišťuje, že zprávy putují od očekávaného odesílatele k očekávanému příjemci. Ověřování entit musí také zajistit, že komunikace není narušena třetí stranou. Ta se může například pro první stranu představit jako strana druhá (a naopak) a dále komunikaci zprostředkovávat (3, s. 8).
- **Autorizace:** Je to proces, na základě kterého je možné vydat oficiální schválení k provádění konkrétní činnosti nebo k právu vydávat se za někoho.
- **Řízení přístupu:** Omezení nebo řízení přístupu k informacím na základě identity nebo role (3, s. 8).
- **Vlastnictví:** Entita, která se stane vlastníkem, má legální právo na užívání nebo přenášení určitých zdrojů.
- **Anonymita:** Vztahuje se ke konkrétnímu prvku komunikačního procesu. Úkolem anonymity je utajit identitu tohoto prvku.
- **Svědectví:** Ověření vytvoření nebo existence informace na základě svědectví třetí entity (svědkem nesmí být tvůrce).
- **Certifikace:** Potvrzení předávaných informací od důvěryhodné osoby.
- **Časové razítko:** Popisuje časový okamžik stvoření, odeslání, nebo změny dat. Nadále je časové razítko neopomenutelnou součástí těchto dat.
- **Nepopiratelnost:** Pokud se komunikující strany předem dohodnou na některých akcích či závazcích, není možné je pak ignorovat.
- **Odvolatelnost:** Možnost odebrání autorizace nebo certifikace.

1Seznam cílů informační bezpečnosti je čerpán ze zdroje (8, s. 3).

1.3 Kryptologie a její historie

Kryptologie je vědní obor, který pojednává o kryptografii a kryptoanalýze. Někdy je tento vědní obor označován jako věda o kódech a šifrách. Lépe řečeno je to věda o mechanismech pro utajování citlivých informací. Těmito mechanismy jsou citlivé zprávy kódovány neboli šifrovány. Typicky se jedná o záměnu znaků nebo prohazování pořadí znaků. Příjemce zprávy musí vědět podrobnosti tohoto šifrovacího mechanismu, aby mohl být schopen zprávu dekódovat neboli dešifrovat, a tím získat původní zprávu. Pokud se zašifrovaná zpráva dostane do rukou osoby, která nezná dostatečné podrobnosti o šifrovacím mechanismu, může se pokusit o jeho luštění. Proces luštění má stejný záměr jako proces dešifrování, získat původní text. Odlišnost je v tom, že luštitel bývá typicky nezamýšlený příjemce zprávy. Luštitel se tedy může pokusit o prolomení šifrovacího mechanismu. To se mu může povést s různým výsledkem. Buď dekóduje pouze část zprávy nebo celou zprávu. Přestože se mu podaří dekódovat celou zprávu, nemusí to nutně znamenat, že dokáže dekódovat všechny zprávy šifrované daným šifrovacím mechanismem. V případě prolomení šifrovacího mechanismu jsou následky katastrofické, luštitel je schopen dešifrovat jakoukoliv zprávu šifrovanou tímto mechanismem.

Začátky kryptologie je možné stopovat daleko do historie. Jako počátek se uvádí vyrytí hieroglyfů do kamene ve městě Menet Khufu. V tomto případě však nešlo o skrývání textu, nýbrž o vznešenější formu zápisu za účelem projevení úcty. Jednalo se ale o záměrnou záměnu znaků a tedy o jeden ze základních principů šifrování (1, Starověk). Chceme-li být objektivní, musíme si uvědomit, že zárodky kryptologie se projevovaly ještě mnohem dříve. Vše souvisí s lidskou psychologií a potřebou bezpečí. Člověk už od pradávna má potřebu uchovávat tajemství za účelem vlastní bezpečnosti.

Jak už bylo řečeno, záměna a prohazování pořadí znaků jsou základní techniky šifrování. Existují ale i „kreativnější“ způsoby jak utajit informace. Například ve starověké Číně si poslové zprávu napsali na papír, ten zmuchlali a následně ho potáhli voskem a spolkli. Herodotos zase zmiňuje techniku, kdy byl otrok oholen a zpráva mu byla vytetována na hlavu. Jakmile vlasy dorostly, posel byl vyslán (1, Starověké Řecko).

Jak doba plynula, začaly se využívat nejrůznější pomůcky a stroje. Například ve starověké Spartě byl v pátém století před naším letopočtem použit první vojenský kryptosystém (viz Obr. 2 na následující straně). *„Sparťané používali hůl přesného rozměru, na který navíjeli proužek papyru, kůže nebo pergamenu. Zpráva byla napsána po délce hole, následně byl proužek odvinut a odeslán. Jednalo se o prohazování pořadí znaků a dešifrování bylo možné provést pouze pomocí hole shodné tloušťky.“* (1, Starověké Řecko)

Jako nejznámější zástupce kryptografického šifrovacího stroje je Enigma. Tento rotorový stroj používaly Německé vojenské síly za druhé světové války. Prakticky šlo o psací stroj, který byl napojen na šifrovací mechanismus. Ten se skládal z několika kotoučů, skrz které procházel signál. Kotouče se otáčely různou rychlostí, čímž se zvyšoval počet možných kombinací, které stěžovali kryptoanalytikům práci. (15, Enigma)

2 Kryptografie

Je věda o matematických technikách se zaměřením na cíle informační bezpečnosti, jako jsou důvěryhodnost, integrita dat, autentizace a autorizace (8, s. 4). Studium zahrnuje šifrovací algoritmy, kryptografické nástroje, hardwarové implementace šifrovacích algoritmů, kryptografické protokoly apod. (15, Nezbytná teorie na začátek).

Obr. 2 - Historická kryptografická pomůcka, zdroj: (20)

2.1 Rozdíl mezi šifrou a kódem

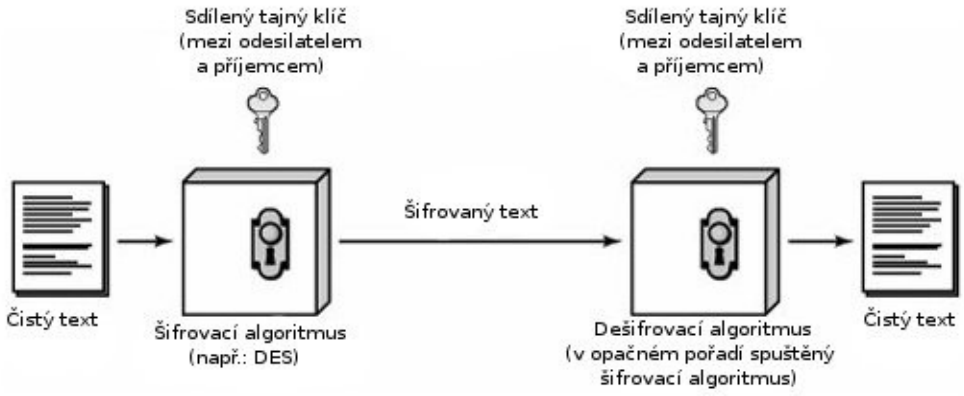
Významový rozdíl těchto dvou výrazů se v odborné literatuře do jisté míry stírá. Existují však poměrně přesné definice. Šifrou (ang. cipher) je myšlena *„transformace typu znak-za-znak nebo bit-za-bit a to bez ohledu na lingvistickou strukturu zprávy“*. Kódem (ang. code) je myšlena *„náhrada jednotlivých slov za jiná slova nebo symboly“*. Dnes se již prakticky v moderních kryptosystémech využívají výhradně šifry. Kódy však byly historicky významnou součástí kryptografie (4, s. 724).

Jako nejúspěšnější kryptosystém založený na kódování se označuje kód Navajo. Využíval se za druhé světové války v Pacifiku. Šlo vlastně o indiánský jazyk, který je *„velmi tónový, mimořádně komplexní, nemá psanou formu a hlavně o něm žádný Japonec neměl ani tušení.“* Technika šifrování pak byla jednoduše založena na rozhovoru dvou osob v tomto jazyce. Kód vznikl přidružením amerických výrazů ke slovům v jazyce Navajo. Tento kód se nepodařilo Japoncům za dobu války prolomit (4, s. 724- 725).

2.2 Model konvenčního kryptosystému

V počátcích rozmachu kryptografie, kdy ještě nebyl dostupný dostatečný výpočetní výkon, se šifrování a dešifrování opíralo o jednodušší a technicky nenáročné postupy. Důvodů bylo hned několik. O šifrování a dešifrování se starali pověření lidé, kteří si techniku museli zapamatovat a být schopni ji v co nejkratším čase provádět. Používali nejrůznější pomůcky nebo jednoduché stroje. To však mělo za následek, že šifry nebyly dostatečně komplexní a byly příliš náchylné na prolomení. V případě prolomení vznikl požadavek na změnu šifrovací techniky, a tím pádem i na přeučení všech lidí angažovaných v tomto procesu. V ideálním případě měl být přechod z jedné šifrovací techniky na druhou instantní. To vedlo k rozmachu konvenčního modelu (4, s. 725).

2.3 Zjednodušený symetrický model



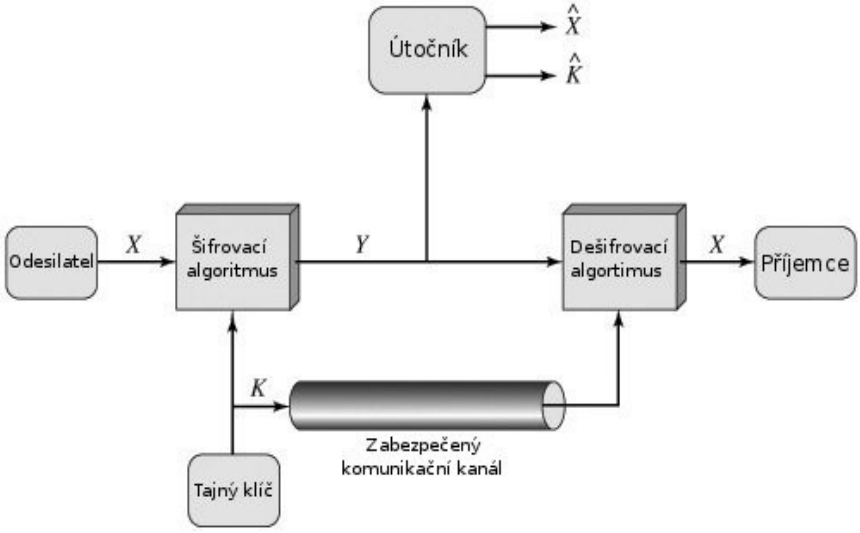
Obr. 3 - Zjednodušený model konvenčního kryptosystému

- **Čistý text:** Jde o srozumitelnou zprávu nebo data. Čistý text je vkládán jako vstup do šifrovacího algoritmu a následně je získáván jako výstup z dešifrovacího algoritmu.
- **Šifrovací algoritmus:** Provádí různorodé kryptografické metody, čímž převádí čistý text na šifrovaný text. Pro převod je vyžadován ještě druhý vstup, kterým je klíč (3, s. 24).
- **Tajný klíč:** Je zcela nezávislí na čistém textu a určuje parametry pro metody šifrovacího algoritmu.
- **Šifrovaný text:** Je výstupem z šifrovacího algoritmu. Jakákoliv změna v klíči nebo čistém textu radikálně změní vzhled a smysl šifrovaného textu. Šifrovaný text má představovat nesmyslný řetězec dat (3, s. 25).
- **Dešifrovací algoritmus:** V podstatě jde o šifrovací algoritmus, který je prováděn v opačném pořadí. Vstupem je šifrovaný text a klíč, výstupem je čistý text. Pakliže jde o symetrický kryptosystém, používá se stejný klíč jako u šifrovacího algoritmu.

2.3.1 Požadavky na konvenční kryptosystém

1) *„Potřebujeme silný šifrovací algoritmus. Minimálně tak silný, aby útočník, který zná algoritmus a má přístup k několika šifrovaným textům, nebyl schopný dešifrovat šifrovaný text nebo odhadnout hodnotu klíče. Ideálně by měl být algoritmus odolný i proti útokům, kdy útočník vlastní sadu čistých textů a k nim jejich šifrované podoby.“* (3, s. 25)

2) Odesilatel i příjemce musí získat kopii klíče a držet jí v tajnosti. K předání klíče dochází buď před samotným zahájením komunikace nebo pomocí jiného zabezpečeného komunikačního kanálu. Tato problematika bude popsána v kap. 4.1.



Obr. 4 - Model konvenčního kryptosystému, zdroj (3)

„Mezi požadavky se obvykle nezařazuje utajení šifrovacích a dešifrovacích algoritmů. Namísto toho se klade větší důraz na utajení klíče.“ Útočník tedy může na základě šifrovaného textu a znalosti algoritmu tipovat hodnotu klíče, a tím získávat původní čistý text. Později však bude vysvětleno, že tento způsob je značně neefektivní. Tento princip[1] navíc přináší velkou výhodu, je možné bez obav implementovat čipy, které urychlí šifrovací proces (3, s. 25).

Pokud tedy změníme šifrovací algoritmus, jedná se o velkou změnu. Pokud ale změníme používaný klíč, jediné co je potřeba zajistit je to, aby klíč bezpečně obdržela i druhá strana. To znamená, že klíč můžeme měnit jednoduše a velmi rychle (4, s. 726). Tato schopnost je nazývána jako agilnost klíče.

1 **Kerckhoffův princip:** „*Všechny algoritmy musí být veřejné, pouze klíče musí být tajné“* (4, s. 726).

2.4 Charakteristika kryptosystému

Kryptosystémy můžeme dělit na základně několika vlastností:

1. **Typ operací používaných k převodu čistého textu do šifrovaného a naopak:** Všechny šifrovací algoritmy jsou založeny na dvou obecných principech: *substituce*, kde každý element čistého textu (bit, písmeno, skupina bitů nebo znaků) je zaměněn za jiný element, a *transpozice*, kde je pořadí elementů zpřeházeno. Základem je, že každá operace je vratná (nedochází ke ztrátě informací). Většina systémů tyto metody používá obě i vícekrát a v různém pořadí (3, s. 27).

2. **Počet použitých klíčů:** Kryptosystémy se dělí na symetrické a asymetrické. V případě kdy šifrovací i dešifrovací algoritmus používají totožný klíč, se jedná o symetrické neboli konvenční šifrování. Pakliže jsou klíče dva, veřejný klíč pro šifrování a tajný klíč pro dešifrování, jedná se o šifrování asymetrické.

3. **Typ zpracování čistého textu:** Používá se šifrování proudové, kde šifrujeme jeden element po druhém, nebo šifrování blokové, kde zpracováváme čistý text v blocích o pevně daném objemu (např.: 128 bitů). Oblíbené jsou hlavně blokové šifry, jelikož je jednodušší definovat komplexnější algoritmus pro skupinu elementů o pevné velikosti. Proudové šifry řeší každý jednotlivý element, není nutné čekat na načtení dostatečného počtu znaků a nemusíme je ukládat do paměti (3, s. 27).

2.4.1 Substituční metoda šifrování

Jednoduše řečeno jde o nahrazování elementů čistého textu za elementy jiné. Elementy je možné přemapovat metodou **znak za znak** nebo metodou **znak za skupinu znaků**. Klíčem se pak označují právě tyto relace. Nejčastěji se pro vysvětlení substituční šifrovací metody uvádí **Caesarova šifra**, pomocí které Caesar utajoval své diplomatické depeše (1, Starověký Řím).

čistý text:	A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
tajný klíč:	X Y Z A B C D E F G H I J K L M N O P Q R S T U V W

Mnemotechnická pomůcka pak praví: „Nahraď písmeno skutečné zprávy písmenem, které je v abecedě o tři pozice dříve.“

příklad zprávy:	TOTO JSOU CITLIVE INFORMACE
šifrovaná podoba:	QLQL GPLR ZFQI FSBF KCLO JXZB

Aby nebyl tajný klíč jednoduše odhadnutelný, právě díky svému charakteru pořadí písmen, je možné použít další pomůcku. Zvolí se tajné heslo (např.: KDO JE TAM) které bude uvádět začátek tajného klíče, ostatní chybějící písmena se pak doplní podle pořadí v abecedě. Výsledný klíč by pak vypadal následovně.

čistý text:	A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
tajný klíč:	K D O J E T A M B C F G H I L N P Q R S U V W X Y Z

Je potřeba si uvědomit, že v této fázi se nemusí skončit. Tuto šifrovací metodu je možné použít opakovaně nebo se pokaždé použije metoda jiná. Je však nutné zapamatovat si pořadí těchto metod.

2.4.2 Transpoziční metoda šifrování

Tato metoda je založena na změně pořadí elementů čistého textu. Jak bylo ukázáno u substituční metody, i zde je možné použít různých mnemotechnických pomůcek. Například je možné zvolit tajné heslo, KDOJETAM, které se napíše na první řádek. Jednotlivé znaky hesla budou uvazovat sloupce. Tyto sloupce jsou na druhém řádku číselně ohodnoceny. Číslo 1 přiřadíme sloupci s nejnižším písmenem v abecedě, které je obsaženo v pomocném hesle, v tomto případě písmeno A. Číslo 2 přiřadíme sloupci s druhým nejnižším písmenem v abecedě, v tomto případě D. Takto se pokračuje do té doby, než jsou ohodnoceny všechny sloupce. Dále je potřeba rozdělit čistý text do bloků o délce počtu znaků pomocného hesla, tedy bloky o 8 znacích. Bloky se vypíší do jednotlivých řádků pod sebe. Šifrovaný text je vytvořen vypsáním jednotlivých sloupců do řádku za sebe a to v pořadí uvedeném na druhém řádku od 1 do 8.

čistý text:	UTOK PROVEDTE ZA ROZBRESKU
pomocné heslo:	K D O J E T A M
tajný klíč:	5 2 7 4 3 8 1 6
čistý text:	U T O K P R O V
	E D E T E Z A R
	O Z B R E S K U
šifrovaný text:	OAK TDZ PEE KTR UEO VRU OEB RZS

Pokud se používá tato metoda sama o sobě, je nutné si uvědomit, že tato metoda nemá žádný vliv na frekvenci znaků. Pokud bychom spočítali četnosti jednotlivých znaků, vyjde nám shodný výsledek před i po šifrování touto metodou. Relativní frekvence znaků bude blíže popsána v kapitole 3.2.

2.4.3 Steganografie

„Slovo "steganografie" pochází z řečtiny a znamená doslova "skryté písmo" (stegos = střecha, kryt; graphos = psaní, vyobrazování). Tento obor zahrnuje spoustu metod tajné komunikace skrývajících samotnou skutečnost, že dochází k zasílání zprávy. Utajuje samotnou existenci komunikace, čímž zvyšuje stupeň jejího zabezpečení. Patří sem neviditelné inkousty, mikrotečky, využití uspořádání znaků (odlišné od kryptografických metod permutace a substituce), digitální podpisy, skryté kanály a širokopásmová komunikace. Tajná zpráva je ukryta v jiné informaci (tzv. krycím objektu) tak, že není zřejmé, že by krycí objekt vůbec obsahoval nějakou další zprávu kromě svého zjevného obsahu. Krycím objektem může být text, obrázek, obrazový nebo zvukový záznam.“ (2, Definice Steganografie)

V definici není zmíněno samotné šifrování zprávy. Jde o to, že obecná steganografie zcela závisí na ukrytí skutečnosti, že ke komunikaci vůbec dochází, a dále není zpráva nijak zvlášť šifrována. Pokud tedy dojde k prozrazení této techniky, jsou kriticky ohroženy veškeré zprávy odeslané v minulosti i budoucnosti. V praxi se tak Steganografie kombinuje s dalšími technikami a tím se elegantně využívá výhod nenápadnosti.

Pokud nejde o to, skrývat samotné uskutečnění komunikace, umožňuje steganografie odeslat zprávu klasickým způsobem. Zpráva tak bude přitahovat menší pozornost. Vyšší pozornost přitahují zprávy šifrované, u kterých se očekává, že nesou důležité informace. (3, s. 49)

Nevýhodou steganografie je nadměrný přenos dat. Pro odeslání každé zprávy je nutné vygenerovat fiktivní data, kterými je kryta skutečná komunikace. Pokud je objem fiktivních dat nedostatečný, může vzniknout podezření, že zpráva obsahuje skryté informace. Proto je nutné generovat velký objem fiktivních dat. (3, s. 49)

3 Kryptoanalýza

S vývojem kryptografie se přirozeně vyvíjela i kryptoanalýza. Ta se zabývá studiem matematických metod, které je možné využít k prolomení nebo ověření odolnosti konkrétního kryptosystému. Tradiční metody jsou založené na náhodném výběru klíče nebo na omezení efektivní délky klíče.

V poslední době se však objevuje nová generace velmi nebezpečných metod, které jsou označovány jako **Útoky postranním kanálem**. Tyto metody zneužívají nedbalé implementace kryptosystému, chybových kanálů a dalších typicky jinak méně zajímavých míst. Výsledkem je naprosté prolomení kryptosystému. Postranní kanály vznikají nevědomě a jejich identifikace je velmi náročná. Některé postranní kanály vznikají až působením (např.: elektromagnetickým) na kryptografický modul. (18 s. 3-4)

3.1 Přístupy útoku na kryptosystém

3.1.1 Útok hrubou silou

Kryptoanalýza popisuje dva základní přístupy, jak útočit na kryptosystém. Prvním z nich je **útok hrubou silou**. Tento útok je základním stavebním kamenem pro kryptoanalýzu. Je založen na předpokladech, že útočník zná dešifrovací algoritmus a má v rukou šifrovaný text. Z definice konvenčního modelu, popsaném v kapitole 2.2 , vyplývá, že k získání čistého textu schází pouze správný klíč. Ovšem je třeba si uvědomit, že klíčů může být vždy jen omezený počet. Myšlenka útoku hrubou silou je tedy v tom, že se vezme sada všech možných klíčů a postupně se zkouší jeden klíč po druhém. Ze statistiky vyplývá, že je nutno průměrně projít polovinu potenciálních klíčů, aby byl nalezen ten správný.

Teoreticky řečeno, má-li útočník dostatek času a dostatečný výpočetní výkon, je schopný takto napadnout v podstatě jakoukoliv šifru. Ve chvíli, kdy jsou vyzkoušeny všechny potenciální klíče a následně jsou z nich odvozeny všechny potenciální zprávy, je úkolem vybrat jednu z nich a označit ji jako původní zprávu. Vyloučeny jsou ty zprávy, které nedávají logický smysl. Nalezeno může být i více původních zpráv s logickým obsahem. V této situaci

Největším problémem je však bezpochyby délka klíče. Už při navrhování kryptosystému se počítá s tím, že se může na systém útočit hrubou silou. Tento útok je ale velmi neefektivní, jelikož s rostoucí délkou klíče roste i počet potenciálních klíčů. Délka klíče se tak přizpůsobuje aktuálním výpočetním možnostem. Důsledkem toho by neměl být útočník schopen projít dostatek klíčů v přijatelném čase.

3.1.2 Kryptoanalytické útoky

Útoky hrubou silou jsou založené na minimálních požadavcích na znalost cílového kryptosystému. Oproti tomu jsou kryptoanalytické útoky založeny na důkladném studiu konkrétního systému a následném zneužití objevených, slabých míst. Cílem tohoto typu útoku nemusí být přímo prolomení klíče, ve skutečnosti stačí omezit množství potenciálních klíčů natolik, aby bylo možné následně na kryptosystém zaútočit hrubou silou a to v přijatelném čase.

Z pohledu kryptografického modelu, je snaha tímto útokem odhadnout přímo čistý text nebo použitý tajný klíč. Pokud se podaří pomocí tohoto útoku odhadnou hodnotu použitého klíče, pak je efekt tohoto útoku katastrofický, jelikož je v tuto chvíli prolomena veškerá komunikace, založená na tomto klíči v minulosti i budoucnosti. (3, s. 27)

3.2 Dělení útoků na základě znalosti

Útoky na kryptosystémy jsou jedním z největších nebezpečí této doby. Při návrhu nových kryptosystémů, je kladen důraz ne jednoduchost návrhu, ovšem právě díky relativní jednoduchosti je pak možné domyslet nejrůznější detaily, ošetřit slabá místa a ve výsledku vytvořit velice komplexní systém. Tomu se musí přizpůsobit i útočník, a tak se postupně celá situace rozvíjí do nepřehledného a těžko uchopitelného problému.

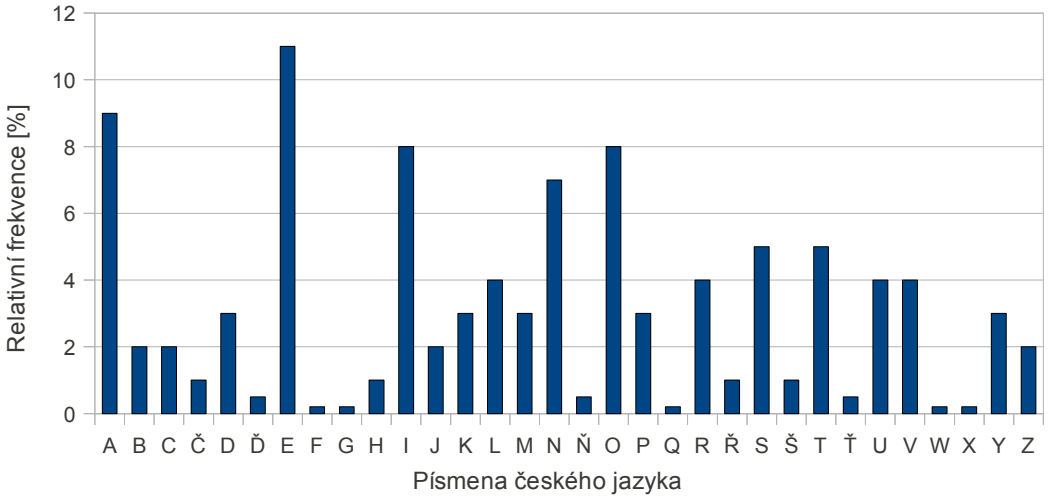
Zlaté pravidlo říká: *„Rozděl velký problém na několik menších částí a věnuj se jim odděleně.“* V kryptoanalýze se typy útoků dělí podle znalostí a dostupných prostředků útočníka.

- **Pouze šifrovaný text:** Tento typ útoku popisuje situaci, kdy útočník odposlechl šifrovanou komunikaci mezi dvěma uzly. Navíc však musí útočník disponovat znalostí šifrovacího, respektive dešifrovacího, algoritmu.
- **Známý čistý text:** U tohoto typu útoku platí stejné předpoklady jako u předchozího typu. Navíc však útočník vlastní jeden nebo více párů čistých textů a k nim jejich šifrované podoby. Podle (3, s. 28) mohou být příkladem známého textu hlavičky souborů. Toto se děje např. U všech dokumentů typu **Postscript**.
- **Vybraný čistý text:** Útočník zná algoritmus, má přístup k šifrovanému textu a navíc je tvůrcem čistého textu.
- **Vybraný šifrovaný text:** Útočník zná algoritmus, má přístup k šifrovanému textu a navíc je tvůrcem šifrovaného textu.
- **Vybraný text:** Útočník disponuje všemi výše zmíněnými znalostmi i zdroji.

3.3 Relativní frekvence písmen

Ilustračním příkladem kryptoanalytického útoku je využití charakteristiky čistého textu. Již zmíněná Caesarova šifra (kapitola 2.4.1) je tímto útokem snadno napadnutelná. Pro realizaci takového útoku je nutné získat statistické údaje o čistém textu.

Například nás může zajímat, které písmeno se v českých textech objevuje nejčastěji. Takovým písmenem je *E* a má relativní zastoupení 11% (6, Letter frequency). O Caesarově šifře už víme, že se jedná o mono-alfabetickou substituční šifru, tedy že nahrazujeme písmena metodou **znak za znak**. Pokud je tedy například 15% původního textu tvořeno znakem *E* a my ho nahradíme jiným, pak bude i tento nahrazující znak mít v šifrovaném textu zastoupení 15%.



Obr. 5 -Relativní frekvence písmen českého jazyka, zdrojová data: (6)

Výše popsaná metoda může pomoci identifikovat několik málo znaků, ovšem je třeba jít ještě o kus dál. Namísto určování zastoupení jednotlivých znaků, je možné určit procentuální zastoupení dvojic nebo trojic znaků. Odborně se jedná o **bigramy** a **trigramy** (3 s. 32- 35). Výčet bigramů a trigramů je čerpán ze zdroje (6).

- **Bigramy:** ST, PR, SK, CH, DN, TR
- **Trigramy:** PRO, UNI, STA, ANI, OVA, YCH, STI, PRI, PRE, OJE, REN, IST, STR, EHO, TER, RED, ICH

Aplikováním bigramů a trigramů, se dají identifikovat znaky s nízkou relativní frekvencí. Čím větší objem šifrovaného textu útočník vlastní, tím jednodušší je uskutečnit kryptoanalytické útoky.

3.4 Rozptyl a zmatek

Tyto pojmy se pojí s velkým jménem, Claud E. Shannon. Ten ve své práci, (5 s. 708-710), definuje metody pro zmatení útočníka na kryptosystém. Útočník může využít znalosti kryptosystému a pomocí statistické analýzy zredukovat počet potenciálních klíčů nebo přímo odhadnout správný klíč. Ideální kryptosystém je z definice proti tomuto typu útoku odolný, ovšem v reálném světě je zapotřebí tyto situace ošetřit. C. E. Shannon navrhuje, že namísto snahy,vytvořit ideální systém, je možné využívat konceptu rozptylu a zmatku.

Jak už bylo řečeno v kapitole o relativní frekvenci písmen, pakliže jeden znak čistého textu má v konečném důsledku vliv pouze na jeden znak šifrovaného textu, můžeme této znalosti jednoduše využít. **Rozptyl** je tedy metoda, kterou chceme docílit, aby vytvoření každého znaku šifrovaného textu bylo ovlivněno více znaky čistého textu. Tím se zajistí, že analýza relativní frekvence šifrovaného textu nebude korespondovat s analýzou čistého textu a nepovede k rozluštění šifrovaného textu. Tato metoda se implementuje pomocí transpozice. (3, s. 66-67)

Rozptyl má tedy za úkol vytvořit vazby mezi šifrovaným a čistým textem tak komplexní, jak to jen lze. **Zmatek** oproti tomu má vytvořit komplexní vazby mezi šifrovaným textem a klíčem, pomocí kterého byl šifrovaný text vygenerován. Důvodem je ochrana tajného klíče v případě, kdy má útočník statistické informace o šifrovaných textech. Tato metoda se implementuje pomocí substituce. (3, s. 66-67)

Pro demonstraci těchto technik lze použít applet, umístěný na webové stránce (17). Následující text bude vstupem pro algoritmus MD5.

```
Dobry den,

Pan Leopold Tajemny ma narok na zakoupeni vypocetni
techniky ve vysi20 000,- Kc

s pozdravem, vedouci IT
```

Výstup algoritmu MD5: `0x66a43ff96be8359b9ba719e3a8215806`

Pokud by došlo k nahrazení částky 20 000 za částku 80 000, výstup bude kompletně změněn. I přesto že došlo k nahrazení pouze jednoho znaku.

Výstup po změně částky: `0x06d0cdab461987a452458b22d6cf0e90`

4 Symetrické šifrování

Symetrické šifry jsou založeny na principu sdíleného klíče. Tento typ šifrování je velmi oblíbený, široké zastoupení mají v praxi především šifry blokové (9). Mezi blokové šifry patří algoritmy **DES, 3DES, IDEA, Blowfish, CAST** a další.

4.1 Proces výměny sdíleného klíče

Nejslabším místem symetrického šifrování je právě nutnost sdílení tajného klíče mezi odesílatelem a příjemcem. Jelikož je veškerý úspěch šifrování založený na udržení tajemství o klíči, mohl by si útočník výrazně ulehčit práci, aktivním nebo pasivním odposlechnutím procesu výměny sdíleného klíče. K tomuto účelu bylo vyvinuto několik metod:

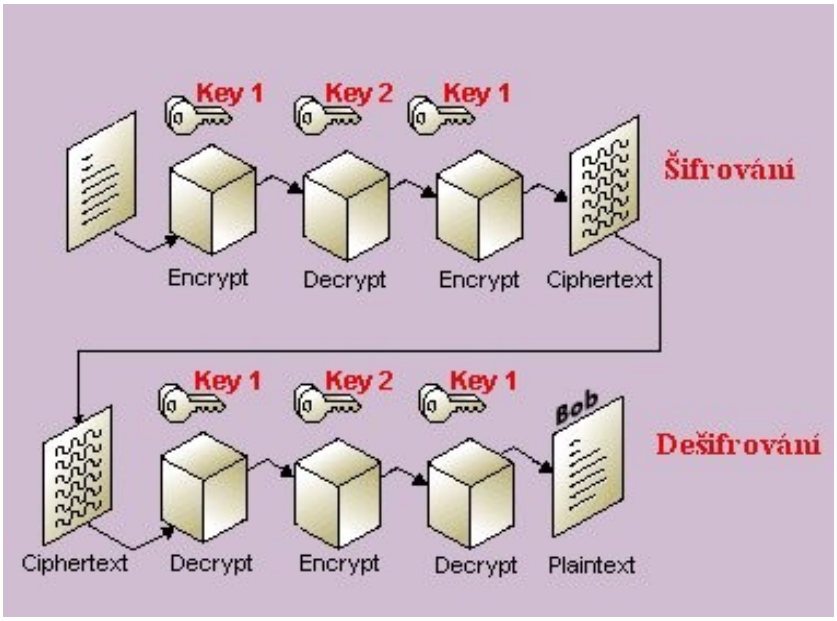
- **Před-sdílený klíč:** Tato metoda je velice efektivní, jelikož využívá toho, že komunikující stanice se dobře znají. Tajný klíč je jednoduše nakonfigurován administrátorem zařízení. Proces výměny klíče se tedy neuskuteční, takže jej není možné odposlechnout. Přesto má však tato metoda i několik nedostatků. Zaprvé je do bezpečnostního procesu zahrnut lidský faktor. Zde se naskýtá šance pro nepřeberné množství technik sociálního inženýrství, ke zmanipulování kritické osoby pro získání tajného klíče. Při dokumentaci počítačových sítí se informace o nastavení zařízení sbírají a archivují. Je tedy nutné dostatečně zabezpečit tyto dokumenty a autorizovat jejich čtení pouze důvěryhodným osobám.

- **Zabezpečený kanál:** K výměně klíče se použije důvěryhodné spojení. Například může jít o vytáčené spojení, kde se za pomocí pomalejší ale důvěryhodné linky vymění tajný klíč a dále je využívána rychlejší nedůvěryhodná síť na které již putují informace šifrovaně. Riziko prolomení stále existuje ale je rozprostřeno na více typech technologií. Spoléhá se tedy na to, že útočník nebude schopen prolomit všechny typy spojení najednou. V prostředí Internetu se namísto sekundární linky využívá šifrovací metoda, určená k předávání tajných klíčů (**Diffie-Hellman** algoritmus). Tato metoda je tedy schopná provést bezpečně výměnu tajného klíče za pomocí nedůvěryhodného spojení.

4.2 Data Encryption Standard

Na konci šedesátých let IBM zahájila výzkumný projekt, který vedl Horst Feistel. Z projektu vzešel v roce 1971 algoritmus LUCIFER, který byl prodán britskému pojišťovacímu trhu Lloyd's of London pro použití v systému výdeje peněz. Protože byl projekt nadějný, společnost IBM se rozhodla věnovat vývoji šifrovacího systému, který by byl ideálně umístěn pouze na jednom čipu. Na vývoji pracovali i externí konzultanti a výsledkem byla přepracovaná verze algoritmu LUCIFER. Tato verze byla mnohem odolnější proti kryptoanalytickým útokům, ale aby jej bylo možné umístit na jeden čip, musela být zredukována délka klíče. V roce 1973 společnost NBD (National Bureau of Standard) vydala požadavek na navrhnutí národního šifrovacího standardu. Jako nejlepší algoritmus byla vybrána upravená verze algoritmu LUCIFER. Tento algoritmus byl prohlášen standardem a označen jako DES. (3, s. 72-73)

Algoritmus DES je blokovou šifrou, která šifruje bloky o velikosti 64 bitů a vytváří z nich bloky šifrovaného textu o stejné velikosti. Algoritmus využívá klíče o velikosti 56 bitů. Šifrování se zpracovává ve fázích, těch je celkem 19. První fází je transpozice, která je nezávislá na klíči. V poslední fázi je pak přesný opak této transpozice. Před poslední fází se prvních 32 bitů prohodí s posledními 32 bity. Funkce zbylých fází je pak totožná, ale jako parametr těchto fázi je tajný klíč. DES algoritmus je symetrický, takže dešifrování lze provést pouze provedením jednotlivých kroků v opačném pořadí. (4, s. 738-739)



Obr. 6 - 3DES za pomoci dvou klíčů, zdroj: (16), (přepracovaný)

V roce 1979 si firma IBM uvědomila, že použitý klíč algoritmu DES je příliš krátký. Jako efektivní řešení se naskytla možnost, použít algoritmus DES třikrát po sobě. Tento způsob se dal vyřešit za pomocí dvou klíčů a třech fází, kde byly klíče použity.

4.3 Advanced Encryption Standard

Tento standard publikoval institut NIST (National Institute od Standard and Technology) v roce 2001. AES je symetrická, bloková šifra, jejímž úkolem je nahradit DES/3DES jako standard pro široké spektrum aplikací. Než byl standard AES publikován, bylo uspořádáno dlouhodobé výběrové řízení. Z celkového počtu 21 kandidátů byl nakonec zvolen a následně adoptován algoritmus Rijndael. (3. s.140-143)

4.3.1 Proces výběru moderního kryptosystému

Před samotným vyhodnocením kandidátů na AES bylo nutné nejprve navrhnout kritéria, podle kterých by bylo možné kandidáty porovnávat (3, s. 141-143). Hlavní kategorie byly následující tři:

1. **Bezpečnost:** Toto kritérium se zabývá odolností kryptosystému vůči kryptoanalytickým útokům. Jako minimální hranice velikosti klíče byla zvolena velikost 128 bitů. To zajišťuje dostatečnou ochranu proti útoku hrubou silou. Počítalo se s aktuální úrovní technologie a jejím očekávaným vývojem.
2. **Náklady:** Moderní algoritmus musí být praktický a široce uplatnitelný. Také musí mít vysokou výpočetní efektivitu, aby mohl být uplatnitelný na pomalejších strojích i na strojích vyžadující extrémní přenosovou rychlost.
3. **Algoritmus a implementační charakteristika:** Tato kategorie zahrnuje rozmanité požadavky. Mezi ně patří například flexibilita, jednoduchost, možnosti implementace.

Po vyhodnocení popsaných kritérií bylo vybráno 15 kandidátů. Následující kritéria byla využita pro konečné vyhodnocení (3, s. 141-143).

1. **Obecná bezpečnost:** Po dobu tří let byli zbývající kandidáti analyzováni kryptoanalytickou veřejností. Odhalovaly se chyby, vyhodnocovaly se silné a slabé stránky jednotlivých řešení. Testovány byly kryptoanalytické útoky na základě diferenciální a lineární kryptoanalýzy.
2. **Softwareová implementace:** V této kategorii se hodnotila rychlost provádění algoritmu, výkonnost na různých platformách a možnosti upravení rychlosti pomocí volby délky klíče.
3. **Prostředí s omezenou pamětí:** Takové omezení platí např. u čipových karet.
4. **Hardwareová implementace:** Hardware lze optimalizovat pro rychlost nebo velikost klíče. Takováto optimalizace může mít velký dopad na celkovou cenu konečného řešení.

5. **Útoky na implementaci:** Kromě analýzy klasických kryptoanalytických útoků byly zkoumány i méně typické útoky, například útok založený na analýze spotřeby el. energie při provádění algoritmu.
6. **Šifrování versus dešifrování:** Toto kritérium zvažuje rozdíly mezi šifrováním a dešifrováním. Např. pokud se algoritmy liší, pak je vyžadován další paměťový prostor.
7. **Agilnost klíče:** Vyžadována je možnost rychle změnit klíč a to s minimální spotřebou zdrojů.
8. **Ostatní všestrannost a flexibilita:** Flexibilita parametru zahrnuje podporu dalších klíčů spolu s velikostí šifrovaného bloku. Dále zahrnuje možnost navýšení počtu fází algoritmu, což by umožnilo obranu proti nově objeveným útokům. Implementační flexibilita umožňuje optimalizovat prvky šifrování pro různé druhy prostředí.
9. **Potenciál pro paralelismus na úrovní instrukcí:** Snaha o přípravu algoritmu pro technologie budoucnosti.

Konečné pořadí hodnocených algoritmů a jejich skóre (4, s. 742):

1. **Rijndael** (Joan Daemen, Vincent Rijmen) – 86 hlasů
2. **Serpent** (Ross Anderson, Eli Biham, LarsKnudsen) – 59 hlasů
3. **Twofish** (Tým, který vedl Bruce Schneier) – 31 hlasů
4. **RC6** (RSA Laboratories) – 23 hlasů
5. **MARS** (IBM) – 13 hlasů

5 Hašovací funkce a asymetrické šifrování

5.1 Princip hašovacích funkcí

Hašovací funkce jsou schopny vypočítat ze vstupu o libovolné délce výstup o pevné délce. Výstup je označován jako otisk. Jako nejdůležitější vlastností je zamezení schopnosti, z výstupu odhadnout původní vstup. Proces hašování musí být rychlý a jakákoliv změna vstupních dat, i kdyby mělo jít pouze o jediný bit, musí vést k naprosto odlišnému výstupu (4, s. 759). Díky těmto vlastnostem, je pak možné hašovací funkce uplatnit jako bezpečnostní prvek pro ochranu integrity přenášených dat.

Délka otisku bývá velmi zredukována oproti vstupnímu textu. Vzniká teoretická možnost, že více vstupních textů bude mít shodný otisk. To je označováno jako kolize a v praxi je nežádoucí. Kvalitní hašovací funkce předcházejí vzniku kolizí (19, 2009, s. 21).

5.2 Princip kryptosystémů s veřejným klíčem

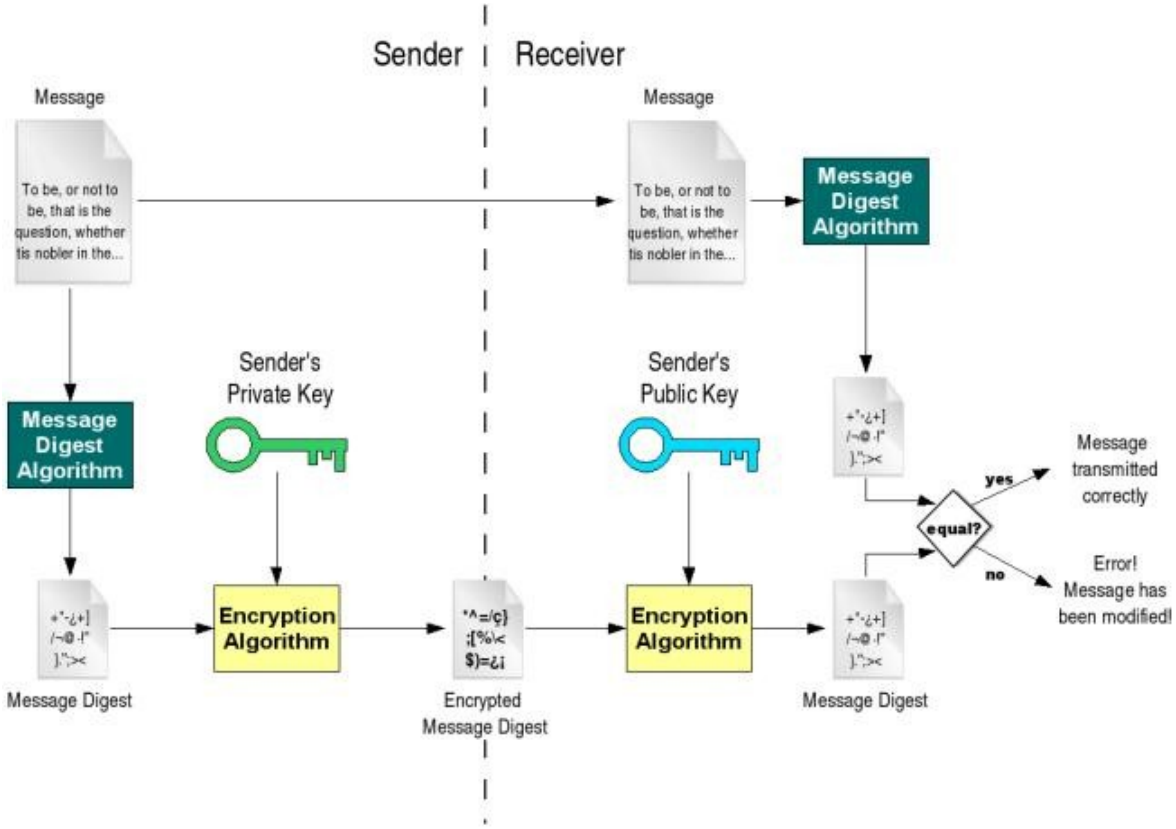
Asymetrické šifrování mění do značné míry šifrovací model. Používají se dva klíče, jeden je určený pro šifrování (**veřejný klíč**) a druhý pro dešifrování (**soukromý klíč**). Oba klíče jsou generovány najednou, takže ten kdo klíče generuje, zná celé tajemství. Aby tvůrce klíčů mohl s někým komunikovat, musí předat jeden z klíčů druhé osobě. Předání klíče se označuje jako **zveřejnění klíče**. Tvůrce klíčů odesílá klíč veřejný. Pokud by zveřejnil klíč dešifrovací, pak by to znamenalo, že tvůrce klíčů je zároveň odesílatelem zprávy. V takovém případě, by s odesláním zprávy zároveň odeslal i návod jak tuto zprávu dešifrovat, což by bylo nelogické.

Z minulého odstavce také vyplývá, že k veřejnému klíči může mít přístup kdokoliv. V praxi se dokonce veřejný klíč uvádí na osobních webových stránkách majitele klíče. Důvodem je jeho přístupnost pro každého, kdo má zájem na tom, aby jeho zprávu mohl přečíst pouze majitel soukromého klíče. S tím se však pojí otázka: *„Nelze ze znalosti veřejného klíče odvodit podobu soukromého klíče?“* Zajistit, aby toto nebylo možné, je jedním z hlavních kritérií asymetrického šifrování. Jaroslav Pinkava (9) píše, že toto kritérium tvoří podstatu šifrování s veřejným klíčem. Také uvádí, že právě díky tomuto kritériu je mnohem obtížnější konstruovat takovéto algoritmy. To má následně vliv na pomalejší rychlost provádění těchto algoritmů. Právě rychlost provádění šifrování pak může hrát roli v tom, jaká šifrovací metoda bude nakonec zvolena.

5.2.1 Využitý asymetrického šifrování

Asymetrické šifrování se úspěšně využívá v kombinaci se symetrickými šifrovacími metodami. Data které je třeba šifrovat se posílají pomocí symetrického šifrování, to je typicky méně náročnější a proto se dosahuje vyššího výkonu (9). Pro výměnu tajného klíče pro symetrické šifrování se naopak využívá asymetrických algoritmů. Ty jsou uzpůsobené k tomu, aby bezpečně přenesli informace, potřebné k vypočítání tajného klíče, skrz nedůvěryhodné spojení a to i za předpokladu, že se komunikující stanice neznají.

Další možností využití asymetrického šifrování je digitální podpis. Tímto mechanismem se zajišťuje pravost elektronických dokumentů. Následující obrázek (viz Obr. 7) znázorňuje základní princip tvorby a ověření podpisu. (19, s. 27)



Obr. 7 - Princip tvorby a ověření podpisu, zdroj: (21)

1. Na vstupu je elektronický dokument, ze kterého odesilatel vypočte haš.
2. Výsledný haš je zašifrován pomocí soukromého klíče odesilatele. Tím je vytvořen digitální podpis.
3. Následně je podpis odeslán spolu s elektronickým dokumentem. Pokud již příjemce nevlastní veřejný klíč odesilatele, je potřeba zaslat i veřejný klíč.
4. Příjemce zprávy dešifruje přijatý podpis pomocí veřejného klíče odesilatele. Takto je získán haš z dokumentu který provedl odesilatel.
5. Příjemce zprávy vypočte haš z obdrženého dokumentu. Nyní příjemce vlastní dva otisky ze stejného dokumentu. Jeden byl pořízen před odesláním a druhý po přijetí.
6. Pokud se oba otisky rovnají, pak mohl být digitální podpis vytvořen pouze majitelem soukromého klíče odesilatele. Zároveň je tím prokázáno, že zpráva nebyla v průběhu přenosu pozměněna. (19, s. 27)

5.3 RSA

Tento algoritmus je typický zástupce asymetrického šifrování. Vytvořen byl autory R. Rivest, A. Shamir a L. Adleman, odtud také pochází název samotného algoritmu. Tento algoritmus je velmi odolný, trvalo téměř čtvrt století, než byl prolomen. Svou odolnost však zakládá na klíči velikosti minimálně 1024 bitů, což provedení algoritmu značně zpomaluje (46, s. 753).

Bezpečnost tohoto algoritmu je založena na obtížnosti úlohy faktorizace velkých čísel. Faktorizací je myšlen rozklad celého čísla na součin prvočísel. Uvádí se, že faktorizace 500ciferného čísla by pomocí útoku hrubou silou trvala 10^{25} let (4, s. 754).

5.3.1 Praktický příklad RSA

Vytváření klíčů pro šifrování pomocí algoritmu RSA, se dá rozdělit do několika kroků: (3, s. 268-271), (4, s. 753-754)

1. Je potřeba zvolit dvě prvočísla, typicky se vybírají prvočísla o velikosti 1024 bitů, pro názornost budou stačit hodnoty: $p=3$, $q=11$
2. Spočítá se takzvaný RSA modulus: $n=p\times z=33$
3. Spočítá se hodnota Eulerovy funkce: $z=(p-1)\times(q-1)=20$
4. Zvolí se celé číslo ***e***, které je nesoudělné se *z* a je menší než *z*: $e=3$
5. Nalezne se celé číslo ***d***, které odpovídá vzorci: $d\times e=1 \bmod z=7$
6. Veřejný klíč je vytvořen jako dvojice (**n**, **e**), soukromým klíčem je dvojice (**n**, **d**).

Tvůrce klíčů následně odešle veřejný klíč osobě, se kterou chce komunikovat. Ta šifruje metodou znak po znaku podle podle vzorce $C=P^e \bmod n=P^3 \bmod 33$, kde P označuje jednotlivé znaky čistého textu a C označuje znaky šifrovaného textu. Proces dešifrování probíhá podle vzorce $P=C^d \bmod n=C^7 \bmod 33$.(3, s. 268-271), (4, s. 753-754)

Symbol	Pořadí v abecedě	Šifrovaný text (bez mod)	Šifrovaný text
A	1	1	1
H	8	512	17
O	15	3375	9
J	10	1000	10

Šifrovaný text	Dešifrovaný text (bez mod)	Dešifrovaný text	Pořadí	Symbol
1	1	1	1	A
17	410 338 673	8	8	H
9	4 782 969	15	15	O
10	10 000 000	10	10	J

6 Laboratorní úlohy

6.1 Pretty Good Privacy

Tato laboratorní úloha je zaměřená na využití techniky asymetrického šifrování v prostředí dobře známém, a to při mailové komunikaci.

V dnešní době už není vůbec zarážející, že někteří administrátoři serverů vlastně nikdy fyzicky neviděli hardware na kterém pracují. Specializované firmy provozují serverové farmy, zajišťují kompletní podporu a ručí za různá rizika. Je běžnou praxí, že uživatelé těchto serverů se připojují přes zabezpečené kanály, ovšem na co se někdy zapomíná, je bezpečnost při předávání hesel. Mailové servery bývají dobře zabezpečené, ovšem pokud předáváme kritické informace, neměli bychom nic ponechat náhodě.

6.1.1 Použité technické vybavení

- **PC1:**
  - Procesor AMD Athlon(tm) II X4 631 Quad-Core Processor 2.60 GHz
  - Operační paměť 8,0 GB
- **Notebook:**
  - HP Compaq 6530b
  - Procesor Intel Centrino 2, P8600 – 2,4GHz.
  - Operační paměť 2 GB

6.1.2 Použité programové vybavení

- **PC1:**
  - Operační systém Windows 7
  - Microsoft Office Outlook 2007 (12.0.4518.1014) + plugin Gpg4win
  - Google Chrome 18.0.1025.151
- **Notebook:**
  - Operační systém Linux Ubuntu 10.04 LTS, jádro 2.6.32-40-generic
  - Mozilla Thunderbird 3.1 + rozšíření Enigmail
  - Google Chrome 18.0.1025.151

6.1.3 Instalace potřebných součástí

Na PC1 byl nainstalován webový prohlížeč Google Chrome, který je možné stáhnout například na adrese www.google.com/chrome. Tento prohlížeč bude využíván k přihlášení do mailové schránky pomocí webového přístupu. Mailová služba byla registrována na adrese www.gmail.com. Výběr webového prohlížeče a mailové služby je založen na osobních preferencích a je možné využit jiných zastupujících služeb.

Realizaci PGP na platformě Windows byla provedena za pomocí tradičního mailového klienta, Microsoft Office Outlook 2007. Instalace probíhá klasicky přes instalačního průvodce. Mailový účet lze svázat s aplikací pomocí volby **Nástroje → Nastavení účtu → Nový…**, dále je třeba uvést jméno, mailovou adresu a přístupové heslo. Pro podporu PGP je potřeba doinstalovat plugin **Gpg4win**, který je šířen pod freeware licencí[11]. Tento plugin obsahuje aplikaci Kleopatra, která spravuje PGP certifikáty a klíče.

Instalaci na distribuci Ubuntu (zařízení Notebook) lze provést pomocí **Systém → Správa → Správce balíků Synaptic**. Je nutné zadat heslo administrátora pro získání oprávnění k provádění změn v systému. Dále je možné jednotlivé balíčky najít pomocí rychlého vyhledávání. Seznam použitých balíčků je uveden v Tab. 1. Balíčky je nutné **označit k instalaci** a potvrdit tlačítkem **Použít**.

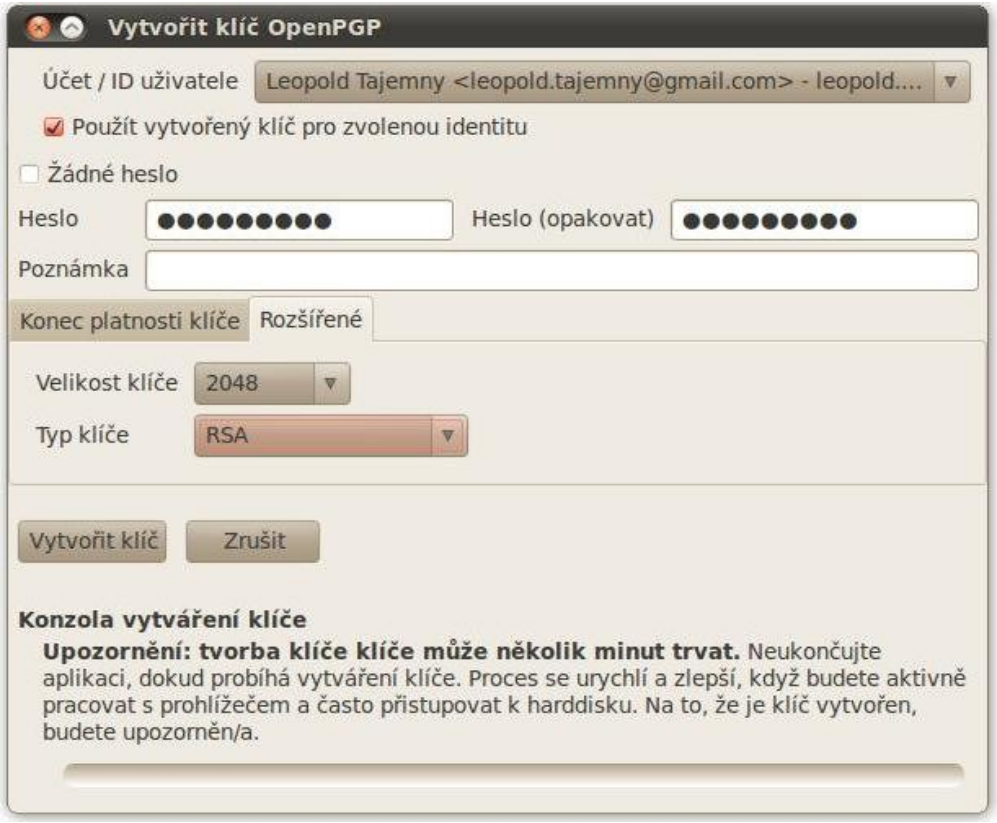
Tab. 1 -Seznam instalovaných balíčků a jejich verzí

Jméno balíčku	**Verze balíčku**
Thunderbird	3.1.20+build1+nobinonly-0ubuntu0.10.04.1
Enigmail	2:1.1.2-0ubuntu0.10.04.1
google-chrome-stable	18.01025.162-r131933

6.1.4 Nastavení a vytvoření šifrovacích klíčů

Mozilla Thunderbird je klientská aplikace umožňující přístup k mailové schránce. Nejprve je třeba svázat mailovou schránku s aplikací Thunderbird. Tato volba se nachází v nástrojovém panelu **Edit → Account setting → Account Action → Add Mail Accoount**. Zde se nám otevře okno, kde musíme vyplnit a potvrdit přihlašovací údaje.

Pakliže aplikace korektně přijímá mailové zprávy, je možné přistoupit k vytvoření šifrovacích klíčů. Jelikož byl nainstalován balíček Enigmail, je přidána do nástrojového panelu aplikace Thunderbird položka **OpenPGP**, ve které je možné spravovat veškeré činnosti ohledně šifrování. K vytvoření klíče se lze dostat přes **OpenPGP → Správa klíčů**. Zde se otevře okno, které zobrazuje všechny vytvořené nebo importované klíče. Vytvoření nového páru klíčů lze uskutečnit pomocí přehledného průvodce (viz Obr. 8), nacházejícího se v nástrojovém panelu pod položkou **Vytvořit → Nový pár klíčů**.



Obr. 8 - Tvorba šifrovacích klíčů na platformě Linux

Nejprve je nutné svázat klíč s mailovým účtem. Pokud využíváme více mailových účtů a chceme šifrovat veškerou komunikaci, je nutné vytvořit pro každý účet samostatný pár klíčů. Dále se zadává heslo, které bude požadováno před dešifrováním zpráv. Pokud by tedy došlo k tomu, že nám někdo soukromý klíč odcizí, stále bude proces dešifrování chráněn heslem. Nakonec je nutné zadat termín platnosti klíčů a šifrovací metodu. Termín je možné zadávat v řádech dnů, měsíců nebo roků. Podporované jsou dvě šifrovací metody s variabilní délkou klíče, **RSA** a **DSA & El Gamal**. Podporované délky klíče jsou **1024**, **2048** a **4096** bitů. Po procesu generace klíčů se doporučuje vytvořit **revokační certifikát**. Ten se používá např. po ztrátě soukromého klíče pro zneplatnění tohoto klíče.

Po procesu vytvoření klíčů je třeba provést podepsání. To by měla provést důvěryhodná certifikační autorita V této úloze byla využita volba vlastnoručního podepsání, volba **OpenPGP → Správa klíčů**, zde se označí vybraný klíč, otevře se kontextová nabídka a vybere se volba **podepsat klíč**. Od této chvíle můžeme šifrovat a podepisovat zprávy. Aby mohl příjemce ověřit validitu přijatého podpisu, ukládají se veřejné klíče na klíčové servery (např. pgp.mit.edu, keys.gnupg.net a další). Tato volba je ve správci klíčů v nástrojové liště: **keyserver → odeslat veřejné klíče**.

Na následujících dvou ilustracích (Obr. 9 a Obr. 10) je textová podoba soukromého a veřejného klíče. Začátek i konec klíče je jasně označen, navíc se v hlavičce udává verze protokolu PGP. Soukromý klíč bývá typicky delší, jelikož veškerá bezpečnost se opírá právě o kvalitní soukromý klíč. I když by se veřejný klíč dostal do rukou útočníka, bylo by velmi obtížné z něho odhadnout přesnou podobu soukromého klíče. Tato vlastnost je typická pro asymetrické šifrování.

```
-----BEGIN PGP PRIVATE KEY BLOCK-----
Version: GnuPG v1.4.10 (GNU/Linux)

lQH+BE+RWIwBBADDHJHpXi6DPSWdE6r2Srs6UQ/wBDteJNw77juQMvbnm+dQGptM
1lAD1gxBrcZOyqpCWdRY8P6fSl/hsSPpmVMBb4HwQ9MOx7tKXnlTtuB9CKcVIvJj
9bN5NEQ5LpxFLq/yDu+D9jHm3vvPHruN0RR/k1Jt5pmd9nFUKcp1kYwslQARAQAB
/gMDAqxB1LgwcmELYLAEie0NZZIEFaVAFAvnMU7KdqV7W/h+/Uqs9+71gKEiLmde
kKr33OEpZEXQxt+WeTIsABlpfUdE4YuLbYWKBtoZzGC2hCyA7aaSgQ1SR4H7PY/X
l1eJ1ZX8ZIKvTQBKQkER6JMM2NgLb2EEIgNMjg+y0QdgZehF+6JBYnRBaISQeCu1
K9C2V47xaXVPXViiXGZDB9oYoaIYkibUbQgIzKJDwbA5nQvbOkXnDBh0WEvO06SN
K6Td5eqD8R7Mpe2SjPdiydfdPSb1JiJdqqS2wAlXW48i1hqrMT9DfGx4J1UTioHv
BcYZH3996jqy0nxngsJ1hkHvwmyL5iTdpHQxsSxXXO01ljPunL1zDlzhp5IWLy31
2kbh8s1cj49FRQQS7RFJduz34YRUrG8lXv2c9i8b87Zm8+HX4zdyzyo/BwHKr6C3
K5ppV37h/IwwJQOs47QGlHkJvaWyiInBaGS8HT2z6qsVtCtMZW9wb2xkIFRhamV0
bnkgPGxlb3BvbGQudGFqZW1ueUBnbWFpbC5jb20+iL4EEwECACgFAk+RWIwCGyM3
CQlmAYAGCwkIBwMCBhUIAgkKCwQWAgMBAh4BAheAAAoJEKyK1dvFgFcvMhcD/R41
cEDuRl/f0vWejvS/CjZJzxkwAg9lAtYMiJdcCrt/cFCKtJciJrj/SNQPbRv8Y739
zDqmsv8yMH6hE5XYq40YrKKbtXqkST3ahP3a6eLa29P38mwSMqQYYUIeENICN2D9
ELY89wkcAEC9O5fhn/CqJkMLThEa5cDjkeBwRBr8nQH+BE+RWIwBBADMW5YwNmq1
e7zhb5XI3gIks3ERdrXEkBHIM+hJPu4qwwaLOXjOw/cOuqFXBtIf456zGIaPt0Il
czGIXRwdnRWtVS2HsOn7aEgrBeQF6LHgXugZKAOQt1IxDav6LcEdqqh5oLEpacMe
wfDT026gSW9OiEplO70DmRloEY+DXBcLwwARAQAB/gMDAqxB1LgwcmELYDTPJJ6N
79tU1oLijw47xXzuoIHtnC0dV41xNTeyxhnYv2tgh3Az6W7ZKERhSrFea8cAr+lK
44EKY5/lxybklvTFtY47y/sihxQb5Jpv+diwu31ifPWHGCtF9d03PQnfsCaQhpma
C0xFDz9gTRQO5yjFhLiYuSuSi357+F2QTranZhQYr4Nu7wPiSW2ifFX3pe05uTEK
mm/pdmYt+WlLYsh/810mrgyZvGmVwZRl2bdu7Wstc5SgjDHOw5pJVk1ESInPSnta
KGHpK9WsRvM01Jnfda1U2/wc8YVAuJoye6hKZeNJrWyOSD8p3fFhJ8XK741e9g8s
yxAvx/FPNNUjmojwZUmGuNymlf47qFUJFXiJNdaG43DE4P/+2c5yUAZT5CfyE6HI
x2Mg8yO8dq8s1RcIUPR11yTRIaCRoN1DzOMqYt/Q1D4M4a1zlIl8fxR2WipLDuRk
ETm4QptCQ7NspmdefMQciKUEGAECAA8FAk+RWIwCGwwFCQlmAYAACgkQrIrV28Wo
Vy+rdAQAvDk5F8Q3DSIkmIQofJCe3vI/Ute4EyCA8XSWvfqD7V9DfV8P+5mrhmNV
qhbqEOMXV9vFMWbwDgG3eg8sOoOOeZmFBvkzYtOAtoXHXyj4gmWvXiwHMleAWtfa
gwIKIWo6OqgvRZCucb010sMDsXOgxJUxF1YRSzNL21q9mw/NNLA=
=PBsF
-----END PGP PRIVATE KEY BLOCK-----
```

Obr. 9 - Ukázka soukromého klíče PGP

```
-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: GnuPG v1.4.10 (GNU/Linux)

mI0ET5FYjAEEAMMckeleLoM9JZ0TqvZKuzpRD/AEO14k3DvuO5Ay9ueb51Aam0zW
UAPWDEGtxk7KqkJZ1Fjw/p9KX+GxI+mZUwFvgfBD0w7Hu0peeVO24H0IpxUi8mP1
s3k0RDkunEUur/IO74P2Mebe+88eu43RFH+TUm3mmZ32cVQpynWRjCyVABEBAAG0
K0xlb3BvbGQgVGFqZW1ueSA8bGVvcG9sZC50YWplbW55QGdtYWlsLmNvbT6IvgQT
AQIAKAUCT5FYjAIbIwUJCWYBgAYLCQgHAwIGFQgCCQoLBBYCAwECHgECF4AACgkQ
rIrV28WAVy8yFwP9HgZwQO5GX9/S9Z6O9L8KNknPGTACD2UC1gyIl1wKu39wUIq0
lyImuP9I1A9tG/xjvfXMOqay/zIwfqETldirjRisopu1eqRJPdqE/drp4trb0/fy
bBIypBhhQh4Q0gI3YO8Qtjz3CRwAQL07l+Gf8KomQwtOERrlwOOR4HBEGvy4jQRP
kViMAQQAzFuWMDZqmXu84W+VyN4CJLNxEXa1xJARyDPoST7uKsMGizl4zsP3Drqh
VwbSH+OesxiGj7dCNXMxiF0cHZ0VrVUth7Dp+2hIKwXkBeix4F7oGSgDkLdSMQ2r
+i3BHaqoeaCxKWnDJsHw09NuoElvTohBpTu9A5kZaBGPg1wXC8MAEQEAAYilBBgB
AgAPBQJPkViMAhsMBQkJZgGAAAoJEKyK1dvFgFcvq3QEALw5ORfENw0iJJiEKHyQ
nt7yP1LXuBMggPF0lr36g+1fQ31fD/uZq4ZjTqoW6hDjF1fbxTFm8A4Bt3oPLDqD
jnmZhQb5M2LTgLaFx18o+IJlr14sBzJXgFrX1oMCCiFqOjqoL0WQrnG9NdLDA7Fz
oMSVMRdWEUszS9tavZsPzTSw
=Hzw8
-----END PGP PUBLIC KEY BLOCK-----
```

Obr. 10 - Ukázka veřejného klíče PGP

Pro ilustraci lze vyzkoušet, jestli se nachází záznam o klíči, na jednom z klíčových serverů. Nejprve je nutné zjistit na jaké servery byl veřejný klíč odeslán. V nabídce **OpenPGP → Předvolby… → Určete Váš/Vaše keyserver(y)** je v prvním okně seznam webových adres používaných klíčových serverů. Pro pozdější účely se ještě zanese do druhého okna adresa jednoho z klíčových serverů (např. pgp.mit.edu). Tento server se bude později používat k ověření podpisu.

Nyní se přistoupí ve webovém prohlížeči na adresu pgp.mit.edu. Pro vyhledání konkrétního záznamu je třeba zadat **ID klíče**, to lze najít ve správci klíčů (v tomto případě je to: 0xC580572F). V některých aplikacích se tento identifikátor vypisuje bez prefixu **0x**, který označuje hexadecimální zápis uváděného čísla. Na zmíněných klíčových serverech je však tento prefix vyžadován). Zaškrtne se volba **Show PGP fingerprints for keys** a provede se vyhledání. Výstup viz Obr. 11 na následující straně.

Obr. 11 - Vyhledaný záznam na klíčovém serveru

Ze záznamu můžeme získat datum vytvoření klíče, haš otisk, jméno a mail. Identifikátor klíče navíc slouží jako odkaz na veřejný klíč (viz Obr. 12). Tyto informace se procházejí při ověřovacím procesu a následně je uživatel upozorněn pomocí jednoduché ikony.

Obr. 12 - Veřejný klíč nalezený na klíčovém serveru

Nyní může tvůrce klíče (stanice Notebook) napsat mail osobě, se kterou chce komunikovat. Přiloží svůj veřejný klíč a zprávu podepíše. Příjemce ověří přiložený podpis oproti záznamu na klíčovém serveru a veřejný klíč si uloží. Nyní může příjemce napsat tajnou zprávu, zašifrovat jí uloženým veřejným klíčem, přiložit podpis a případně také přiložit svůj veřejný klíč.

V aplikaci Thunderbird, lze provést odeslání veřejného klíče jednoduše pomocí nástrojové lišty **OpenPGP → Připojit můj vlastní veřejný klíč**. Následně je možné ještě zprávu podepsat. Tentokrát lze využít ikonu, nešťastně pojmenovanou, **OpenPGP**. Zobrazí se zaškrtávací dialog s nabídkou **šifrovat** a **podepsat**. V tomto případě nelze zprávu šifrovat jelikož není zanesený do správce klíčů veřejný klíč příjemce.

Osoba na stanici PC1 nyní obdržela veřejný klíč v nezašifrované podobě. Ověření podpisu probíhá vždy při otevření podepsaného mailu a jeho výsledek je vypsán do nově otevřeného okna. Po uložení souboru s veřejným klíčem (formát .asc) lze v aplikaci Kleopatra zanést veřejný klíč do seznamu známých klíčů. To lze uskutečnit spuštěním dialogu pomocí ikony **Import Certificates**, kde se vyhledá soubor .asc v souborovém systému. V aplikaci Outlook se šifrování a podepisování zpráv uskutečňuje zaškrtnutím konkrétních požadavků na kartě **Doplňky**.

Obr. 13 zachycuje situaci po přijetí zašifrovaného mailu. Podpis je označen jako správný, takže osoba která vlastní soukromý klíč, je opravdu osobou, která odeslala tuto zprávu. Nicméně je podpis označen jako nedůvěryhodný. To je způsobeno tím, že podpis byl proveden vlastnoručně. Nikdo tak nemůže zaručit, že osoba vlastnící tento soukromý klíč, je opravdu tou osobou, za jakou se vydává. Aby byl podpis důvěryhodný hodný, musela by si ho dotyčná osoba nechat podepsat od důvěryhodné certifikační autority. To ovšem stojí peníze (19, s. 56-58).

Obr. 13 - Upozornění na nedůvěryhodný ale správný podpis

Pokud důvěřujeme veřejnému klíči a provedli jsme kontrolu ID klíče, otisku a přidruženého mailu, je možné klíč lokálně podepsat, a tak ztvrdit důvěryhodnost veřejného klíče. To se provádí ve správci klíčů přes kontextovou nabídku volbou **podepsat klíč**. Na obrázku Obr. 14 je vidět výpis úspěšného ověření podpisu.

Obr. 14 - Upozornění na důvěryhodný klíč

6.2 Šifrování sériové linky mezi dvěma směrovači

V prostředí síťových zařízení, je šifrované spojení samozřejmostí. Nejde jen o data aplikační vrstvy, která sítí prochází. Je nutné si uvědomit, že k uskutečnění přenosu dat z jedné sítě do druhé, je nutné vybudovat i celý aparát pro komunikaci mezi směrovači. Ty si mezi sebou vyměňují informace o stavu linky, směrovací tabulky dále musí být schopny autentizace. Všechny tyto pakety, nesouvisející s aplikační vrstvou můžeme označit jako **režie**. Ačkoliv koncové uživatele tyto data nezajímají, ve skutečnosti jde o velice důležitá a citlivá data.

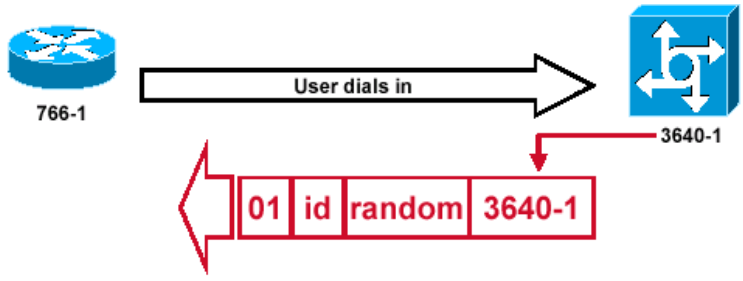
6.2.1 Point-to-Point Protokol

Tento protokol se používá jako standard pro komunikaci mezi dvěma síťovými uzly. PPP je určen pro komunikaci na sériových linkách. Podporovány jsou sériové linky synchronní (např. ISDN) i asynchronní (např. Vytáčené spojení). PPP komunikace probíhá na linkové vrstvě modelu ISO/OSI. Datagramy PPP se zapouzdřují pomocí protokolu HDLC. K navázání, autentizaci, otestování a ukončení spojení se využívá protokolu LCP. Pro komunikaci s vyššími protokoly, jako jsou IP, IPX, AppleTalk, se využívá rodina protokolů NCP. Podpora pro každý vyšší protokol je tvořena modulem, který se zavádí při konfiguraci linky. PPP tak umožňuje jednoduché rozšiřování, které je založené na pouhém přidaní dalšího modulu (10, CCNA4).

6.2.2 Challenge Handshake Authentification Protocol

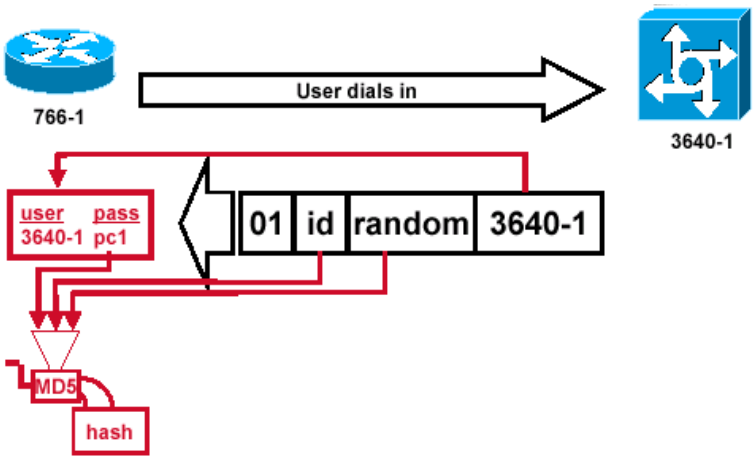
Poté co protokol LCP vyjedná nastavení linky, zahájí se výběr autentizačního protokolu. CHAP je protokol pro prokázání totožnosti na základě symetrického šifrování. Před zahájením komunikace je stanicím nakonfigurován shodný šifrovací klíč, následně ověření probíhá ve třech krocích. Vysvětlení těchto kroků bude předvedeno na příkladě.

1. Stanice 3640-1 (viz Obr. 15) odpovídá a zahajuje proces autentizace vygenerováním náhodného řetězce dat (označen jako random). Pro odpověď se použije paket typu 01 (typ challenge). Dále je přidáno sekvenční číslo (id), které identifikuje konkrétní paket. Nakonec je přidán identifikátor zařízení (3640-1).



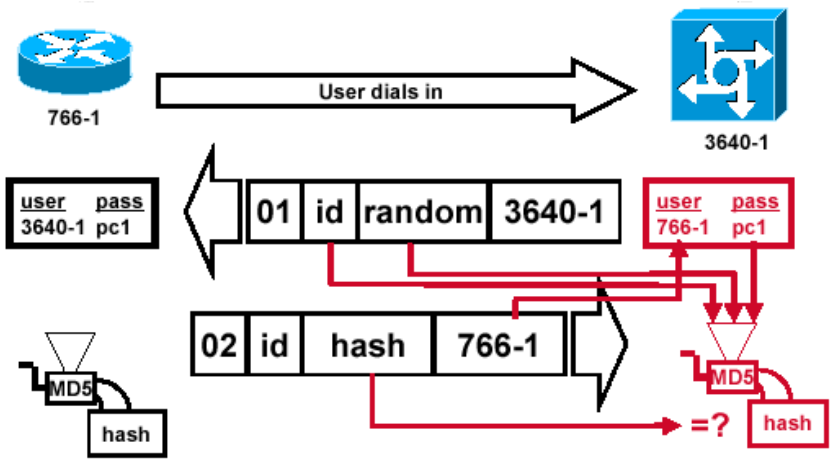
Obr. 15 - Autentizační metoda CHAP, dotaz, zdroj: (13)

2. Jakmile stanice 766-1 obdrží paket typu challenge, ověří zda má uživatele 3640-1 zaznamenaného v lokální databázi uživatelů. Pokud ano, vybere příslušné heslo z databáze, připojí k němu sekvenční číslo paketu a náhodný řetězec. Z těchto dat je následně pomocí funkce MD5 vytvořen haš (viz Obr. 16).



Obr. 16 - Autentizační metoda CHAP, odpově, zdroj: (13)

Odpověď je odesílána s typem paketu 02 (typ response), pole id je kopírováno, dále je uveden vytvořený haš a identifikátor zařízení (viz Obr. 17)

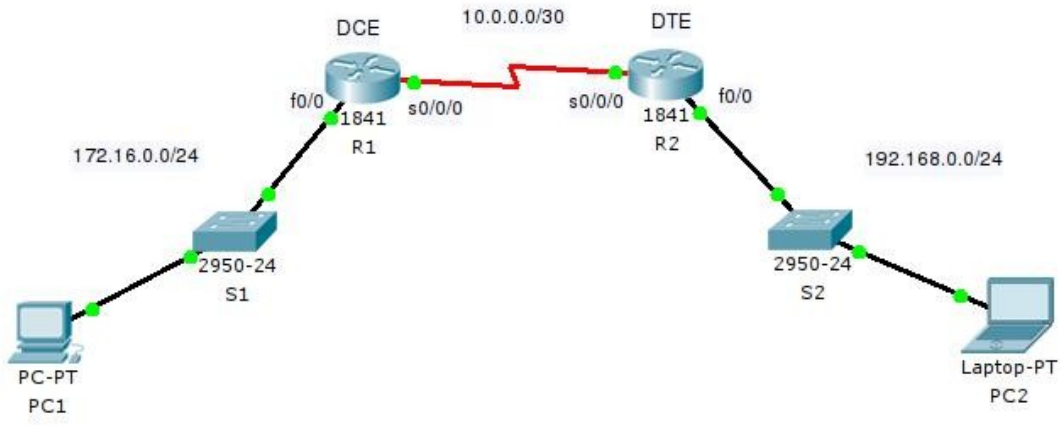


Obr. 17 - Autentizační metoda CHAP, ověření, zdroj: (13)

3. V posledním kroku musí stanice 3640-1 zkontrolovat správnost zaslaného haše. Stejně tak jako druhá stanice, vezme příslušné hodnoty a vytvoří svůj vlastní haš. Tentokrát je ale použit identifikátor stanice 766-1 pro vyhledání záznamu v lokální databázi uživatelů. Na závěr zbývá porovnat oba haše a rozhodnout o úspěchu autentizace.

6.2.3 Konfigurace PPP CHAP na Cisco směrovačích

Konfiguraci je možné vyzkoušet v programu Packet Tracer. Tato aplikace se standardně používá při studiu počítačových sítí ve studijním programu Cisco akademie[10]. Packet Tracer slouží k simulaci počítačových síti a obsahuje i analytické nástroje. Ke konfiguraci se přistupuje pomocí grafických oken nebo klasicky přes příkazový řádek.



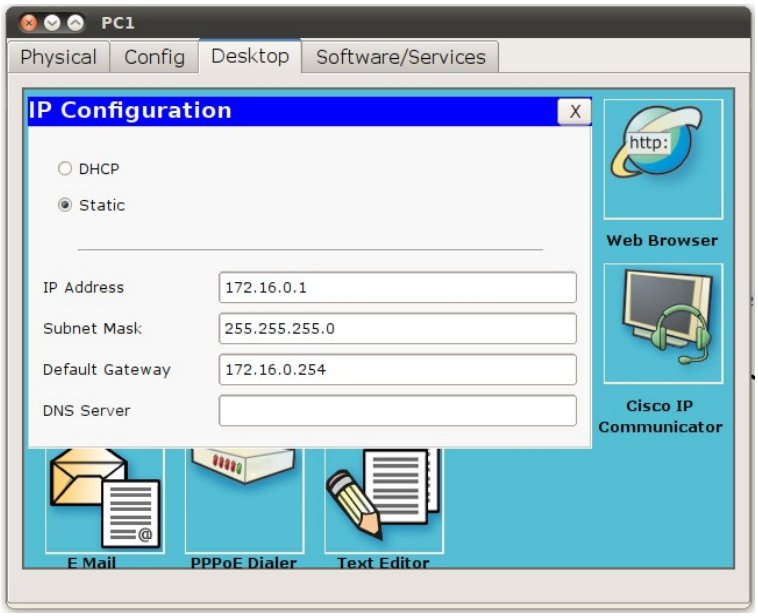
Obr. 18 - Topologie sítě se sériovou linkou

Na výše uvedeném obrázku je znázorněna topologie, na které bude předvedena konfigurace šifrovaného, sériového spojení. Pod každým zařízením je nejprve uveden jeho identifikátor (v rámci aplikace Packet Tracer) a následně jméno zařízení (v rámci této konkrétní topologie). F0/0 a S0/0/0 jsou identifikátory rozhraní (interface) jednotlivých síťových zařízení. DCE označuje zařízení které generuje taktovací frekvenci sériového spoje, na druhé straně tohoto spoje je vždy zařízení DTE, které taktovací frekvenci přizpůsobuje zařízení DCE. Celkově jsou použity 3 adresní segmenty, bližší informace viz Tab. 2.

Tab. 2 - Tabulka adres

Název zařízení	Rozhraní	IP adresa	Maska podsítě	Výchozí brána
R1	S0/0/0	10.0.0.1	255.255.255.252	N/A
	F0/0	172.16.0.254	255.255.255.0	N/A
R2	S0/0/0	10.0.0.2	255.255.255.252	N/A
	F0/0	192.168.0.254	255.255.255.0	N/A
PC1	NIC	172.16.0.1	255.255.255.0	172.16.0.254
PC2	NIC	192.168.0.1	255.255.255.0	192.168.0.254

Pro konfiguraci osobních počítačů je možné využít mini-aplikaci **IP Configuration**, umístěnou na kartě **Desktop** (viz Obr. 19). Jelikož grafické prostředí nedovoluje komunikovat rozsáhlejší nastavení, jako je PPP CHAP, je nutné přes kartu **CLI** přistoupit do příkazového řádku.



Obr. 19 - Konfigurace PC1

Následující příkazy přiřadí směrovači jméno zařízení a nakonfigurují IPv4 adresy pro rozhraní FastEthernet0/0 a Serial0/0/0. Jelikož se jedná o DCE rozhraní, je uveden i příkaz **clock rate**, který nastavuje taktovací frekvenci sériové linky.

```
Router> enable
Router# configure terminal
Router(config)# hostname R1
R1(config)# interface f0/0
R1(config-if)# ip address 172.16.0.254 255.255.255.0
R1(config-if)# no shutdown
R1(config-if)# exit
R1(config)# interface s0/0/0
R1(config-if)# ip address 10.0.0.1 255.255.255.252
R1(config-if)# clock rate 128000
R1(config-if)# no shutdown
```

Obdobná konfigurace platí i pro zařízení R2. Použity musí být správné adresy, které jsou uvedeny v Tab. 2. Na směrovači R2 také nebude příkaz **clock rate**, jelikož se konfiguruje rozhraní DTE.

V této fázi by měli mít počítače přístup na své výchozí brány. Ověření lze provést příkazem **ping**. Co se ale nezdaří je ping na druhou stanici PC. Pro tento účel je třeba nastavit směrování a jelikož se konfigurují Cisco směrovače, je možné využít např. protokolu EIGRP. Základní nastavení je možné provést následujícími příkazy.

```
R1# configure terminal
R1(config)# router eigrp 1
R1(config-router)# no auto-summary
R1(config-router)# network 10.0.0.0
R1(config-router)# network 192.168.0.0
```

Obdobné příkazy opět použijeme i u zařízení R2. Shodné musí být číslo autonomního systému, které se uvádí za slovem **eigrp**. Pro umožnění beztřídního adresování, například v případě, kdy využíváme podsítě, je nutné zadat příkaz **no auto-summary**. Nakonec se uvádí příkazem **network** adresní prostory směrovaných sítí. Za tímto příkazem se uvádí takzvaná **wildcard mask**, která se počítá inverzí bitů z masky podsítě.

V tuto chvíli již je možné komunikovat se stanicemi z druhé sítě. Pro komunikaci mezi směrovači je použit protokol HDLC, který se ve výchozím nastavení používá na Cisco směrovačích. Následující příkazy slouží k nastavení PPP pro sériovou linku mezi směrovači.

```
R1# configure terminal
R1(config)# username R2 password 0 crypto
R1(config)# interface s0/0/0
R1(config-if)# encapsulation ppp
R1(config-if)# ppp authentication chap
```

Obdobné nastavení je požadováno i pro druhý směrovač. Jediným rozdílem je příkaz **username**. Tento příkaz slouží pro zavedení uživatele do lokální databáze uživatelů. Ve výchozím nastavení používají Cisco směrovače k vzájemné identifikaci své **hostname**.[13] Je tedy důležité aby R1 zavedl do lokální databáze uživatelů záznam o uživateli R2 a naopak. Dále se uvádí za slovem **password** metoda zadávání hesla. V tomto případě je zvoleno zadávání hesla ve nešifrované podobě, heslo je tedy **crypto**. V této chvíli je ale heslo uloženo v čisté podobě i v konfiguraci směrovače, a tak je jednoduše zjistitelné po zadání příkazu **show running-config**. Abychom zamezili ukládání hesel do konfiguračního souboru v čisté podobě, je možné využít následujícího příkazu.

```
R1# configure terminal
R1(config)# service password-encryption
```

Pro kontrolu funkčnosti, je možné využít režimu debug, který je možné spustit na několika úrovních.

```
R1# debug ppp authentication
R1# debug ppp negotiation
R1# debug ppp packet
```

Pomocí prvního příkazu je možné sledovat autentizační proces CHAP. Druhým příkazem lze sledovat proces vyjednávání parametrů linky. Třetí příkaz umožňuje nahlížet na pakety PPP na nízké úrovni. Pro ukončení debug módu lze použít standardní příkaz.

```
R1# no debug all
```

7 Závěr

V teoretické části bakalářské práce byly popsány všechny požadované oblasti kryptologie. Při studiu odborné literatury jsem dospěl k názoru, že pro ucelení teoretické části bude vhodné nastínit i problematiku útoků na kryptografické systémy. Uvedeny byly základní postupy vedoucí k prolomení kryptosystému a možnosti obrany proti kryptoanalytickým útokům.

V praktické části bylo cílem první laboratorní práce, předvést praktické využití asymetrického šifrování. Pro splnění tohoto cíle jsem se rozhodl uskutečnit zašifrovanou emailovou komunikaci mezi zařízeními využívající různé operační systémy. Pro tento úkol jsem vybral program Pretty Good Privacy, který je v oboru šifrování emailu označován za standard. Instalace byla intuitivní a potřebná rozšíření byla snadno přístupná. Aplikace Outlook a Thunderbird přistoupili k implementaci PGP odlišně.

V programu Outlook jsou ovládací prvky PGP umístěni na kartě „Doplňky“. To působí zmatečně a uživatel navíc nemá dostatečný přehled a komfort při psaní zprávy. Pro správu klíčů je nutné využít externí aplikace Kleopatra, ta sice zastává svou práci dobře, ale její základní funkce nejsou v aplikaci Outlook přímo přístupné. Dále aplikace Outlook vyžaduje interakci uživatele při přijetí zprávy. Je mu zobrazen dialog s informacemi o stavu ověření a je povinen ho potvrdit. Nevýhodou je, že mu je tento dialog nucen, i když pouze prolistovává přečtené zprávy.

V programu Thunderbird je implementace řešena přidáním položky „OpenPGP“ přímo do nástrojové lišty. Tato položka obsahuje veškeré nástroje, které se vztahují k modulu PGP. Obsahuje také správce klíčů, a tak jsou veškeré potřebné nástroje přístupné přímo z aplikace Thunderbird. O výsledku ověření je uživatel informován pomocí ikony a jednoduchého výpisu pod nástrojovým panelem. Pro bližší informace je možné zobrazit podrobnosti.

V druhé laboratorní práci bylo cílem předvedení autentizační metody CHAP na sériové lince mezi dvěma směrovači. Pro splnění tohoto cíle byla navržena jednoduchá topologie sítě v programu Packet Tracer 5.3.3. Tento program se standardně využívá pro studijní účely v mezinárodním výukovém programu Cisco Akademie. Program umožnil veškeré potřebné prostředky pro simulaci dané úlohy. Pro pochopení procesů probíhajících při autentizaci byla doplněna potřebná teorie, která vysvětluje způsob využití funkce MD5 v tomto autentizačním procesu.

V závěru práce se mi dostala do rukou kniha Velký průvodce infrastrukturou PKI od autorů Dostálek, Vohnoutová a Knotek. Po přečtení několika kapitol jsem byl překvapen složitostí systému, který se skrývá za procesem certifikování veřejných klíčů. Problém certifikace však nebyl v zadání této bakalářské práce, a tak jsem tyto poznatky nemohl uvést. Do budoucna ale budu uvažovat o diplomové práci na téma certifikace.

8 Použité zdroje

1 VORLÍČEK, Jaroslav. Historie kryptografie [online]. [c2003] [cit. 2012-04-24]. Dostupné z: http://kryptologie.uhk.cz/historie.htm. Univerzita Hradec Králové.

2 RYŠÁNKOVÁ, Alžběta. Steganografie [online]. 2.4.2003 [cit. 2012-04-24]. Dostupné z: http://kryptologie.uhk.cz/81.htm. Univerzita Hradec Králové.

3 STALLINGS, William. Cryptography and network security: principles and practice. 3rd ed. Upper Saddle River: Prentice Hall, c1999, 681 s. ISBN 01-311-1502-2.

4 TANENBAUM, Andrew S. Computer networks. 4th ed. New Jersey: Prentice-Hall, c2003, 891 s. ISBN 01-303-8488-7.

5 SHANNON, Claude Elwood. Communication Theory of Secrecy Systems [online]. [cit. 2012-04-24]. Dostupné z: http://netlab.cs.ucla.edu/wiki/files/shannon1949.pdf

6 Algoritmy.net: Příručka vývojáře. [online]. [cit. 2012-04-24]. Dostupné z: http://en.algoritmy.net/article/40382/Letter-frequency-Czech

7 KŘEPELKOVÁ, Helena. Úvodní slovo k novému seriálu o informační bezpečnosti ze všech úhlů: Terminologie. [online]. [cit. 2012-04-24]. Dostupné z: http://www.ictsecurity.cz/serial-o-informacni-bezpecnosti/uvodni-slovo-k-novemu-serialu-o-informacni-bezpecnosti-ze-vsech-uhlu.html

8 MENEZES, Alfred J. Handbook of applied cryptography [online]. Vyd. 1. Boca Raton: CRC Press, 1997, 780 s. [cit. 2012-04-24]. ISBN 08-493-8523-7. Dostupné z: http://cacr.uwaterloo.ca/hac/

9 PINKAVA, Jaroslav. Šifrovat?…..Rozhodně Ano!: Základy kryptografie I., Kryptografie dneška. In: [online]. [cit. 2012-04-24]. Dostupné z: http://crypto-world.info/pinkava/bulletin1.pdf

10 Cisco Networking Academy [online]. [2012] [cit. 2012-05-08]. Dostupné z: http://www.cisco.com/web/learning/netacad/index.html

11 Gpg4win: v2.1.0. [online]. 2011 [cit. 2012-04-24]. Dostupné z: http://www.gpg4win.org/download.html

12 Cisco Networking Academy. Cisco akademie: materiály pro výuku CCNA [online]. [cit. 2012-04-24]. Dostupné z: http://www.cisco.com/web/learning/netacad/index.html

13 Understanding and Configuring PPP CHAP Authentication. [online]. Document ID: 25647, 2012-09-09 [cit. 2012-04-24]. Dostupné z: http://www.cisco.com/en/US/tech/tk713/tk507/technologies_tech_note09186a00800b4131.shtml

14 Design PLUS, our knowledge at your service: Informační bezpečnost. Design PLUS, our knowledge at your service: Úvodní stránka[online]. [c2001-] [cit. 2012-05-08]. Dostupné z: http://www.designplus.cz/nase-reseni/dp-netservice/systemova-a-aplikacni-infrastruktura-sai/informacni-bezpecnost

15 BITTO, Ondřej. Historie kryptologie [online]. [2003] [cit. 2012-04-27]. Dostupné z: http://www.fi.muni.cz/usr/jkucera/pv109/2003/xbitto.htm#_Trochu_teorie_na_začátek. Masarykova univerzita - fakulta informatiky.

16 Smart Card Security, Part 2. Smart Card Basics [online]. © 2010 [cit. 2012-05-08]. Dostupné z: http://www.smartcardbasics.com/smart-card-security_2.html

17 MD5 Hash Generator. Crypo [online]. © 2012 [cit. 2012-05-08]. Dostupné z: http://www.crypo.com/tools/eng_md5.php

18 PINKER, Jiří. Vybrané aspekty moderní kryptoanalýzy. [online]. [2003] [cit. 2012-05-06]. Dostupné z: http://crypto-world.info/klima/2003/st_2003_03_03_07.pdf

19 DOSTÁLEK, Libor, Marta VOHNOUTOVÁ a Miroslav KNOTEK. Velký průvodce infrastrukturou PKI a technologií elektronického podpisu. 2., aktualiz. vyd. Brno: Computer Press, 2009, 542 s. ISBN 978-802-5126-196.

20 INGLIS-ARKELL, ESTHER. 10 Charming, Old-Timey Spy Gadgets. [online]. [cit. 2012-05-06]. Dostupné z: http://io9.com/5865792/10-charming-old+timey-spy-gadgets

21 Public key cryptography: 9.3. Http://www.globus.org/toolkit/ [online]. c2005 [cit. 2012-05-08]. Dostupné z: http://gdp.globus.org/gt4-tutorial/multiplehtml/ch09s03.html