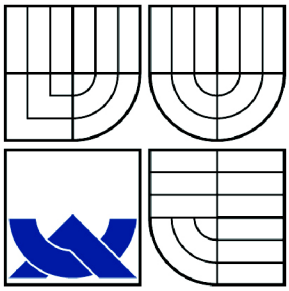
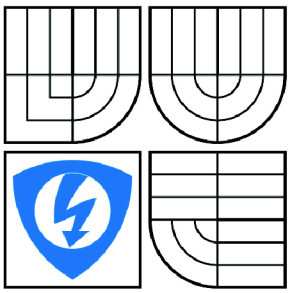


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

DEPARTMENT OF TELECOMMUNICATIONS

ANALÝZA ZABEZPEČENÍ PŘENOSU DAT NA RŮZNÝCH VRSTVÁCH REFERENČNÍHO MODELU OSI

ANALYSIS OF DATA TRANSFER SECURITY ISSUES AT PARTICULAR OSI MODEL LAYERS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

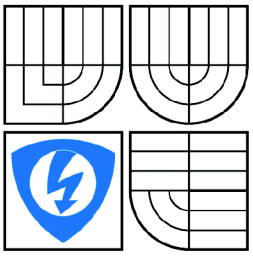
AUTOR PRÁCE — PAVEL KŇAZOVICKÝ

AUTHOR

VEDOUCÍ PRÁCE — Ing. JIŘÍ SOBOTKA

SUPERVISOR

BRNO 2009



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student:	Pavel Kňazovický	***ID:***	102394
Ročník:	3	***Akademický rok:***	2008/2009

NÁZEV TÉMATU:

Analýza zabezpečení přenosu dat na různých vrstvách referenčního modelu OSI

POKYNY PRO VYPRACOVÁNÍ:

Analyzujte protokoly zajišťující zabezpečený přenos dat na jednotlivých vrstvách modelu OSI. Zvolte si jeden z protokolů a proveďte jeho podrobnou analýzu. Navrhněte zabezpečení přenosu dat s využitím těchto protokolů.

DOPORUČENÁ LITERATURA:

[1] Stallings, W.: Cryptography and Network Security: Principles and Practice. Prentice Hall, Englewood Cliffs 2003. ISBN: 0-13-091429-0

[2] Dostálek, L, Kabelová, a.: Velký průvodce protokoly TCP/IP a systémem DNS, Computer press, Brno, ISBN: 978-80-251-2236-5

Termín zadání:	9.2.2009	***Termín odevzdání:***	2.6.2009

Vedoucí práce: Ing. Jiří Sobotka

prof. Ing. Kamil Vrba, CSc.
Předseda oborové rady

UPOZORNĚNÍ:

ANOTÁCIA

Táto práca sa zaoberá analýzou protokolov zaisťujúcich zabezpečený prenos dát. V prvej časti je stručne popísaný referenčný model ISO/OSI. Druhá časť je zameraná na samotné zabezpečené protokoly na jednotlivých vrstvách ISO/OSI modelu, z ktorých je v tretej časti podrobne analyzovaný protokol SSL/TLS. Posledná časť sa venuje často používaným útokom v oblasti počítačových sietí a ich služieb, kde je uvedená aj základná prevencia proti nim.

KĽÚČOVÉ SLOVÁ

analýza, protokol, OSI model, SSL, TLS, sieťové útoky, kryptografia, kryptoanalýza, prevencia

ABSTRACT

The aim of this Bachelor's thesis is the analysis of secured data transfer protocols. The very first part is dedicated to the short description of the reference model ISO/OSI. The second one is focused to the secured protocols at particular layers of ISO/OSI model, of which SSL/TLS protocol is closely analysed in the third part. The last part is about often used attacks in the area of computer networks and their services and the basic protection against them is also mentioned.

KEY WORDS

analysis, protocol, OSI model, SSL, TLS, network attacks, cryptography, cryptoanalysis, prevention

KŇAZOVICKÝ, P. *Analýza zabezpečení přenosu dat na různých vrstvách referenčního modelu OSI*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2009. 59 s. Vedoucí bakalářské práce Ing. Jiří Sobotka.

PREHLÁSENIE

Prehlasujem, že svoju bakalársku prácu na tému „Analýza zabezpečení přenosu dat na různých vrstvách referenčního modelu OSI“ som vypracoval samostatne pod vedením vedúceho semestrálneho projektu a s použitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky uvedené v zozname literatúry na konci práce.

Ako autor uvedeného semestrálneho projektu ďalej prehlasujem, že v súvislosti s vytvorením tohto projektu som neporušil autorské práva tretích osôb, predovšetkým som nezasiahol nedovoleným spôsobom do cudzích autorských práv osobnostných a som si plne vedomý následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona č. 121/2000 Sb., vrátane možných trestnoprávnych dôsledkov vyplývajúcich z ustanovení § 152 trestného zákona č. 140/1961 Sb.

V Brne dňa ....
(podpis autora)

POĎAKOVANIE

Ďakujem vedúcemu bakalárskej práce Ing. Jiřímu Sobotkovi za poskytnutie odborných pripomienok a rád pri riešení tejto práce.

V Brne dňa ....
(podpis autora)

OBSAH

ÚVOD

Počítačové siete, kde samozrejme patrí aj Internet, sa stali každodennou súčasťou nášho života. Stretávame sa s nimi napr. v práci, pri výbere z bankomatu, platení v obchode cez platobný terminál a v neposlednom rade aj doma. Rozmach Internetu spôsobil, že ho môžeme využiť v rôznych smeroch, avšak jeho prostredie je veľmi často nezabezpečené. Nešifrovaný prenos informácií sa stáva ľahkou korisťou pre útočníka.

Práve preto sa každý rozumný užívateľ snaží čo najlepšie zabezpečiť svoje osobné dáta, či citlivé dáta. Je jasné, že pre prenos týchto dát nemôžeme využívať bežné prenosové protokoly, ale protokoly špeciálne vyvinuté pre tento účel. Uvedená problematika prešla značným vývojom. Pôvodné prenosové protokoly boli zamerané len na prenos dát, a nie na ich zabezpečenie. Postupne vznikla potreba zabezpečenia z dôvodu prenosu informácií, ktoré boli dôverného charakteru. Začali sa vyvíjať protokoly, ktoré zaisťovali zabezpečený prenos informácií. Spomínané protokoly sú témou mojej bakalárskej práce. Experti na bezpečnosť neustále zdokonaľujú bezpečnostné techniky a protokoly. Pretože žiadny ochranný systém nie je nedotknuteľný a ľudia sú čoraz vynaliezavejší.

V bakalárskej práci som sa zameral na stručný obsah referenčného modelu ISO/OSI a popis niekoľkých používaných bezpečnostných protokolov, rozdelených podľa vrstvy modelu, na ktorej sa nachádzajú. Zameral som sa hlavne na podrobný rozbor protokolu SSL/TLS, ktorý vytvára privátny kanál pre dva komunikujúce uzly, čím vlastne zabezpečuje prenos dát. Používa sa hlavne na vytvorenie protokolu HTTPS, čím zabezpečuje Internetové aplikácie. V práci sú tiež sú uvedené často používané útoky v oblasti počítačových sietí a ich služieb.

Očakávaným výsledkom práce je materiál, obsahujúci poznatky z okruhu zabezpečenia dát prostredníctvom protokolov vyvinutých k danému účelu a následne zabezpečenie prenosu použitím rozpracovaných protokolov.

1 REFERENČNÝ MODEL ISO/OSI

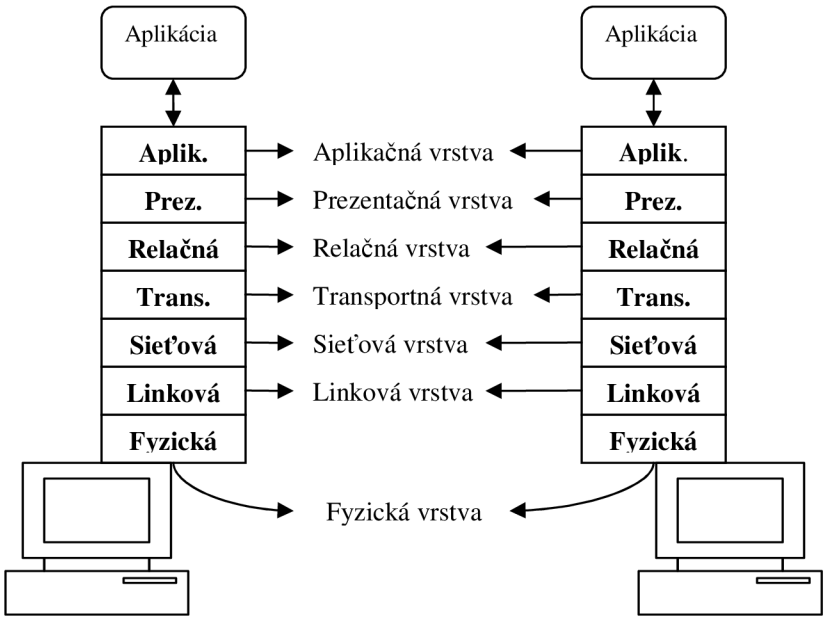
Je abstraktný model popisujúci sieťovú komunikáciu. Bol vytvorený medzinárodnou organizáciou ISO v roku 1979. Pri návrhu sa tvorcovia riadili zásadami, ktoré môžeme zhrnúť do nasledujúcich bodov:

- samostatná vrstva by mala vzniknúť všade tam, kde je potrebný iný stupeň abstrakcie;
- každá vrstva by mala zaisťovať presne vymedzené funkcie, ktoré by mali byť volené s ohľadom na vytvorenie štandardizovaných protokolov s medzinárodnou pôsobnosťou pri ich realizácii;
- rozhranie medzi vrstvami by malo byť volené tak, aby bol minimalizovaný tok dát cez toho rozhranie;
- počet vrstiev by mal byť primerane veľký, aby vzájomne odlišné funkcie nemuseli byť zaraďované do rovnakej vrstvy, a súčasne primerane malý, aby celá architektúra zostala dostatočné prehľadná.

Výsledkom aplikácie týchto princípov bolo vymedzenie siedmich vrstiev a špecifikácie úloh, ktoré by tieto vrstvy mali zaisťovať.

Názov celého štandardu je **Reference Model of Open Systems Interconnection** (referenčný model prepojovania otvorených systémov) a ako norma ISO má číslo 7498. V praxi sa obvykle stretávame so skratkou RM OSI alebo len ISO/OSI, čo súčasne zdôrazňuje jeho vzťah k organizácii ISO (bol však súčasne prevzatý aj organizáciou CCITT ako jej štandard X.200).

Komunikácia medzi dvomi počítačmi je schematicky znázornená na obr. 1.1.



Obr. 1.1 Sedemvrstvová architektúra ISO/OSI

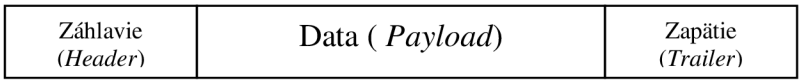
1.1 FYZICKÁ VRSTVA

Fyzická vrstva popisuje elektrické či optické signály používané pri komunikácii medzi počítačmi. Na fyzickej vrstve je vytvorený tzv. fyzický okruh. Na fyzický okruh medzi dvoma počítačmi bývajú často vkladané ďalšie zariadenia, napr. modemy, ktoré modulujú signál na

telefónne vedenie a iné. Ďalšie zariadenie pracujúce na fyzickej vrstve je rozbočovač (*hub*) alebo opakovač (*repeater*).

1.2 LINKOVÁ VRSTVA

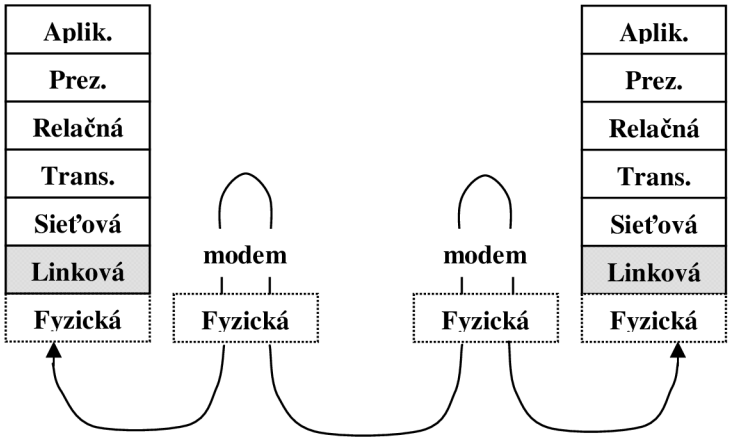
Linková vrstva zaisťuje v prípade sériových liniek výmenu dát medzi susednými počítačmi a v prípade lokálnych sietí výmenu dát v rámci lokálnej siete. Adresácia medzi sieťovými rozhraniami je na základe ich fyzických adries – 48 bitová unikátna adresa daná výrobcom. Základnou jednotkou pre prenos dát je na linkovej vrstve dátový rámec (obr. 1.2). Dátový rámec sa skladá zo záhlavia (*Header*), prenášaných dát (*Payload*) a zápätia (*Trailer*).



Obr. 1.2 Linkový rámec

Dátový rámec nesie v záhlaví linkovú adresu príjemcu, linkovú adresu odosielateľa a ďalšie riadiace informácie. V zapätí nesie okrem iného aj kontrolný súčet z prenášaných dát. Pomocou neho môžeme zistiť, či nedošlo pri prenose k porušeniu dát. V prenášaných dátach je potom spravidla nesený paket sieťovej vrstvy.

Zariadenie pracujúce na linkovej vrstve je prepínač (*switch*) alebo most (*bridge*). Komunikácia na linkovej vrstve je na obr. 1.3.



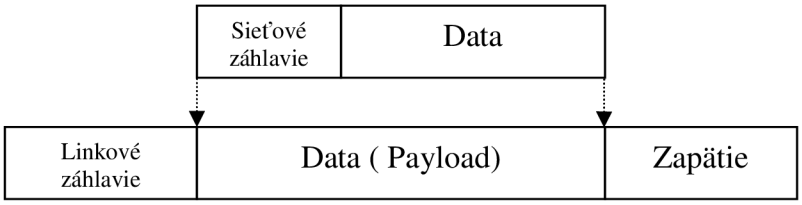
Obr. 1.3 Komunikácia na linkovej vrstve

1.3 SIEŤOVÁ VRSTVA

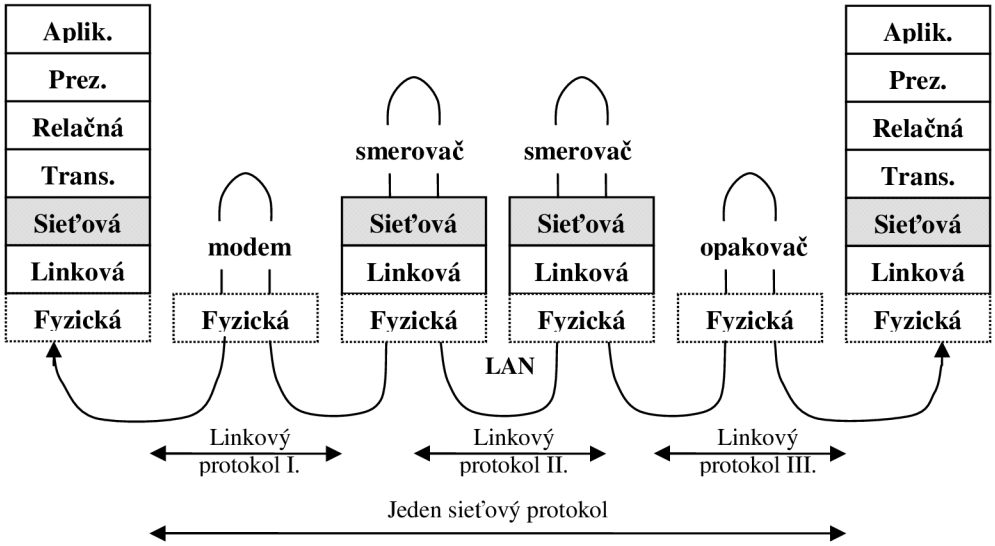
Sieťová vrstva zabezpečuje prenos dát medzi vzdialenými počítačmi v globálnej počítačovej sieti (WAN). Základnou jednotkou prenosu je sieťový paket, ktorý je balený do linkového rámca. Skladá sa zo záhlavia a dátového poľa, so zapätím sa stretávame len zriedka.

Sieťové záhlavie spoločne s dátami sieťového paketu tvoria dáta linkového rámca (obr. 1.4). Adresácia medzi sieťovými rozhraniami je na základe ich logických (IP) adries – 32 bitová adresa. Zariadenie pracujúce na sieťovej vrstve je smerovač (*router*).

V rozsiahlych sieťach (WAN) leží spravidla jeden, prípadne viacero smerovačov. Medzi susednými smerovačmi je na linkovej vrstve vždy priame spojenie. Smerovač vybalí sieťový paket linkového rámca a pred odoslaním ho zabalí do ďalšieho linkového rámca (obr. 1.5).



Obr. 1.4 Sieťový paket a jeho vkladanie do linkového rámca



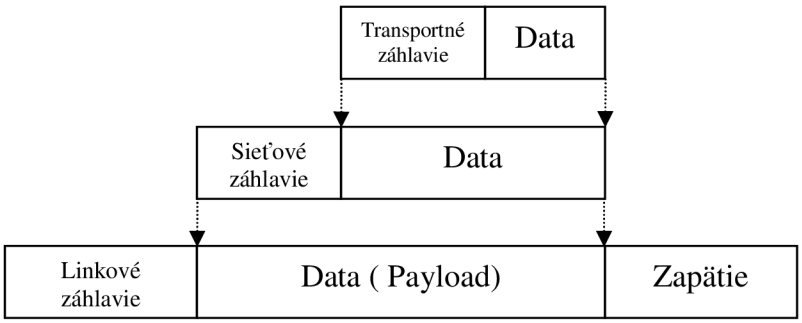
Obr. 1.5 Komunikácia na sieťovej vrstve

Sieťovú vrstvu nezaujíma aké linkové protokoly boli na ceste medzi koncami spojenia použité. Na sieťovej vrstve je jednoznačne v celej WAN adresované sieťové rozhranie.

1.4 TRANSPORTNÁ VRSTVA

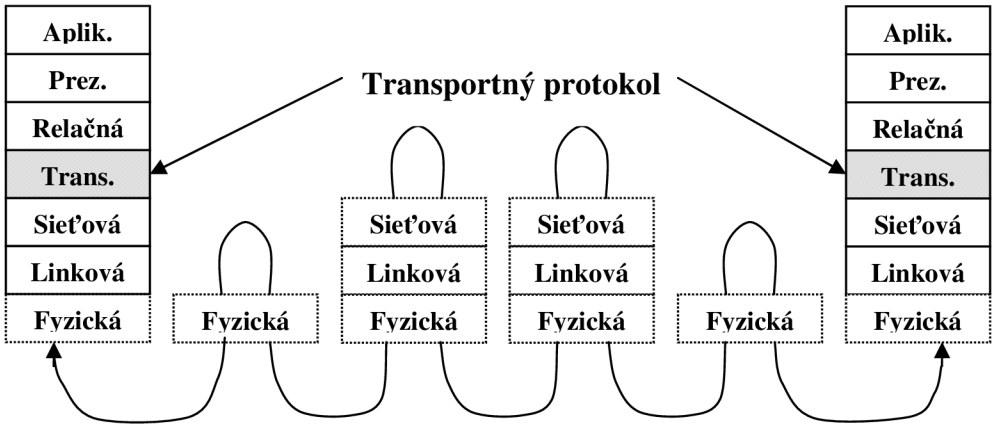
Transportná vrstva sa celkovo spolieha na služby nižších vrstiev. Predpokladá, že spojenie medzi počítačmi je zaistené, a preto sa venuje spojeniu medzi aplikáciami na vzdialených počítačoch.

Jednotkou prenosu je transportný paket (obr. 1.6), ktorý sa tiež skladá zo záhlavia a dátovej časti a prenáša sa v dátovej časti sieťového paketu.



Obr. 1.6 Spôsob vytvorenia transportného paketu

Medzi dvomi počítačmi môže byť aj niekoľko transportných spojení súčasne. Z hľadiska sieťovej vrstvy sú pakety adresované pomocou adresy sieťového rozhrania počítača. Z hľadiska transportnej vrstvy sú adresované jednotlivé aplikácie a tie sú jednoznačne adresované v rámci jedného počítača. Spojenie na úrovni transportnej vrstvy je na obr. 1.7.



Obr. 1.7 Spojenie na transportnej vrstve

1.5 RELAČNÁ VRSTVA

Relačná vrstva má za úlohu nadväzovanie, udržovanie a rušenie relácií (*sessions*) medzi koncovými aplikáciami. V rámci nadväzovania relácie si táto vrstva vyžiada na transportnej vrstve spojenie, prostredníctvom ktorého potom prebieha komunikácia medzi obomi účastníkmi relácie. Ak je potreba túto komunikáciu riadiť (napr. určovať, kto má kedy vysielať, ak nemôžu obe strany súčasne), zaisťuje to práve táto vrstva, ktorá ma na starosti všetko, čo je potrebné k ukončeniu relácie a zrušenie existujúceho spojenia.

Základnou jednotkou je relačný paket, ktorý sa opäť vkladá do transportného paketu. V literatúre, rozoberajúcej túto problematiku, sa vyskytuje obrázok, ktorý zreteľne znázorňuje relačný paket skladajúci sa z relačného záhlavia a relačných dát vloženého do transportného paketu. Avšak, u vrstiev od transportnej nahor to tak byť nemusí. Informácie relačnej vrstvy môžu byť prenášané vo vnútri dát. Ešte zreteľnejšia je táto situácia u prezentačnej vrstvy, ktorá dáta napr. zašifruje, a tým zmení celý obsah paketu.

1.6 PREZENTAČNÁ VRSTVA

Prezentačná vrstva je zodpovedná za reprezentáciu a zabezpečenie dát. Reprezentácia dát môže byť na rôznych počítačoch rôzna. Zabezpečením sa rozumie šifrovanie, zabezpečenie integrity dát, digitálny podpis a iné.

1.7 APLIKAČNÁ VRSTVA

Aplikačná vrstva predpisuje v akom formáte a akým spôsobom majú byť dáta preberané/predávané od aplikačných programov. Príkladom úlohy aplikačnej vrstvy môže byť prenos súborov, elektronická pošta, prípadne správa siete. Programy môžu získať prístup k službám aplikačnej vrstvy pomocou elementov aplikačnej vrstvy.

2 PROTOKOLY ZAISŤUJÚCE BEZPEČNÝ PRENOS DÁT

Bezpečnostný protokol na rozdiel od bežných komunikačných protokolov zaisťuje rôzne bezpečnostné funkcie.

Úlohou bezpečných komunikačných protokolov je zaistiť bezpečnosť informácií prenášaných v komunikačných systémoch, ku ktorým majú prístup aj útočníci. Všetky tieto protokoly sú prakticky kryptografickými aplikáciami. V tejto kapitole sú rozpísané niektoré zabezpečené protokoly a sú rozdelené podľa vrstvy ISO/OSI modelu, na ktorej sa nachádzajú.

Najskôr si ale ozrejmíme základné definície z oblasti zabezpečenia.

2.1 ZÁKLADNÉ DEFINÍCIE Z OBLASTI ZABEZPEČENIA

Kryptológia

Kryptológia je vedecká disciplína, ktorá sa zaoberá ochranou dát pred neoprávneným čítaním. Rozdeľuje sa na dve časti:

- **kryptografiu –** venujúcu sa kódovaniu a šifrovaniu dát;
- **kryptoanalýzu –** venujúcu sa predovšetkým analýze algoritmov a zašifrovaných dát, čiže dešifrovaniu.

Autentizácia

Autentizácia je technika, ktorou si užívateľ overuje identitu iného užívateľa. Tým sa zaručí, že komunikácia je dôveryhodná. Ako jednoduchý príklad môžeme uviesť zadanie hesla pre vstup na serverový počítač. Autentizáciu delíme na jednosmernú (autentizuje sa len klient) a vzájomnú (autentizácia klienta aj servera).

Autorizácia

Autorizácia je proces získavania prístupu k informáciám, funkciám a ďalším objektom. Skladá sa z nasledovných čiastkových procesov:

- autentizácie subjektu;
- vyhľadania v zozname oprávnených subjektov;
- udelenia oprávnenia alebo odoprenia prístupu.

Kontrola integrity

Kontrolou integrity nazývame službu, ktorá zaručí, že prijaté správy sú rovnaké v porovnaní so správami, ktoré boli poslané (bez modifikácie). Modifikácia zahrňuje zmenu, zmazanie, prepísanie, alebo reprodukciu vyslanej správy. Použitím tejto služby si je príjemca istý, že žiaden útočník nenahradil originálnu správu.

Nepopretie (*non-repudation*)

Nepopretie zabraňuje odosielateľovi alebo príjemcovi poprieť, že sa zúčastnil komunikácie. Konkrétne, odosielateľ nemôže poprieť, že správu odoslal a príjemca, že ju prijal.

Šifrovanie

Šifrovanie je proces, pri ktorom je zreteľná správa utajená do nezreteľnej formy za účelom zabezpečenia.

Dešifrovanie

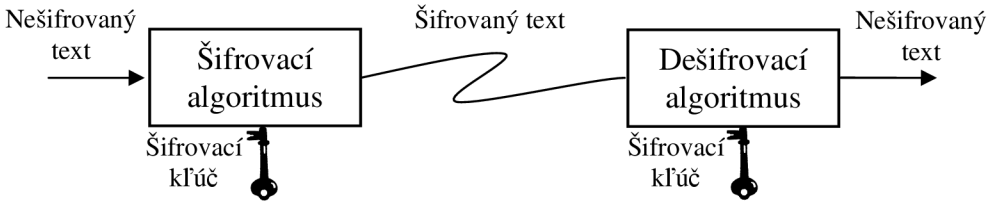
Dešifrovanie je opakom šifrovania. Utajená správa je transformovaná do zreteľnej formy.

Šifrovací kľúč

Šifrovací kľúč je sada inštrukcií, ktoré riadia šifrovací alebo dešifrovací algoritmus. Zvyčajne sú šifrovacie a dešifrovacie algoritmy všeobecne známe a kľúče sú tajné, čo robí komunikáciu bezpečnou.

Symetrické šifrovanie

Symetrické šifrovanie (obr. 2.1) odkazuje na šifrovacie algoritmy s inverznými dešifrovacími algoritmami, ktoré používajú rovnaký šifrovací kľúč. To znamená, že so symetrickým šifrovaním potrebujeme len jeden kľúč pre šifrovanie aj dešifrovanie, pri ktorom ale použijeme inverzný šifrovací algoritmus.



Obr. 2.1 Proces symetrického šifrovania

Využívané algoritmy sú veľmi rýchle. Používajú sa nasledovné algoritmy:

- **DES** (*Data Encryption Standard*) - používa 56 bitový kľúč, mapuje 64 bitovú vstupnú správu na výstupnú správu rovnakej veľkosti;
- **Triple DES** - zvyšuje bezpečnosť DES algoritmu tým, že ho aplikuje 3-krát s tromi rôznymi kľúčmi, používa 128 bitový kľúč, mapuje 64 bitovú vstupnú správu na výstupnú správu rovnakej veľkosti;
- **AES** (*Advanced Encryption Standard*);
- **IDEA** (*International Data Encryption Algorithm*) - používa 128 bitový kľúč, mapuje 64 bitovú vstupnú správu na výstupnú správu rovnakej veľkosti;
- **RC4** (*RSA Data Security*) - prúdová šifra s premennou dĺžkou kľúča, používa sa pri WEP protokole.

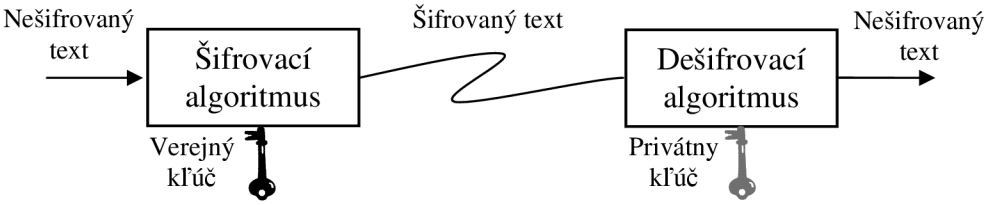
Hlavné bezpečnostné riziko symetrického šifrovania je distribúcia šifrovacieho kľúča.

Asymetrické šifrovanie

Asymetrické šifrovanie (obr. 2.2) odkazuje na šifrovacie algoritmy, ktoré nie sú inverzné (jednosmerné funkcie). Správa zašifrovaná týmto algoritmom vyžaduje dva rôzne algoritmy, jeden pre šifrovanie a druhý pre dešifrovanie. To znamená že potrebujeme aj dva rôzne kľúče (verejný a privátny). Každá strana komunikácie vlastní tento unikátny pár kľúčov. Zo znalosti jedného kľúča nemôžeme odvodiť druhý.

Príklad používaných algoritmov:

- **RSA** (*Rivest-Shamir-Adleman*) - je tisíckrát pomalší ako symetrický algoritmus DES
- **DSS/DSA** (*Digital Signature Standard*)
- **DH** (*Diffie-Hellman*)
- **Knapsack**
- **EC** (*Elliptic Curves*)



Obr. 2.2 Proces asymetrického šifrovania

Odstraňuje nevýhodu symetrického šifrovania s distribúciou šifrovacieho kľúča. Používa sa aj pre digitálny podpis.

Hašovacia funkcia

Hašovacia funkcia je zložitá inverzná funkcia na transformáciu ľubovoľného vstupu na fixne dlhý výstup, akúsi jeho "jedinečnú" skratku, pre rovnaký vstup vždy rovnakú. Tento reťazec sa označuje ako *hash* a jeho dĺžka závisí od použitej hašovacej funkcie. Funkcia môže slúžiť ku kontrole integrity dát, k rýchlemu porovnávaniu dvojice správ, indexovaniu, vyhľadávaniu a pod.

Príklad používaných algoritmov:

- **CRC**
- **MD2, MD4, MD5**
- **SHA-0, SHA-1...**
- **HAVAL**
- **RIPEMD**

PKI (*Public Key Infrastructure*)

PKI je prostredie, umožňujúce ochranu informačných systémov, elektronických transakcií a komunikácie. Zahrňuje celý softvér, všetky technológie a služby, ktoré umožňujú využitie šifrovania s verejným a privátnym kľúčom.

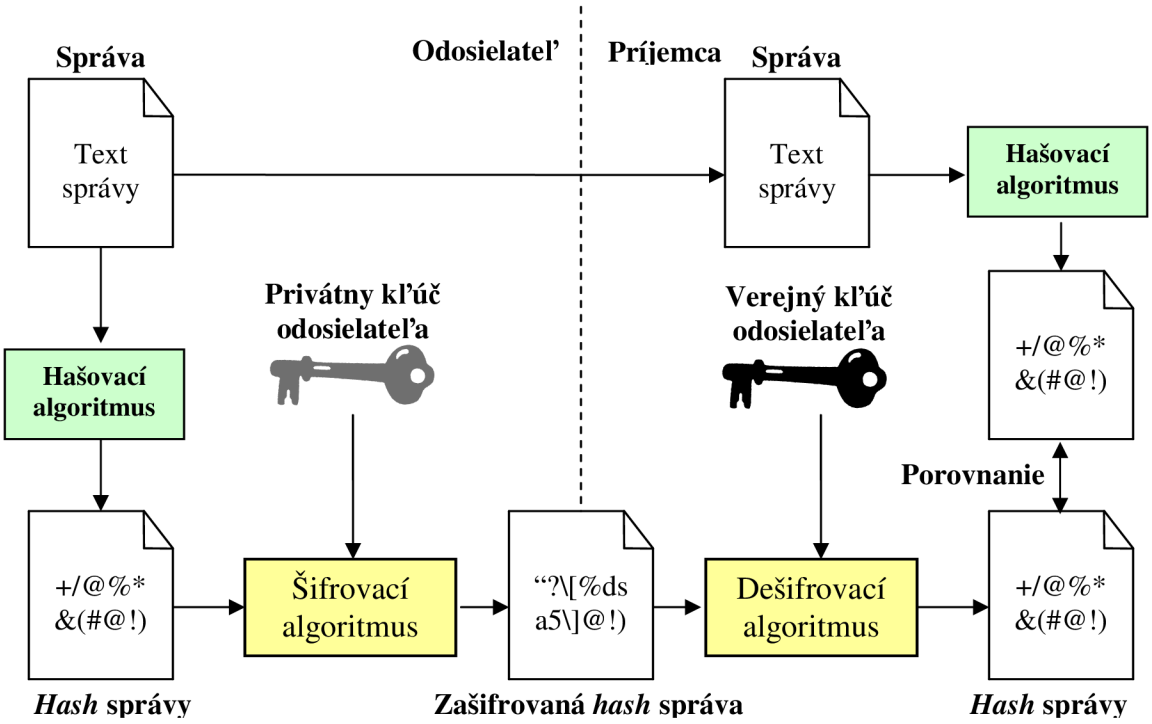
Podporuje rôzne spôsoby ochrany informácií, napr. nasledovné spôsoby:

- autentizáciu prístupu;
- preverovanie integrity správ;
- nepopierateľnosť;
- privátnosť.

Digitálny podpis

Digitálny podpis je metóda pre autentizáciu identity odosielateľa v digitálnej komunikácii. Podobne ako pri písomnom podpise, aj pri digitálnom podpise je hlavným účelom garancia identity odosielateľa (teda, že odosielateľ je skutočne ten, za koho sa vydáva).

Odosielateľ zašifruje správu (alebo *hash*) vlastným privátnym kľúčom a výstup priloží k správe. Príjemca šifrovanú a podpísanú správu rozšifruje verejným kľúčom odosielateľa a overí podpis (rovnaký *hash* správy). Ak *hash* je rovnaký, tak prenos prebehol v poriadku, ak nie, tak správa bola modifikovaná.



Obr. 2.3 Princíp digitálneho podpisu

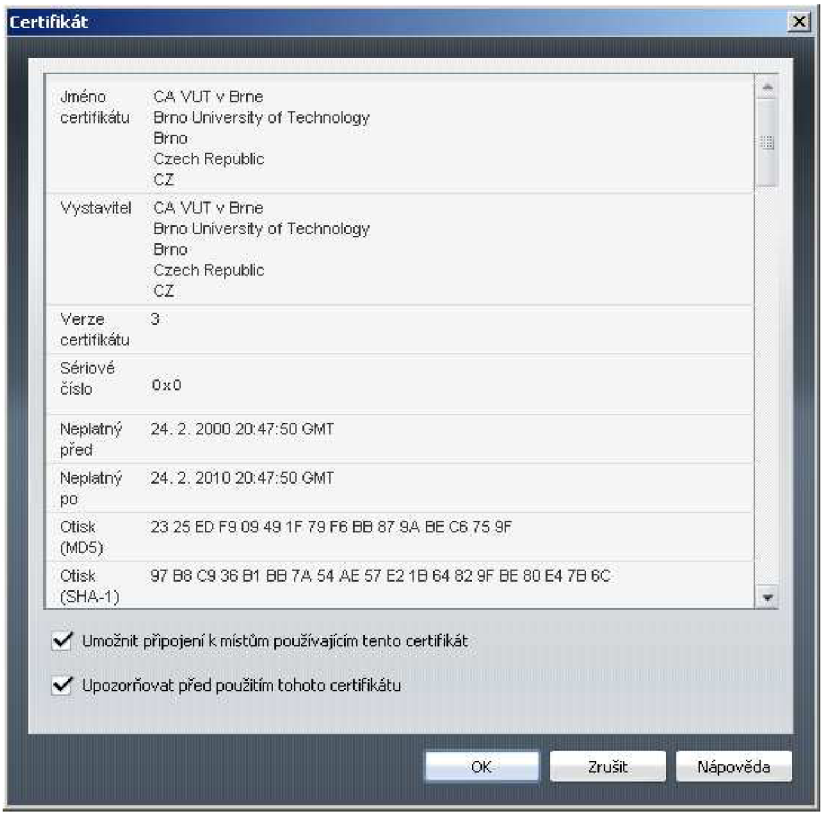
MAC (*Message Authentication Code*)

MAC je kryptografická funkcia podobná hašovacím funkciám. Rozdielom je, že funkcia nie je len výsledkom spracovaných dát, ale aj kľúča. Obe strany komunikácie sa dohodnú na tajnej hašovacej funkcii, alebo na privátnom kľúči používanom so známou hašovacou funkciou. Príjemca správy si vypočíta MAC z prijímanej správy a porovná s MAC, ktorú dostal. Používa sa pre symetrické digitálne podpisy.

Certifikát

Certifikát sa používa k overeniu, overuje, či elektronicky doručený verejný kľúč skutočne patrí udávanej osobe a nie útočníkovi, ktorý sa za danú osobu iba vydáva. Certifikát je určité tvrdenie digitálne podpísané dôveryhodnou stranou. Týmto tvrdením môže byť napr. to, že určitá osoba má určitý verejný kľúč alebo, že daný software je skutočne originálnym produktom danej firmy a pod. Dôveryhodnou stranou je spravidla určitá inštitúcia, ktorá má všeobecnú dôveru. Táto inštitúcia sa nazýva certifikačná autorita (CA) a jej verejný kľúč je bezpečne doručený všetkým užívateľom. Pomocou tohto kľúča potom užívatelia môžu overovať všetky certifikáty CA a tým sprostredkovane aj všetky tvrdenia osôb, ktoré majú certifikát na svoje verejné kľúče.

Príklad certifikátu webového servera je na obr. 2.4.



Obr. 2.4 Príklad certifikátu

Certifikačná autorita

Certifikačná autorita je dôveryhodný poskytovateľ certifikačných služieb. Jeho úlohy sú nasledovné:

- vydávať certifikáty, viesť ich správu;
- zverejňovať záznamy vydaných certifikátov, ktoré boli zbavené platnosti;
- overovať platnosť certifikátov a iné.

2.2 FYZICKÁ VRSTVA

Zabezpečené protokoly na úrovni fyzickej vrstvy nie sú špecifikované, ale z bezpečnostného hľadiska je aktívom sama linka a jej schopnosť prenášať dáta. Najznámejšie hrozby sú:

- **prerušenie komunikácie -** najjednoduchší typ útoku, zabrániť mu môžeme zálohovaním linky;
- **rušenie komunikácie –** najčastejšie si ho spôsobujeme sami, použitím nevhodných materiálov pre vedenie, vytváraním segmentov dlhších ako stanovuje daná technológia, chybnými spojmi a konektormi alebo u bezdrôtových technológií atmosférickými vplyvmi, v prípade výskytu rušenia je potrebné odhaliť zdroj a zamedziť ďalšiemu rušeniu;
- **odpočúvanie –** môže byť veľmi užitočné pre správcu siete pri vyhľadávaní chyby, na tento účel sa používajú rôzne softvérové analyzátory (*Wireshark, tcpdump* a iné) alebo špecializované hardvérové analyzátory, problém nastáva, ak tieto analyzátory využije útočník, musíme sa, teda, zamerať na fyzickú ochranu rozvodov a samozrejme používať šifrovanie prenášaných dát na vyšších vrstvách modelu ISO/OSI;

- **modifikácia prenášaných dát –** ak chce útočník cieľavedome zmeniť prenášané dáta, musí vložiť svoje zariadenie do vedenia, podobne ako pri vkladaní hardvérových analyzátorov.

2.3 LINKOVÁ VRSTVA

Bezpečnosť na 2. vrstve ISO/OSI modelu je na úrovni Point-to-Point komunikácie, čiže medzi dvoma susednými komunikujúcimi účastníkmi. Zahrňuje bezpečný prenos rámcov vrátane:

- autentenzácie užívateľa,
- šifrovania rámcov,
- kontroly integrity dát.

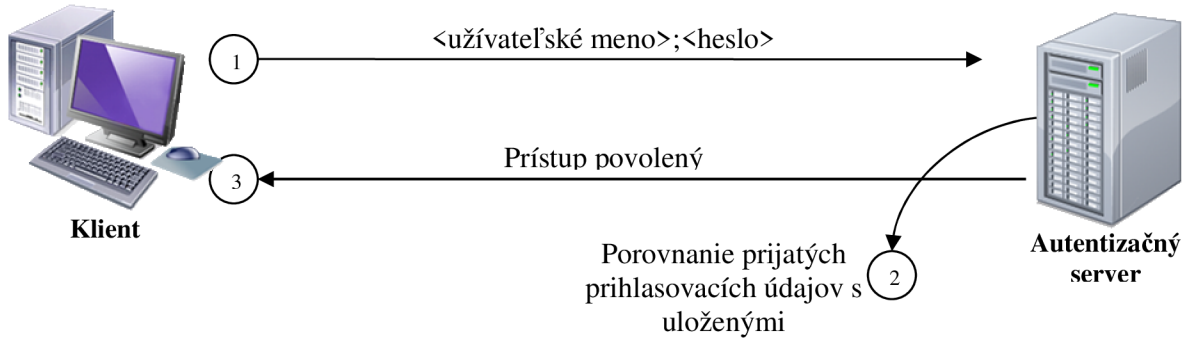
Z hľadiska princípu modelu ISO/OSI, zaisťuje určitý stupeň bezpečnosti pre všetky vyššie vrstvy, pre ktoré je táto bezpečnosť transparentná.

Bezpečnosť na úrovni linkovej vrstvy je zameraná na oblasti drôtových a najmä bezdrôtových sietí, pretože pri nich je okrem iného možné:

- komunikáciu jednoducho odpočúvať,
- aktívne sa vložiť do komunikácie.

2.3.1 PAP

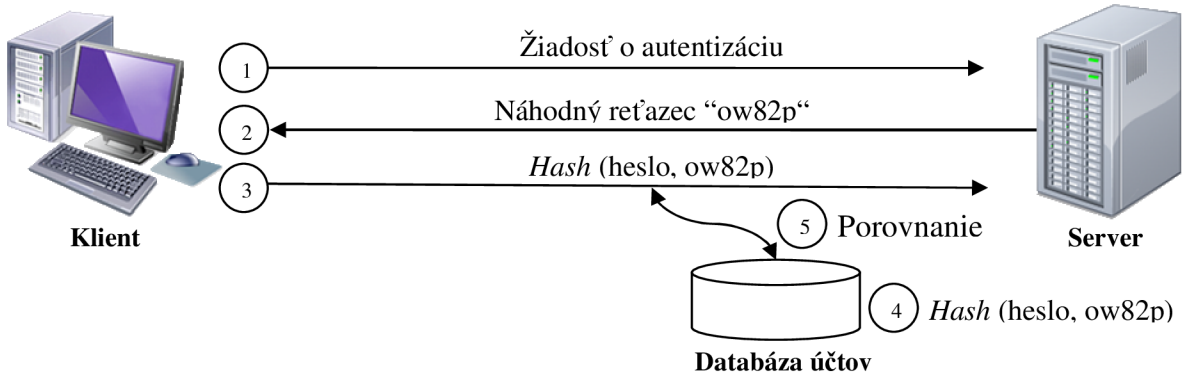
PAP (*Password Authentication Protocol*) je základný autentizačný mechanizmus definovaný v RFC 1334 „PPP Authentication Protocols“. Je podporovaný PPP (*Point-to-Point Protocol*) protokolom. Prenáša login/heslo v otvorenom tvare, čo je najmenej bezpečný spôsob autentizácie, takže určite by sme ho nemali používať (obr. 2.5).



Obr. 2.5 Priebeh PAP autentizácie

2.3.2 CHAP

CHAP (*Challenge Handshake Authentication Protocol*) je autentizačný štandard pre WAN a je podporovaný protokolom PPP. Autentizácia užívateľa (obr. 2.6) prebieha bez nutnosti prenosu ich hesiel, pretože sa používa hašovací algoritmus MD5 pre prenos *hash*-u hesla.



Obr. 2.6 Priebeh CHAP autentizácie

Predstavuje minimálnu bezpečnosť, pretože algoritmus MD5 je prelomiteľný slovníkovým útokom. Microsoft vytvoril ako rozšírenie CHAP protokol MS-CHAP.

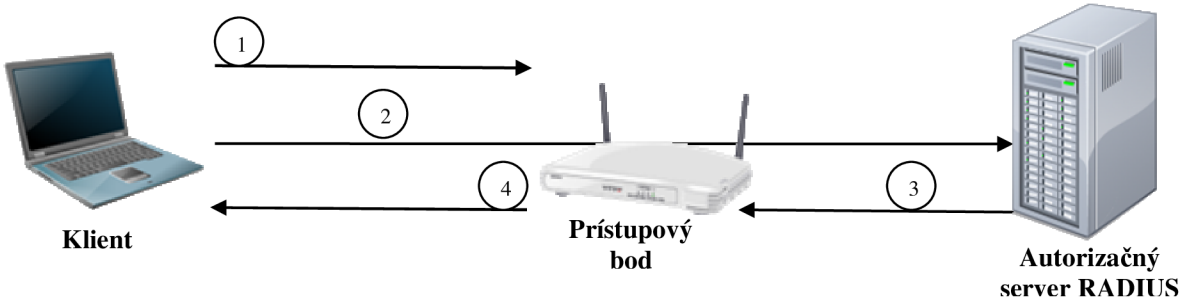
2.3.3 802.1x

Tento protokol umožňuje autentizáciu na portoch (chápeme, ako fyzické porty, napr. na prepínači). Blokuje sa celá komunikácia na danom porte, až kým sa klient neautentizuje prostredníctvom údajov uložených na back-end serveri, ktorým je typicky RADIUS. 802.1x vychádza z protokolu PPP, ktorý je obmedzený tým, že autentizáciu má založenú len na kombinácii užívateľského mena a hesla. Protokol EAP (*Extensible Authentication Protocol*) bol vytvorený ako rozšírenie protokolu PPP a jeho cieľom bolo vytvoriť všeobecnú platformu pre rôzne autentizačné metódy (napr. heslá, certifikáty, tokeny, PKI, čipové karty, biometrika atď.). Štandard tým, že je otvorený, zaisťuje, že kedykoľvek v budúcnosti sa budú môcť použiť mechanizmy, ktoré v súčasnosti nie sú známe.

Protokol 802.1x umožňuje používať EAP na metalických alebo bezdrôtových sieťach. Medzi základné komponenty patrí **žiadateľ, autentizátor** a **autentizačný server** .

V bezdrôtovej sieti zabezpečuje overovanie prístupový bod pre klientov na základe ich výzvy. Pomocou zoznamu, alebo externého autentizačného systému založeného na serveri Kerberos alebo RADIUS (*Remote Authentication Dial In User Service*). K prístupu na bezdrôtovú sieť má možnosť iba overený používateľ. Autentizácia protokolom 802.1x je znázornená na obr. 2.7.

K šifrovaniu dátovej komunikácie sa pre každé autentizované zariadenie používajú dynamické kľúče, ktoré sú známe len danému zariadeniu a majú obmedzenú životnosť. Využívajú sa k šifrovaniu rámcov na danom porte až kým sa zariadenie neodhlási alebo neodpojí.



Obr. 2.7 Autentizácia protokolom 802.1x

2.3.4 PPTP

PPTP (*Point-to-Point Tunneling Protocol*) je protokol vychádzajúci z protokolu PPP pomocou ktorého sa môžeme pripojiť do virtuálnych privátnych sietí (VPN). Tým môže napr. firma rozšíriť svoju podnikovú sieť o privátne tunely vedené po verejnom Internete.

Protokol PPTP preberá dáta od protokolu PPP a zapuzdrí ich do IP paketov, ktoré prenesie cez tunel siete VPN v bežnom Internete. Podporuje šifrovanie a kompresiu IP paketov. Pri zapuzdrení využíva protokol GRE (kapitola 2.4.1).

Vytvorenie tunela PPTP prebieha v dvoch krokoch:

1. Užívateľ sa najskôr prihlási do siete pomocou bežného spojenia PPP.
2. Spustí sa klient PPTP, ktorý vytvorí riadiace spojenie so serverom na porte TCP/1723, čím sa vytvorí zabezpečený tunel.

Pre naviazanie komunikácie s PPTP tunelom sa používajú dva typy informácií: *riadiace správy* a *dátové pakety*. K svojej činnosti využíva PPTP služieb protokolov CHAP alebo PAP.

Nevýhodou protokolu PPTP je, že nedefinuje akým spôsobom sa má spracovávať autentizácia a šifrovanie dát (napr. implementácia postavená na autentizácii PAP nebude schopná komunikovať s implementáciou postavenou nad CHAP).

Analýza PPTP protokolu Breucom Schneierom[1] v skratke hovorí, že autentizácia je veľmi slabá a ľahko napadnuteľná slovníkovým útokom a tiež proti šifrovaniu môžeme viesť ďalšie útoky. Celý dokument je prístupný na [9].

Ako alternatívu môžeme použiť protokol IPsec (kapitola 2.4.2).

2.3.5 L2TP

L2TP (*Layer 2 Tunneling Protocol*) je rozšírením protokolu PPTP a slúži k zaisteniu činnosti VPN nad verejným Internetom.

Najdôležitejšie dva komponenty protokolu L2TP sú:

- prístupový koncentrátor LAC – zariadenie, kde je fyzicky zakončený vytáčaný hovor;
- sieťový server LNS – zakončuje a prípadne autentizuje dátový prúd PPP.

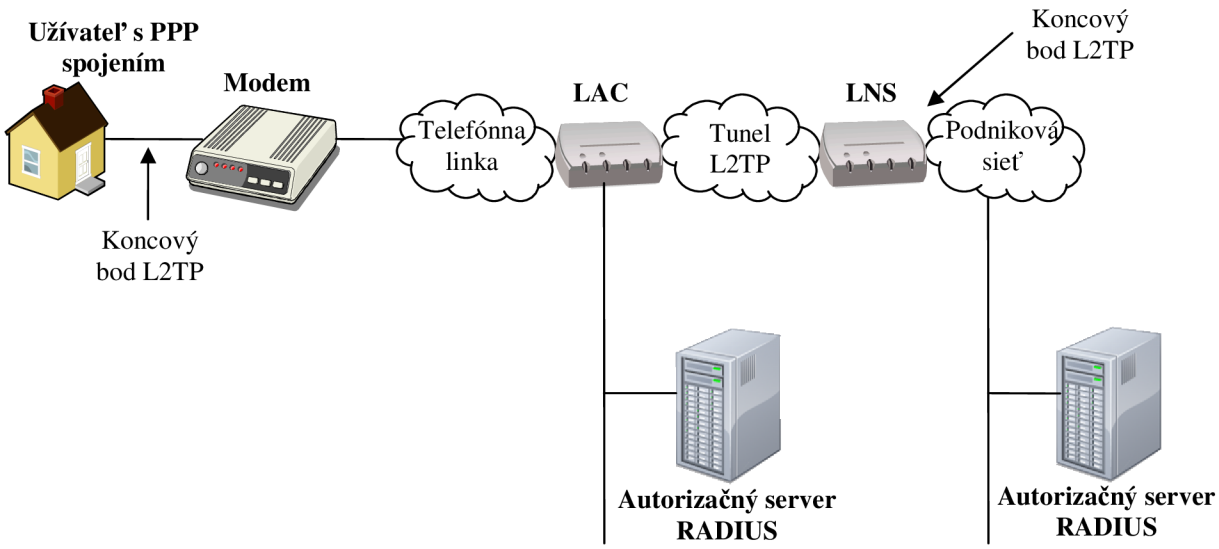
Oproti protokolu PPTP má protokol L2TP radu zmien, a to napr.:

- šifrovanie dát nastupuje už pred procesom spojenia PPP;
- L2TP využíva šifrovací algoritmus DES alebo 3DES;
- L2TP vyžaduje autentizáciu počítača (nielen užívateľa), a tá je definovaná jeho certifikátom.

Medzi výhody L2TP nesporne patrí podpora QoS (pri sieťach postavených na zariadeniach Cisco). Šifrovanie má na starosti protokol IPsec, ktorý zaisťuje aj autentizáciu každého jednotlivého paketu, integritu a dôvernosť dát, ochranu proti opakovaniu relácie.

Na obrázku 2.8 je znázornená najbežnejšia architektúra pre implementáciu sietí L2TP.

[1] Breuce Schneier je americký spisovateľ a medzinárodne uznávaný expert na počítačovú bezpečnosť. Je autorom niekoľkých kníh o počítačovej bezpečnosti.

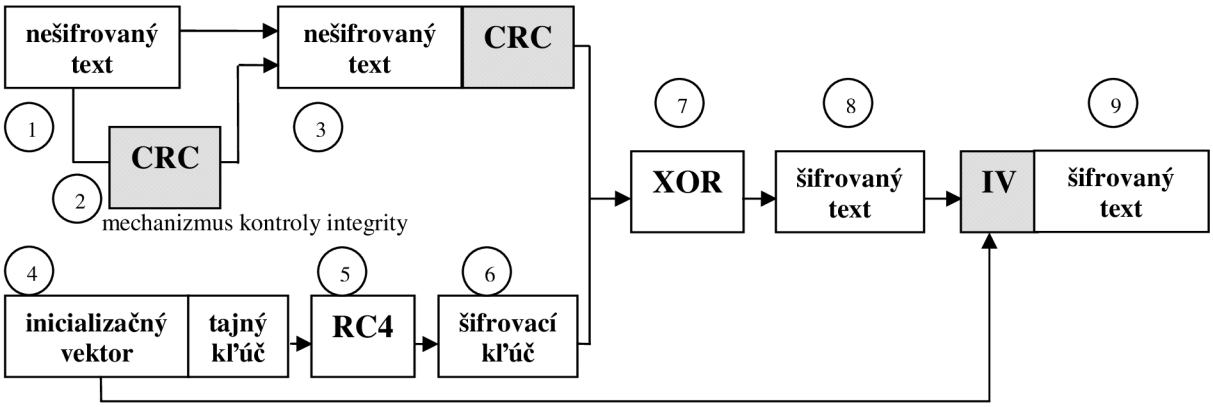


Obr. 2.8 Architektúra siete L2TP

2.3.6 WEP

Skratka WEP (*Wired Equivalent Privacy*) je protokol štandardu 802.11, ktorý ako prvý zaisťoval bezpečnosť. Účelom protokolu nie je celkové zabezpečenie siete, ale ochrana dát pred pasívnym a nechceným odpočúvaním sieťovej komunikácie. Algoritmus WEP nešifruje hlavičku 802.11, identifikátor siete, ani inicializačný vektor (IV).

Protokol zabezpečuje komunikáciu medzi WiFi zariadeniami až na úroveň prístupového bodu (AP), ale za ním už bezpečnosť nezaistí. WEP používa symetrickú prúdovú šifru RC4, teda šifru s tajným kľúčom. Šifra RC4 spočíva vtom, že sa odosielaná správa šifruje podľa nejakého kľúča (zvyčajne to býva slovo alebo sekvencia znakov) a na cieľovom bode sa správa pomocou tohto kľúča zasa dešifruje. Prebieha to tak, že sa kľúč expanduje v pseudonáhodný kľúčovací tok (*keystream*) s rovnakou dĺžkou akú má šifrovaná správa. Generátor pseudonáhodných čísel PRNG sa stará o „pseudonáhodnosť“. PRNG je vlastne zostava pravidiel, z ktorých sa kľúč rozšíri na dĺžku správy do kľúčovacieho toku. Samotné šifrovanie prebieha tak, že na šifrovanej hodnote sa prevedie logická operácia XOR (exkluzívny logický súčet) s kľúčovacím tokom, dešifrovanie prebieha rovnako. Obe strany komunikácie, medzi ktorými má byť komunikácia šifrovaná, musia obsahovať rovnaké pravidlá PRNG a musia poznať tajný kľúč, ktorého problémom je, že WEP nijako nerieši jeho automatickú distribúciu. Celý proces šifrovania je znázornený na obr. 2.9.



Obr. 2.9 Šifrovanie protokolom WEP

V súčasnosti existuje viac dĺžok šifrovacieho kľúča, konkrétne výrobcovia udávajú dĺžky 64 a 128. V skutočnosti je to ale tak, že WEP kľúč má dĺžku 40 bitov, pred ktoré sa predsadí 24 bitov inicializačného vektora (IV), ktorý sa práve používa pre pseudonáhodnosť kľúčovacieho toku. Dokopy je to teda 64 bitov. Podobné je to aj pri dĺžke 128. Pre inicializačný vektor (IV) je vždy vyhradených 24 bitov.

Protokol WEP je možno považovať za odstrašujúci príklad návrhu zabezpečenia komunikačného systému. Tvorcovia protokolu neodborne aplikovali kryptografické ochrany a výsledkom ich snaženia bolo zabezpečenie s množstvom slabín. Použitie WEP sa neodporúča ani pre domácnosti. Ak je kľúč už používaný, je potrebné ho pravidelne meniť (napr. so spojením s 802.1x).

Druhá verzia protokolu WEP2 odstraňuje niektoré pôvodné chyby. Rozširuje inicializačný vektor (IV) a zosilňuje 128-bitové šifrovanie. Používa sa väčšinou na zariadeniach, ktoré výkonovo nestačia na novšie šifrovania. Je stále ľahko prelomiteľný, útočníkovi zaberie prienik len viac času.

2.3.7 WPA

WPA (*Wi-Fi Protected Access*) označuje protokol, ktorý vznikol ako reakcia na vážne bezpečnostné nedostatky objavené v predchádzajúcom protokole WEP. Bol ohlásený v roku 2003 alianciou WiFi a je to vlastne kompromisné riešenie, pretože niektoré časti štandardu 802.11i už boli hotové (802.1x, TKIP) a iné ešte nie (AES, zabezpečená deautentizácia a disasociácia). WPA je podmnožinou štandardu 802.11i, ktorý sa dá implementovať prostredníctvom aktualizácie softvéru a firmvéru. Rieši ako šifrovanie (TKIP), tak aj autentizáciu (802.1x).

WPA rieši problémy protokolu WEP prostredníctvom mechanizmov TKIP (*Temporal Key Integrity Protocol*) a 802.1x nasledovne:

TKIP rieši nasledovné slabiny:

- útok opakovaním – možnosť opakovaného použitia hodnoty inicializačného vektora (IV);
- podvrhnutie – inicializačný vektor (IV) požíva 32 bitovú lineárnu hodnotu CRC, s ktorou sa dá manipulovať;
- útoky založené na kolízii – kolízie inicializačného vektora (IV);
- útoky na slabé kľúče – šifra RC4 je napadnuteľná útokom FMS.

Protokol 802.1x rieši nasledovné slabiny:

- chýbajúca správa kľúčov;
- chýbajúca podpora „pokročilých“ autentizačných metód (tokeny, čipové karty, biometrika, jednorazové heslá a iné);
- chýbajúca identifikácia a autentizácia užívateľov;
- chýbajúca centralizovaná autentizácia a autorizácia.

2.3.8 WPA 2

Tento protokol implementuje povinné prvky štandardu 802.11i, ktorý je založený na šifrovaní pomocou šifry AES (*Advanced Encryption Standard*) v rámci autentizačného rámca EAP.

Šifra AES bola navrhnutá ako náhrada za šifru RC4 a ponúka rôzne režimy činnosti. V štandarde 802.11i sa používa čítačový režim s protokolom CCM (*Counter mode encryption with CBC-MAC*), označovaný aj ako AES-CCMP. Čítačový režim zaisťuje šifrovanie, CBC-

MAC potom zaisťuje autentizáciu a integritu dát. Šifra AES je symetrickým kľúčom, takže text šifruje aj dešifruje rovnakým zdieľaným tajným kľúčom. Na rozdiel od šifry RC4, ktorá šifruje lineárne každý bajt funkciou XOR s náhodnou postupnosťou, AES pracuje s blokmi o veľkosti 128 bitov a preto je označovaná ako bloková šifra. CCM obsahuje nový algoritmus MIC (*Message Integrity Code*), ktorý zaisťuje, aby nedošlo k modifikácii prenášaných dát. Výpočet tohto algoritmu je založený na inicializačných hodnotách vychádzajúcich z inicializačného vektora (IV) a ďalších hlavičkových informácií. MIC pracuje v 128 bitových blokoch a počíta sa cez jednotlivé bloky až na koniec originálnej správy.

Čítačový režim šifrovania pomocou AES sa výrazne odlišuje od predošlých (WEP/TKIP, RC4). Výstupom šifry je po inicializácii len 128 bitový blok. Celý text sa rozdelí na 128 bitové bloky, na ktorých sa postupne prevádza logická operácia XOR so 128 bitovým, vždy nanovo generovaným výstupom AES tak dlho, pokiaľ sa nezašifruje celá pôvodná správa. Nakoniec sa čítač vynuluje, na hodnote MIC sa prevedie logická operácia XOR, ktorá sa následne pridáva na koniec rámca. Výsledkom toho je oveľa silnejšia šifra, ktorá ale vyžaduje výkonnejší hardvér, preto nie je kompatibilná so všetkými bezdrôtovými zariadeniami.

2.4 SIEŤOVÁ VRSTVA

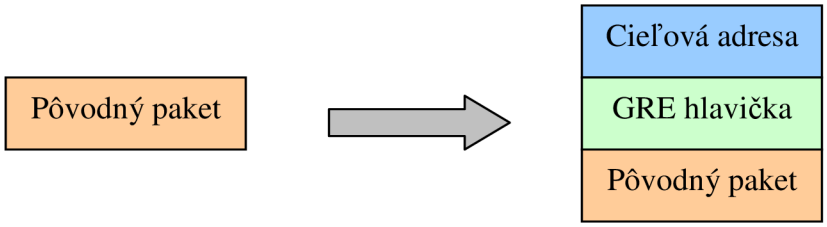
Bezpečnosť na úrovni 3. vrstvy ISO/OSI modelu je zameraná na zabezpečenie komunikácie medzi dvomi vzdialenými uzlami v Internete.

2.4.1 GRE

GRE (*Generic Routing Encapsulation*) je tunelovací protokol vytvorený firmou Cisco určený k zapúzdreniu paketov a vytvoreniu virtuálnych privátnych tunelov medzi Cisco smerovačmi.

GRE tunel je najčastejšie používaný spôsob klasického tunelovania pre spojenie medzi zdrojovým a cieľovým smerovačom. Tieto tunely sú budované smerovačmi ako vstupné a výstupné body *backbone* siete pre jednotlivé časti VPN.

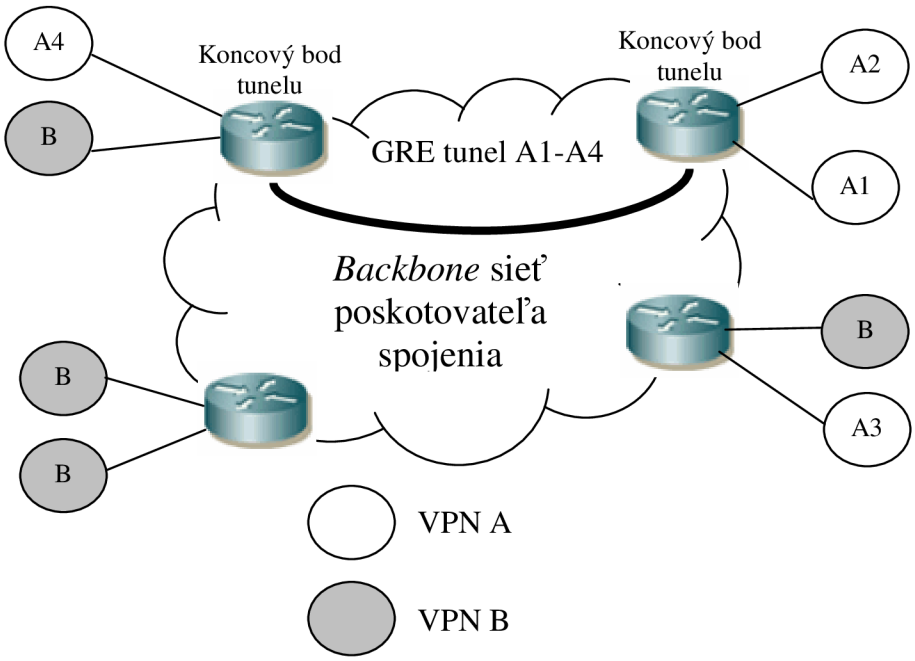
Špeciálne zapúzdrené pakety (obr. 2.10) obsahujú prídavnú GRE hlavičku a cieľovú adresu odpovedajúcu smerovaču na konci tunela.



Obr. 2.10 Zapúzdrený paket protokolom GRE

Vytvorené tunely sú väčšinou typu Point-to-Point, čiže existuje len jedna zdrojová a jedna cieľová adresa. Existujú ale aj firemné implementácie umožňujúce typ Point-to-Multipoints, čím vzniká výhoda možnosti oddelenia adresácie.

Všetky prístupové a zároveň aj koncové body do *backbone* siete vytvorených tunelov používajú adresáciu a smerovanie spoločnej *backbone* siete. Príklad GRE tunelu je uvedený na obr. 2.11.



Obr. 2.11 Príklad GRE tunelu

2.4.2 IPsec

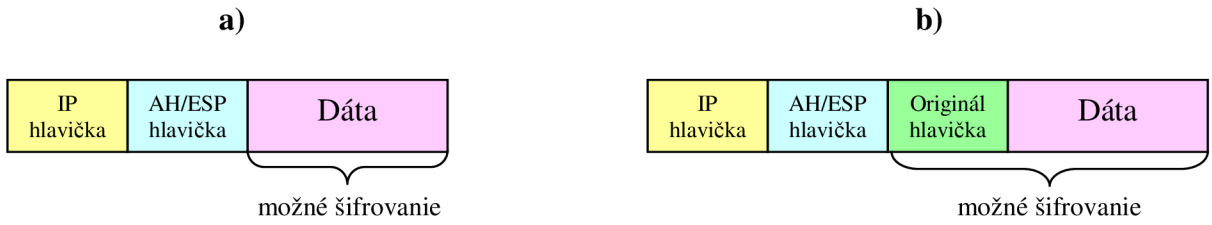
IPsec (*Internet Protocol security)* je bezpečnostné rozšírenie IP protokolu (platí pre verziu IPv4 aj IPv6). Toto rozšírenie je nezávislé na protokoloch vyššej vrstvy TCP/UDP. Je definovaný v desiatkach RFC, ale jeho základné sú 2401 a 2411. Tento protokol vytvára logické kanály SA (*Security Agreements*), ktoré sú jednosmerné, a tak sa pre full-duplex prenos používajú dva SA kanály.

Bezpečnostné rozšírenie IPsec zahrňuje:

- **autentizáciu –** pri prijatí paketu môže dôjsť k overeniu, či vyslaný paket odpovedá odosielateľovi a či odosielateľ vôbec existuje;
- **šifrovanie -** obe strany sa na začiatku dohodnú na forme šifrovania paketu, potom dôjde k zašifrovaniu celého paketu okrem IP hlavičky, prípadne celého paketu a bude pridaná nová IP hlavička (napr. pomocou šifrovania DES, 3DES, AES);
- **integritu** (napr. pomocou hašovacích algoritmov MD5, SHA).

IPsec zabezpečuje spojenie, a to vytvorením šifrovacieho tunelu medzi dvoma koncovými zariadeniami. Používa dva režimy šifrovania:

- **Transportný mód** (obr. 2.12a) – pôvodná hlavička je zachovaná (upravuje sa len dátová časť). Má nižšie nároky na prenosové pásmo a behom prenosu môžeme uplatňovať nadštandardné funkcie (napr. QoS). Najčastejšie použitie je pri autentizácii vzdialených klientov VPN.
- **Tunelovací mód** (obr. 2.12b) – pôvodná hlavička a dáta IP paketu je zabalená a chránená v novo vytvorenom IP pakete, ktorý obsahuje IP adresu príjemcu a odosielateľa z transportnej IP siete. Má vyššie nároky na prenosové pásmo a môžu ho využívať protokoly AH aj ESP.



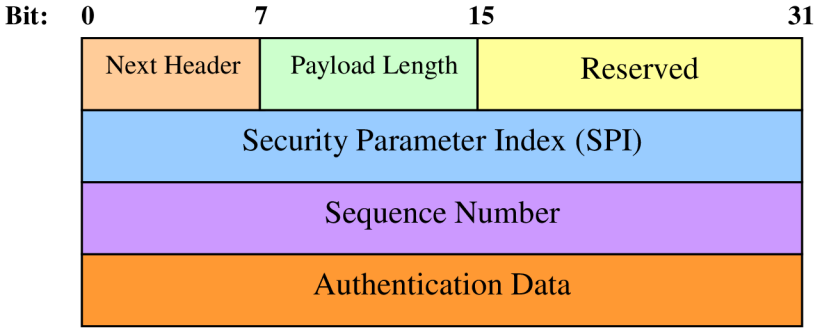
Obr. 2.12 Porovnanie paketu: a)transportný mód b) tunelový mód

IPsec je zostavený prakticky z troch čiastkových protokolov: AH, ESP a ISAKMP.

2.4.2.1 AH

Protokol AH (*Athentication Header*) poskytuje autentizáciu a aj možnosť ochrany proti zopakovaniu relácie. Jeho služby sú obmedzené na časť hlavičky IP a rozšírenej hlavičky, ale nezaisťuje už šifrovanie dát – pomocou hašovacieho algoritmu vytvorí len *hash* správy. Autentizačná hlavička sa vkladá do samostatných chránených dát.

Protokol AH môžeme využívať samostatne alebo aj so spojením s protokolom ESP.



Obr. 2.13 Hlavička protokolu AH

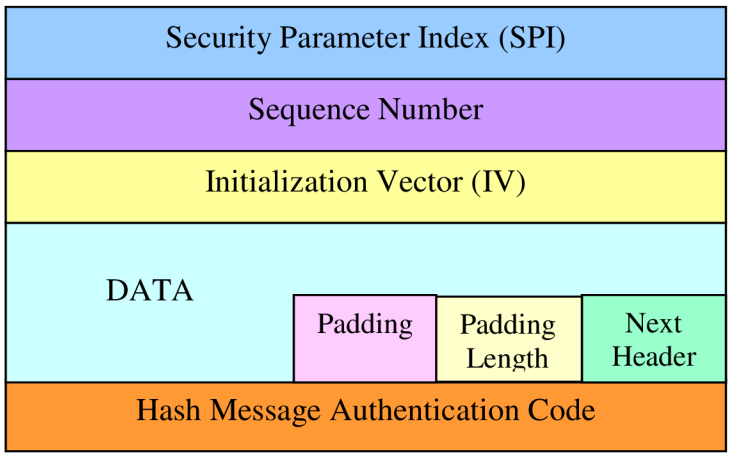
Štruktúra hlavičky AH (obr. 2.13) protokolu pozostáva z nasledovných častí:

- *Next Header* (8 bitov) – určuje protokol prenášaných dát;
- *Payload Length* (8 bitov) – určuje veľkosť AH paketu;
- *Reserved* (16 bitov) – pre budúce využitie;
- *SPI* (32 bitov) – spoločne s *Next Header* a cieľovou IP adresou identifikuje SA;
- *Sequence Number* (32 bitov) – poradové číslo paketu v danom SA spojení, chráni pred útokom opakovaním, pretože ak čítač dosiahne maximum je spojenie zrušené a musí sa vytvoriť nové;
- *Authentication Data* – obsahuje tzv. ICV (*Integrity Check Value*), ktorý sa počíta cez celý datagram.

2.4.2.2 ESP

ESP (*Encapsulating Security Payload*) beží na portoch TCP/50 a TCP/51. Zaisťuje dôvernosť a ochranu dát s voliteľnými službami autentizácie a detekcie opakovaním relácie. Zabezpečuje autenticitu originálu, integritu a dôvernosť.

Protokol ESP môžeme využívať samostatne alebo aj so spojením s protokolom AH.



Obr. 2.14 Hlavička protokolu AH

Štruktúra hlavičky ESP (obr. 2.14) protokolu je nasledovná:

- *SPI* (32 bitov) - spoločne s *Next Header* a cieľovou IP adresou identifikuje SA;
- *Sequence Number* (32 bitov) – poradové číslo paketu v danom SA spojení, chráni pred útokom opakovaním, pretože ak čítač dosiahne maximum je spojenie zrušené a musí sa vytvoriť nové;
- *Data* – pole premennej dĺžky obsahujúce prenášané dáta, niektoré šifrovacie algoritmy vyžadujú inicializačné dáta (*Initialization Vector*);
- *Padding* – niektoré šifrovacie algoritmy vyžadujú zarovnanie;
- *Pad Length* – dĺžka zarovnania (0 – 255);
- *Next Header* – určuje protokol prenášaných dát.

2.4.2.3 ISAKMP

Protokol ISAKMP (*Internet Security Association and Key Managment Protocol*) popisuje fázu dohody o spojení v IPsec.

Súčasťou protokolu je taktiež štandard IKE (*Internet Key Exchange*), ktorý definuje postup pre dohodu bezpečnostných parametrov a postup pre potvrdenie vierohodnosti kľúčov. Je to obojsmerný protokol a vytvára bezpečný komunikačný kanál medzi obom zariadeniami, ktoré sa dohodnú na šifrovacom, hašovacom algoritme, autentizačnej metóde a prípadných informáciách o skupine. Výmena kľúčov je založená na algoritme *Diffie-Hellman*. Proti útoku Man-in-the-Middle sa používa vylepšená verzia STS.

2.4.2.4 Analýza

Podrobnú analýzu IPsec protokolu uskutočnil Breuce Schneier, ktorá v skratke hovorí, že na jednej strane je IPsec oveľa lepší ako predchádzajúce bezpečnostné protokoly (napr. PPTP, L2TP, atď), ale na druhej strane je veľmi zložitý, čo má za následok aj príliš veľa slabých stránok. Celý dokument je prístupný na [10].

2.5 RELAČNÁ A PREZENTAČNÁ VRSTVA

Bezpečnosť na 5. a 6. vrstve ISO/OSI modelu je zameraná na zabezpečenie ľubovoľnej aplikácie. Patrí tu protokol SSL/TLS, ktorý je podrobne popísaný v 3. kapitole bakalárskej práce.

2.6 APLIKAČNÁ VRSTVA

Otázku bezpečnosti na najvyššej vrstve ISO/OSI modelu rieši už každá aplikácia samostatne.

2.6.1 SSH

Sieťový protokol SSH (*Secure Shell*) umožňuje bezpečnú komunikáciu medzi dvomi počítačmi pomocou transparentného šifrovania a voliteľnej komprimácii prenášaných dát. SSH pracuje na porte TCP/22 a zahrňuje dve základné oblasti bezpečnej komunikácie:

- autentizáciu oboch účastníkov komunikácie,
- šifrovanie a integritu prenášaných dát, ktoré sú prenášané medzi počítačmi cez nezabezpečenú vonkajšiu sieť.

V počítačovej terminológii je SSH používaný ako názov prenosového protokolu aj ako názov programu sprostredkujúceho spojenie. Názov *Secure Shell* bol odvodený z existujúceho programu *Rsh*, ktorý má podobné funkcie, ale nie je zabezpečený.

Jeho prvá verzia SSH-1 vznikla na Helsinskej Technickej univerzite v roku 1995 a navrhol ju Tatu Ylőnen. V roku 1996 vyvinul vylepšenú verziu protokolu SSH-2, ktorá je nekompatibilná s predošlou. Ako novinky v druhej verzii sú napr. zvýšená bezpečnosť výmeny kľúčov (*Diffie-Hellman key-exchange*) alebo prísna kontrola integrity dát. Novou vlastnosťou v tejto verzii je tiež možnosť spustiť ľubovoľný počet *shell*-ov vo vnútri jedného SSH spojenia.

Protokol SSH je používaný ako bezpečná náhrada starších protokolov a ponúka aj nové vlastnosti. Sú nasledovné:

- náhrada protokolu *Telnet* – práca na vzdialenom počítači cez nezabezpečenú sieť;
- náhrada protokolu *Rlogin* – prihlásenie na vzdialený počítač;
- náhrada protokolu *Rsh* – spúšťanie príkazov na vzdialenom počítači;
- tunelovanie spojenia;
- presmerovanie TCP portov a X11 spojení zabezpečeným kanálom;
- bezpečný prenos súborov pomocou protokolov SFTP alebo SCP.

Niekoľko príkladov použitia SSH:

- prihlásenie k vzdialenému počítaču

```
ssh uzivatel@adresa.pocitaca.cz
```

- zabezpečené spúšťanie vzdialených príkazov

```
ssh adresa.pocitaca.cz ~/adresar/adresar2
```

- zabezpečený prenos súborov

```
scp subor uzivatel@adresa.pocitaca.cz:
```

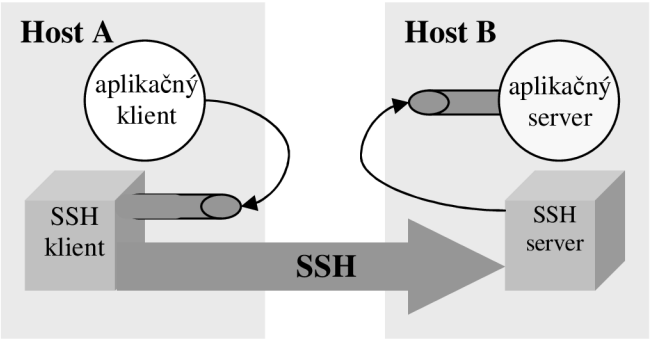
SSH ako program je dnes bežne používaný pre vzdialenú prácu a pre vzdialenú správu. Väčšinou sa spojuje s sshd (*SSH deamon*) pre naviazanie spojenia. Sshd rozhoduje podľa svojho nastavenia: či spojenie príjme a akú formu autentizácie bude požadovať, prípadne na ktorom porte ma načúvať. SSH klient/server (*SSH deamon*) je dostupný na skoro akejkoľvek platforme.

2.6.1.1 SSH tunel

Cieľom SSH tunela je zabezpečiť aj iné aplikácia ako napr. email, web a pod. Tunel môžeme použiť pri dvoch druhoch smerovania, konkrétne:

- **miestne smerovanie** (obr. 2.15) – všetky dáta vložené na lokálnom počítači na port 1234 budú smerované na port 23 vzdialeného počítača;

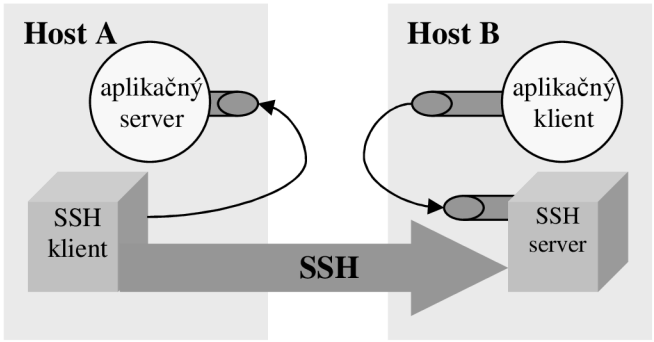
```
ssh -L 1234:localhost:23 user@ssh_server
```



Obr. 2.15 Príklad miestneho smerovania pri SSH tuneli

- **vzdialené smerovanie** (obr. 2.16) – všetky dáta vložené na vzdialenom počítači na port 1234 budú smerované na port 23 lokálneho počítača.

```
ssh -R 1234:localhost:23 user@ssh_server
```



Obr. 2.16 Príklad vzdialeného smerovania pri SSH tuneli

Medzi výhody SSH tunelu patrí samozrejme zabezpečenie komunikácie, ďalej tunelovanie viac portov z rôznych destinácií, automatické forwardovanie paketov a ďalšie. Naopak medzi nevýhody patrí možnosť zriadenia tunelov len pre TCP komunikáciu a pomalšie spojenie vďaka zabezpečeniu.

2.6.1.2 SCP

SCP (*Secure Copy*) je protokol, ktorý slúži k bezpečnému prenosu súborov medzi dvomi počítačmi v počítačovej sieti pomocou protokolu SSH (ten zaisťuje šifrovanie aj autentizáciu). SCP má obmedzené možnosti, a tak je nahradzovaný komplexnejším protokolom SFTP.

Protokol SCP je podobný protokolu *Rcp* z BSD (*Berkeley Software Distribution*), ale na rozdiel od neho sú dáta pri prenose šifrované. Tým znemožňuje odpočúvaním získať z prenášaných dát citlivé informácie (napr. prihlasovacie mená, heslá i samotné prenášané dáta).

Pri kopírovaní súboru na vzdialený počítač je v rámci protokolu SCP možné predať aj ďalšie informácie o súbore (oprávnenie, časy modifikácie a iné), čo je oproti protokolu FTP (*File Transfer Protocol*) významným vylepšením.

Pri kopírovaní súboru zo vzdialeného počítača sú druhej strane komunikácie zasielané požiadavky, ktoré server vybavuje (kopírovanie súboru alebo celej adresárovej štruktúry), čo môže spôsobiť vážny bezpečnostný problém, ak sa pripojíme k serveru, ktorý je napadnutý útočníkom.

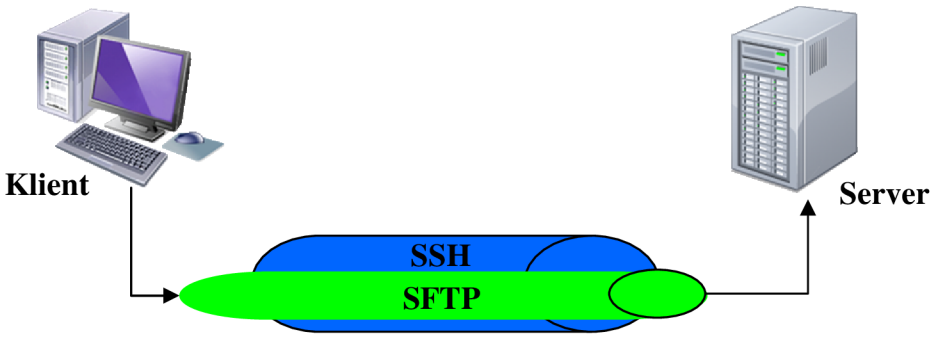
2.6.1.3 SFTP

SFTP (*SSH File Transfer Protocol*) sa používa pre bezpečný prenos súborov pomocou počítačovej siete.

SFTP bol navrhnutý pracovnou skupinou IETF (*Internet Engineering Task Force*) ako multiplatformný. Protokol taktiež sám o sebe nezaisťuje šifrovanie ani autentizáciu, zvyčajne k zaisteniu týchto služieb využíva protokol SSH-2. Je však navrhnutý tak, aby mohol byť použitý aj akýkoľvek iný protokol.

Protokol SFTP je možné použiť aj nad protokolom SSH-1, ale v tomto prípade je narušená nezávislosť na architektúre počítača. Protokol SSH-1 nepodporuje subsystémy, a tak pripojujúci klient musí poznať plnú cestu k programu SFTP servera, aby si mohol sám spustiť serverovú časť.

Na rozdiel od protokolu SCP ponúka SFTP široké možnosti pre doplňujúce operácie so súbormi (podobne ako FTP), takže ho môžeme označiť aj za jednoduchý vzdialený súborový systém. Umožňuje pokračovať v prerušených prenosoch, vypisovať adresáre, aj odstraňovať súbory na vzdialenom počítači. Príklad použitia je na obr. 2.17.



Obr. 2.17 Ukážka použitia protokolu SFTP

2.6.2 HTTPS

HTTPS (*Hyper Text Transport Protocol Secure*) je zabezpečená verzia protokolu HTTP, ktorý je komunikačný protokol webu. Zvyšuje bezpečnosť pred odpočúvaním alebo sfalšovaním dát. Nie je to úplne iný protokol, dáta sú prenášané pomocou HTTP, ale sú šifrované pomocou SSL/TLS, čo zaručuje ochranu proti odpočúvaniu i útokom MITM. HTTPS implicitne komunikuje prostredníctvom portu TCP/443 (u obyčajného protokolu HTTP je to TCP/80).

Pre komunikáciu pomocou HTTPS musí najskôr server vlastniť certifikát. Certifikát musí byť podpísaný certifikačnou autoritou CA, ktorá zaručí, že vlastník certifikátu sa nevydáva za niekoho iného. Internetové prehliadače sú väčšinou vybavené podpisovými certifikátmi najväčších podpisových autorít (napr. THAWTE, VeriSign a iné).

Najslabším miestom tohto protokolu je práve závislosť vysokej bezpečnosti na digitálne podpísaných certifikátoch, pretože bez podpísaného certifikátu je zraniteľný útokom „Man-in-the middle“. V praxi sa však často stretávame s nepodpísanými certifikátmi, čím sa stráca bezpečnosť, ktorú HTTPS ponúka.

2.6.3 S/MIME

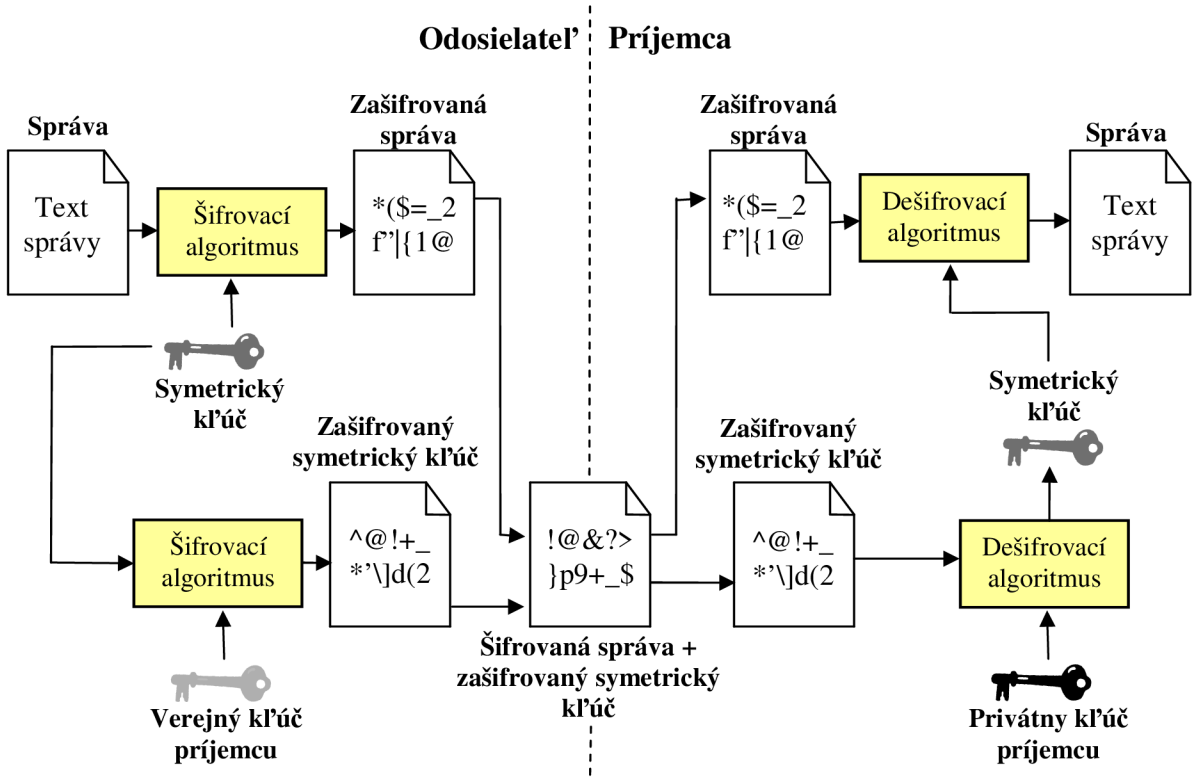
S/MIME (*Secure MIME*) je e-mailová služba založená na formáte MIME (*Multipurpose Internet Mail Extensions*), ktorý umožňuje prenos binárnych dát, ale samotný neobsahuje žiadne bezpečnostné mechaniky. Na rozdiel od toho S/MIME poskytuje utajenie, autentizáciu a integritu.

S/MIME je založené na symetrickom a asymetrickom šifrovaní, hašovacích funkciách a certifikátoch (X.509). Obsah správy sa šifruje symetrickou šifrou (RC2, DES). Pre bezpečnú distribúciu symetrického kľúča sa ten šifruje asymetrickou šifrou (RSA).

2.6.4 PGP

PGP (*Pretty Good Privacy*) je volne šíriteľný program na ochranu správ posielaných elektronickou poštou. Jeho autorom je Philip Zimmermann. Pre každú správu sa vygeneruje náhodný kľúč (relačný kľúč), ktorý je jednorazový a po prenesení a dešifrovaní správy je zničený. Postup pri použití PGP (obr. 2.18) je nasledovný:

1. Odosielateľ si po vytvorení správy, ktorú chce zašifrovať, vygeneruje náhodný kľúč, ktorý je použitý len pre túto správu.
2. Správa je zašifrovaná vygenerovaným symetrickým kľúčom algoritmus (napr. IDEA, AES). Kľúč je potom zašifrovaný pomocou asymetrickej šifry (napr. RSA) verejným kľúčom príjemcu a je predradený zašifrovanej správe.
3. Príjemca dešifruje kľúč svojim privátnym kľúčom a potom dešifruje správu samotnú



Obr. 2.18 Princíp zabezpečenia správy pomocou PGP

Posledné verzie programu PGP sú už hybridné a umožňujú, aby okrem PGP certifikácie boli verejné kľúče opatrené certifikátom poskytovateľa certifikačných služieb. Certifikačná politika s poskytovateľmi certifikačných služieb splňuje štandard X.509v3. Certifikáty verejných kľúčov vydané poskytovateľovi certifikačných služieb sú súčasťou distribúcie verejných kľúčov.

Bruce Schneider analyzoval e-mailové bezpečné protokoly (S/MIME, PGP, atď) proti útoku CCA (*Chosen Ciphertext Attack*). Jedná sa o útok s možnosťou voľby šifrovaného textu. Jeho odporúčania môžeme zaradiť do nasledujúcich bodov:

- všetky šifrované správy by mali byť podpísané, poprípade nereagovať na nepodpísané správy citáciami týchto správ;
- e-mailový dešifrovací softvér by mal ukladať všetky relačné kľúče, ktoré boli poslané od každého užívateľa a varovať, ak by sa nejaký zhodoval, pretože je to veľmi málo pravdepodobné, užívateľ by mal byť veľmi opatrný pri odpovedaní na takú správu;
- ako neimplementované riešenie pre vývojárov podporuje riešenie, ktoré pridá *hash* správy k originálnemu textu pred šifrovaním, pri dešifrovaní by mal softvér kontrolovať, či *hash* dešifrovaného výsledku odpovedá dešifrovanému *hash*-u, ktorý je pripojený do správy;
- pri PGP by mala byť vždy zapnutá kompresia a užívateľ by nemal ignorovať výstražnú správu informujúcu o zlyhaní integrity správy.

Celý dokument o útoku CCA proti e-mailovým bezpečným protokolom (PGP, PEM, MOSS, S/MIME, CMS, atď.) je prístupný na [11] a o útoku CCA len proti PGP a GnuPG je prístupný na [12].

3 SSL/TLS

Protokol SSL (*Secure Sockets Layer*) bol vytvorený firmou Netscape. V praxi sa ujal protokol SSLv3, ale oficiálnym protokolom internetu sa však stal až protokol TLS (*Transport Layer Security*), ktorý vychádza z SSLv3, preto je označovaný aj ako jeho nástupca s názvom SSLv3.1. Protokoly TLS a SSLv3 sú si veľmi blízke, ale nemôžu pracovať súčinne. SSL aj TLS používajú architektúru klient/server.

Vrstva SSL/TLS rieši zabezpečenie prenášaných dát. V rámci OSI modelu sa nachádza medzi aplikačným protokolom a protokolom TCP, takže zaberá funkciu relačnej a prezentačnej vrstvy (obr. 3.1).

SSL/TLS preberá od aplikačnej vrstvy paket po pakete a jednotlivo ich zabezpečuje. U každého paketu vie zaistiť jeho privátnosť, integritu dát a autorizáciu dát. Autorizácia dát sa uskutočňuje na základe kontrolného súčtu počítaného z dát a zdieľaného tajomstva.

Pri nadväzovaní spojenia vrstva SSL/TLS vždy uskutočňuje autentizáciu servera. U klienta je len voliteľná.

Obr. 3.1 Vrstva SSL/TLS v rámci ISO/OSI modelu

Komunikácia protokolom SSL/TLS medzi klientom a serverom je plne duplexná, čo je vlastnosťou protokolu TCP, ale zaujímavosťou je, že pre každý smer komunikácie používa iné symetrické šifrovacie kľúče a iné tajomstvo pre výpočet kontrolného súčtu.

3.1 VLASTNOSTI PROTOKOLU

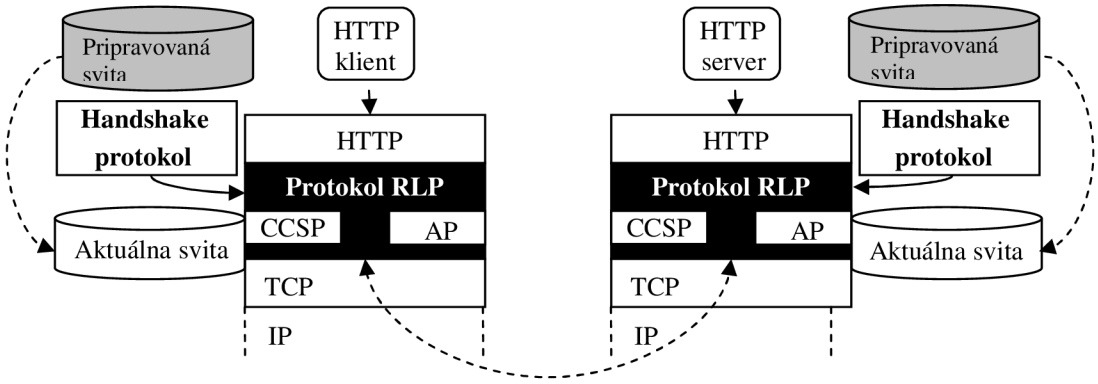
Charakteristické vlastnosti, zhrnuté v základných bodoch, sú nasledovné:

- autentizácia servera sa uskutočňuje na základe certifikátu servera;
- autentizácia klienta sa môže uskutočniť na základe certifikátu klienta, autentizácia sa neuskutočňuje pri komunikácii klienta s anonymným SSL/TLS serverom;
- autentizácia sa uskutočňuje za využitia asymetrickej kryptografie;
- súčasťou úvodného dialógu je výmena dát, z ktorých sa odvodí tzv. zdieľané tajomstvo – je to blok čísel, ktoré poznajú len účastníci komunikácie a odvodzujú sa od neho symetrické šifrovacie kľúče a tzv. tajomstvo pre výpočet kontrolného súčtu;
- SSL/TLS môže šifrovať prenos dát medzi oboma účastníkmi komunikácie, pre šifrovanie sa používa symetrická šifra, ktorej šifrovací kľúč je odvodený od zdieľaného tajomstva;

- jednotlivé prenášané fragmenty dát sa doplňujú o kontrolný súčet zabezpečujúci integritu prenášaných dát, kontrolný súčet sa počíta so zreťazeného fragmentu s tajomstvom pre výpočet kontrolného súčtu, a tak je veľmi ťažké poopraviť prenášané dáta behom prenosu, čím je zabezpečená integrita prenášaných dát;
- SSL/TLS nevidí do aplikačných dát, pretože je to nižšia vrstva a tak nevie rozoznať v aplikačných dátach jednotlivé transakcie;
- je možné ho použiť pre aplikácie, ktoré nevyžadujú elektronické podpisovanie jednotlivých fragmentov či transakcií.

Vrstva SSL/TLS sa skladá zo 4 čiastkových protokolov (obr. 3.2):

1. **Protokol *Record Layer Protocol*** (RLP) – protokol berie dáta od aplikačných protokolov, ktoré následné šifruje a počíta z prenesených fragmentov kontrolný súčet. Druhá strana komunikácie protokolom RLP overuje kontrolný súčet, dešifruje dáta a predáva ich aplikačnému protokolu. Protokol sa nestará o to, aký je použitý typ šifrovacieho algoritmu, stanovenie šifrovacieho kľúča a iné. To má pripravené protokolom HP.
2. ***Handshake protocol*** (HP) – pakety protokola HP sa balia do protokolu RLP. Protokolom HP si strany komunikácie pripravia protokolovú svitu (typ šifrovacieho protokolu a algoritmus pre výpočet kontrolného súčtu), dohodnú si kompresný algoritmus a vymenia dáta pre výpočet hlavného tajomstva, z ktorého si odvodia symetrické kľúče a zdieľané tajomstvo pre výpočet kontrolného súčtu (*MAC secret*). Všetky tieto informácie protokol HP pripraví do tzv. pripravovanej svity (nie aktuálnej).
3. **Protokol *Change Cipher Specification Protocol*** (CCSP) – ak protokol HP pripraví novú protokolovú svitu a všetky potrebné dáta pre protokol RLP, tak je treba skopírovať pripravené parametre spracovania na aktuálne parametre spracovania a začať podľa nich šifrovať. To zaisťuje tento protokol, ktorý oznámi, že nová svita bola skopírovaná a začína sa šifrovať podľa nej.
4. ***Alert Protocol*** (AP) – ak pri komunikácii dôjde k akejkoľvek chybe, tak protokolom AP si to môžu signalizovať strany komunikácie.



Obr. 3.2 Sústava protokolov SSL/TLS

Klient so serverom pre komunikáciu zriaďujú tzv. reláciu (*session*). Relácia môže zahrňovať jedno alebo viac spojení (*connection*). Medzi dvomi počítačmi môže byť súčasne zriadených aj viac relácií.Obe strany komunikácie vo svojich štruktúrach udržujú tieto informácie:

Pre reláciu:

- identifikačné číslo relácie (*session identifier*) – až 32 bajtový identifikátor relácie,

- certifikát druhej strany (*peer certificate*),
- komprimačný algoritmus (*compression metod*) – pre kompresiu dát,
- protokolovú svitu (*cipher spec*) – špecifikuje symetrický šifrovací algoritmus a algoritmus pre výpočet kontrolného súčtu,
- zdieľané tajomstvo (*master secret*) – 48 bajtov známych iba účastníkom komunikácie,
- príznak, či je možné reláciu obnovovať (*is resumable*) alebo je nutné vytvoriť novú;

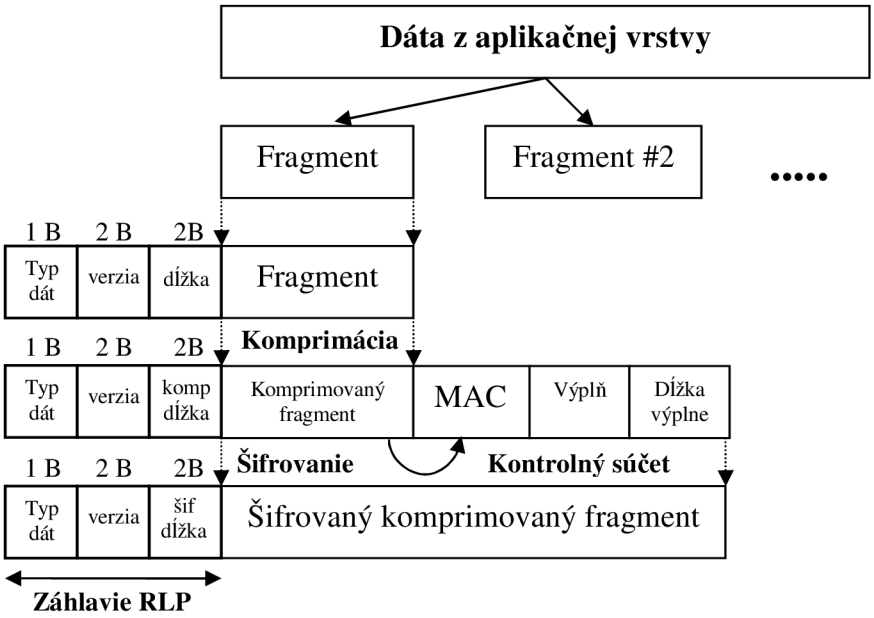
Pre spojenie:

- náhodné čísla generované klientom/serverom (*ClientRandom/ServerRandom*),
- tajomstvo pre výpočet kontrolného súčtu používané klientom/serverom (*client/server write MAC secret*),
- symetrický šifrovací kľúč, ktorým šifruje klient/server (*client/server write key*),
- použité inicializační vektory (IV) pre blokové šifry,
- číslo prijatej a odoslanej správy.

3.2 RECORD LAYER PROTOCOL

Protokol RLP preberá dáta od aplikačných protokolov a postupne uskutoční nasledujúce fázy (obr. 3.3):

- dáta sú rozdelené na fragmenty o dĺžke 2^{14} bajtov a menej;
- komprimácia dát, ak sa na nej predtým dohodli server s klientom v HP protokole;
- vypočítanie kontrolného súčtu algoritmom definovaným v HP protokole, počíta sa zreťazený fragment s tajomstvom pre výpočet kontrolného súčtu (*write MAC secret*) kvôli ochrane pred zmenením obsahu prenášaného fragmentu;
- doplnenie fragmentu o výplň, ak nie je násobkom šifry;
- šifrovanie;
- doplnenie RLP hlavičky.

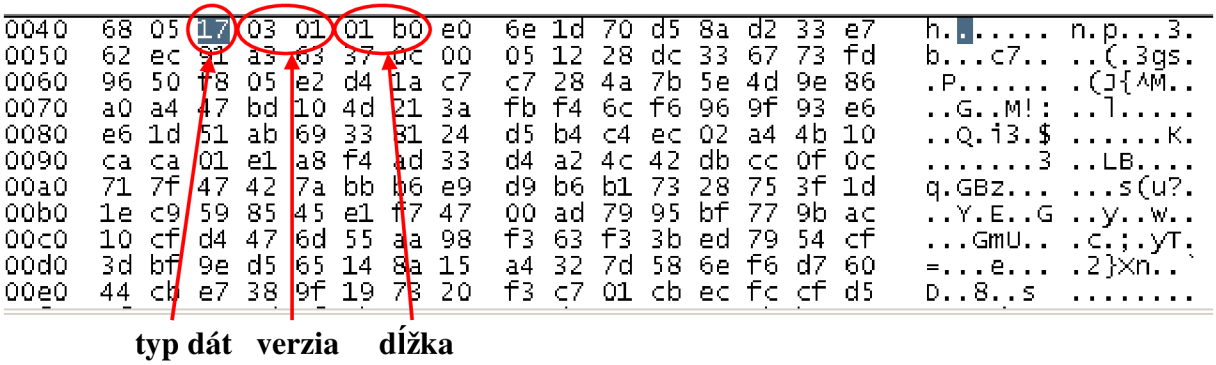


Obr. 3.3 Proces spracovania dát RLP protokolom

Hlavička RLP protokolu sa skladá z troch častí, sú nasledovné:

- *Typ dát* (1 B) - špecifikuje prenášané dáta, napr. dáta aplikačných protokolov (23) alebo služobné dáta SSL/TLS (CCSP=20, AP=21, HP=22);
- *Verzia* (2 B) - označuje verziu SSL/TLS protokolu;
- *Dĺžka* (2 B) - označuje dĺžku fragmentov.

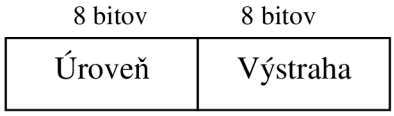
Na obrázku 3.4 je znázornená časť odchyteného paketu, pri RLP protokole sú dáta aplikačného typu (17_{16}=23_{10}), je použitá verzia 3.1 (TLS) a dĺžka fragmentu je 432 ($1B0_{16}$ = 432_{10}).



Obr. 3.4 Paket s protokolom RLP

3.3 ALERT PROTOCOL

AP je jednoduchý protokol, pomocou ktorého sa účastníci komunikácie informujú o chybách v behom SSL spojenia. Patria tu dve úrovne výstrah: fatálna a varovná výstraha. Ak nastane fatálna výstraha, tak spojenie je ihneď ukončené. Správa je znázornená na obr. 3.5.



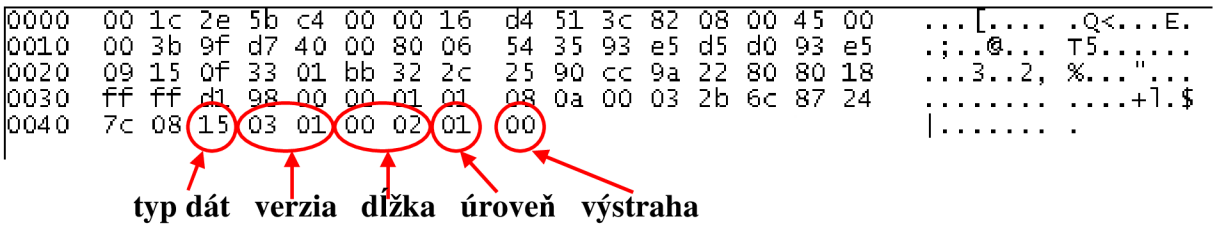
Obr. 3.5 Správa protokolu AP

Pole *Úroveň* indikuje fatálnu alebo varovnú výstrahu. V poli *Výstraha* sa indikuje špecifikácia výstrahy. V tabuľke 3.1 sú uvedené špecifikácie výstrah.

Tab. 3.1: Špecifikácie výstrah

Fatálna výstraha	Varovná výstraha
unexpected_message: bola prijatá správa, ktorá príjemcovi nepatrí	**close_notify:** oznamuje príjemcovi, že odosielateľ ukončil spojenie
bad_record_mac: zlý výpočet MAC	**no_certificate:** klient nemá žiaden vhodný certifikát
decompression_failure: dĺžka po dekompresii prekročila maximum	**bad_certificate:** prijatý certifikát je poškodený
handshake_failure: indikuje chybu pri dohode o zabezpečovacích parametroch	**unsupported_certificate:** nepodporovaný typ certifikátu
ilegal_paramater: nezrovnalosť parametrov vo vnútri „SSL handshake“ protokolu	**certifikate_revoked:** certifikát bol zrušený
	certificate_expired: platnosť certifikátu vypršala
	certificate_unknown: nastala neočakávaná situácia pri spracovaní certifikátu, čo ho robí neprijateľným

Na obrázku 3.6 je zachytený paket, ktorý pri RLP protokole má dáta typu *Alert Protocol* (15_{16}=21_{10}), je použitá verzia 3.1 (TLS), dĺžka fragmentu je 2, úroveň výstrahy je varovná (01) a špecifikácia výstrahy je 00 - *close notify* (ukončenie spojenia).



Obr. 3.6 Paket s protokolom AP

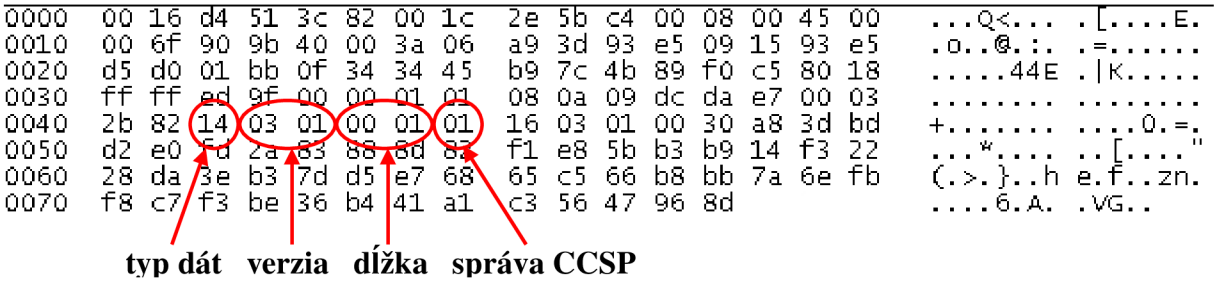
3.4 CHANGE CIPHER SPECIFICATION PROTOCOL

CCSP protokol sa skladá z jedinej správy a slúži na signalizovanie, že došlo k skopírovaniu pripravovaných parametrov pre zabezpečenie spojenia do aktuálnej dátovej štruktúry používanej protokolom RLP. Dáta, ktoré nasledujú za touto správou sú už zabezpečené pomocou novo definovaných parametrov.

CCSP protokol teda slúži na zmenu šifrovacieho kľúča a jeho správa obsahuje len jeden bajt so znakom "1" (obr. 3.7)). Ak sa CCSP nachádza uprostred paketu, tak sa môže stať, že RLP fragment nachádzajúci sa pred ním je šifrovaný inak ako RLP fragment za ním.

Obr. 3.7 Správa protokolu CCSP

Na obrázku 3.8 je zachytený paket, ktorý pri RLP protokole má dáta typu *Change Cipher Specification Protocol* (14_{16}=21_{10}), je použitá verzia 3.1 (TLS), dĺžka fragmentu je 1 a nakoniec nasleduje správa protokolu CCSP.



Obr. 3.8 Paket s protokolom CCSP

3.5 HANDSHAKE PROTOCOL

HP protokol je jadrom SSL/TLS a je zodpovedný za vytvorenie zabezpečenej relácie medzi dvomi stranami. HP protokol môže byť rozdelený na niekoľko dôležitých úrovní:

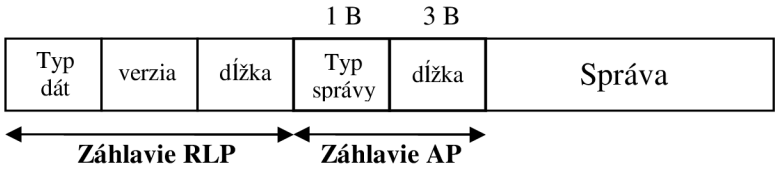
1. autentizáciu servera ku klientovi;

2. dohodu na šifrovacom algoritme alebo šifrovacom kľúči, ktorý podporuje klient aj server;
3. autentizáciu klienta k serveru (ak je vyžadovaná);
4. použitie šifrovania verejným kľúčom na zmenu kryptografických parametrov;
5. vytvorenie zašifrovaného SSL spojenia.

Komunikácia v HP prebieha pomocou správ a rozdeľujeme ju na dva základné prípady:

- nadväzovanie novej relácie,
- obnovenie relácie.

Štruktúra HP paketu (obr. 3.9) je podobná paketu RLP, avšak pri ňom môže byť šifrované aj záhlavie.

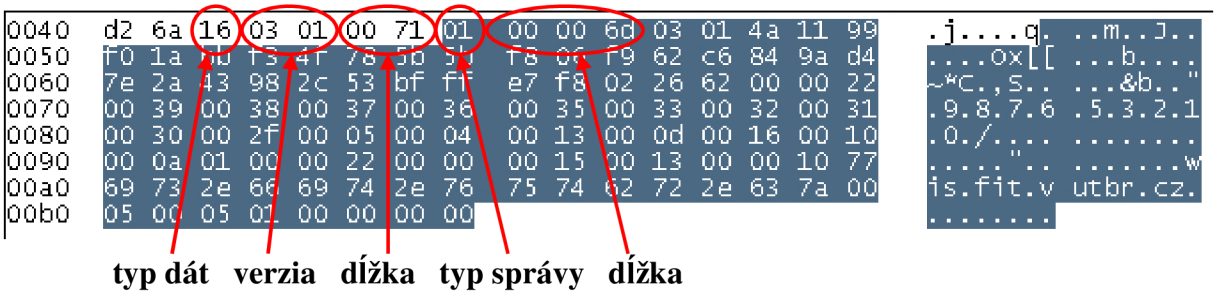


Obr. 3.9 Správa protokolu HP

3.5.1 Správy

ClientHello

Túto správu posiela klient za účelom inicializácie relácie. Na obrázku 3.10 je znázornený paket, kde označená časť tvorí HP protokol. Pred ním sa nachádza záhlavie protokolu RLP (typ správy – *Handshake Protocol*, verzia – 3.1, dĺžka – $0071_{16}=113_{10}$). Za ním nasleduje záhlavie protokolu HP (typ správy 01 – *ClientHello*, dĺžka $00006D_{16}=109_{10}$).



Obr. 3.10 Paket protokolu HP so správou *ClientHello*

Správa zahrňuje nasledujúce parametre:

Version (2 B): nachádza sa za záhlavím a informuje o najvyššej podporovanej verzii SSL/TLS protokolu:

```
03 01
```

Ďalej nasleduje 32 B položka *Random* – náhodné dáta (*ClientRandom*) vygenerované klientom (prvé 4 B obsahujú dátum a čas, aby nebolo možné vygenerovať rovnaké číslo):

```
4A 11 99 F0 1A BB F3 4F 78 5B 5B F8 06 F9 62 C6 84 9A D4 7E 2A
43 98 2C 53 BF FF E7 F8 02 26 62
```

Nasleduje *Session ID* – identifikátor relácie (je nulový - prvé nadväzovanie spojenia):

```
00
```

Potom nasleduje položka *CipherSpec* - protokolové svity identifikujúce šifrovací protokol a protokol pre výpočet kontrolného súčtu, ktoré klient podporuje. Prvé 2 B vyjadrujú dĺžku reťazca predávaných protokolových svit:

```
00 22 00 39 00 38 00 37 00 36 00 35 00 33 00 32 00 31 00 30 00
2F 00 05 00 04 00 13 00 0D 00 16 00 10 00 0A
```

Z toho vyplýva, že daný klient podporuje 17 protokolový svit, ktoré sú uvedené v tabuľke 3.2.

Tab. 3.2: Podporované protokolové svity daného klienta

ID	Protokolová svita	ID	Protokolová svita
0039	TLS_DHE_RSA_WITH_AES_256_CBC_SHA	002F	TLS_RSA_WITH_AES_128_CBC_SHA
0038	TLS_DHE_DSS_WITH_AES_256_CBC_SHA	0005	TLS_RSA_WITH_AES_128_SHA
0037	TLS_DH_RSA_WITH_AES_256_CBC_SHA	0004	TLS_RSA_WITH_AES_128_MD5
0036	TLS_DH_DSS_WITH_AES_256_CBC_SHA	0013	TLS_DHE_DSS_WITH_3DES_EDE_CBC_SHA
0035	TLS_RSA_WITH_AES_256_CBC_SHA	000D	TLS_DH_DSS_WITH_3DES_EDE_CBC_SHA
0033	TLS_DHE_RSA_WITH_AES_128_CBC_SHA	0016	TLS_DHE_RSA_WITH_3DES_EDE_CBC_SHA
0032	TLS_DHE_DSS_WITH_AES_128_CBC_SHA	0010	TLS_DH_RSA_WITH_3DES_EDE_CBC_SHA
0031	TLS_DH_RSA_WITH_AES_128_CBC_SHA	000A	TLS_RSA_WITH_3DES_EDE_CBC_SHA
0030	TLS_DH_DSS_WITH_AES_128_CBC_SHA		

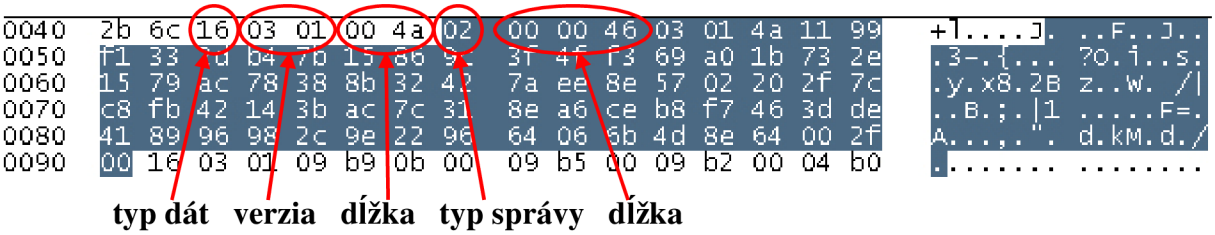
Za tým nasleduje položka *Compression Method* - zoznam kompresných algoritmov, ktoré klient podporuje. Prvý bajt určuje počet podporovaných algoritmov (v tomto prípade jeden), každý nasledujúci bajt potom popise daný algoritmus (00 znamená, že klient nepodporuje žiaden kompresný algoritmus):

```
01 00
```

Za týmito povinnými položkami ešte v danej správe nasleduje rozšírenie (napr. meno servera).

ServerHello

ServerHello je odpoveď servera na správu *ClientHello*. Na obrázku 3.11 vidíme odchytený paket, kde označená časť tvorí HP protokol. Pred ním sa nachádza záhlavie protokolu RLP (typ správy 16 – *Handshake Protocol*, verzia – 3.1, dĺžka – $004A_{16}=74_{10}$). Začiatok označenej časti tvorí záhlavie protokolu HP (typ správy 02 – *ServerHello*, dĺžka $000046_{16}=70_{10}$).



Obr. 3.11 Paket protokolu HP so správou *ServerHello*

Táto správa obsahuje rovnaké parametre ako vyššie uvedené:

Pole *Version* zahrňuje nižšiu verziu navrhovanú klientom a najvyššiu podporovanú serverom:

```
03 01
```

Taktiež generuje náhodné dáta *Random* obsahujúce: dátum a čas (4 B), *ServerRandom* (28 B):

```
4A 11 99 F1 33 2D B4 7B 15 86 9C 3F 4F F3 69 A0 1B 73 2E 15 79
AC 78 38 85 32 42 7A EE 8E 57 02
```

Server potom hľadá zhodu so *Session ID* klienta pre obnovenie spojenia. Ak nájdené nebolo (ako v tomto prípade), server vytvorí nové *Session ID*. Prvý bajt označuje dĺžku (20_{16}=32_{10}), potom nasleduje samotný identifikátor:

```
20 2F 7C C8 FB 42 14 3B AC 7C 31 8E A6 CE B8 F7 46 3D DE 41 89
96 98 2C 9E 22 96 64 06 6B 4D 8E 64
```

Ďalej si server vyberie zo zoznamu *CipherSuite* klienta protokolovú svitu (v tomto prípade TLS_RSA_WITH_AES_128_CBC_SHA) :

```
00 2F
```

Posledný bajt obsahuje serverom zvolený komprimačný algoritmus (v tomto prípade komprimácia nebude použitá):

```
00
```

ServerCertificate

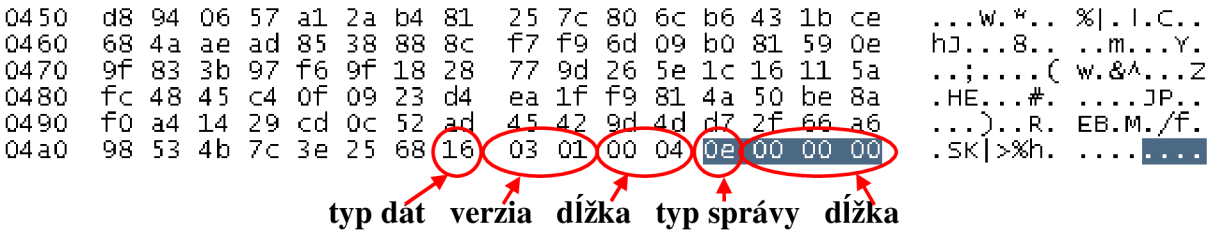
Server posiela svoj certifikát klientovi, čím autentizuje sám seba. Posiela sa hneď za správu *ServerHello*.

CertificateRequest

Touto správou žiada server klienta o jeho certifikát. Súčasťou správy je zoznam podporovaných typov certifikátov a zoznam certifikačných autorít, ktorým server dôveruje.

ServerHelloDone

Táto správa označuje, že server skončil v posielaní svojich šifrovacích a poznávacích parametrov. Po poslaní server čaká na odpoveď klienta. Na obrázku 3.12 vidíme odchytený paket, kde označená časť tvorí HP protokol. Pred ním sa nachádza záhlavie protokolu RLP (typ správy – *Handshake Protocol*, verzia – 3.1, dĺžka – 0004_{16}=4_{10}). Začiatok označenej časti tvorí záhlavie protokolu HP (typ správy 0E – *ServerHelloDone*, dĺžka 000000_{16}=0_{10} – neobsahuje žiadne dáta).



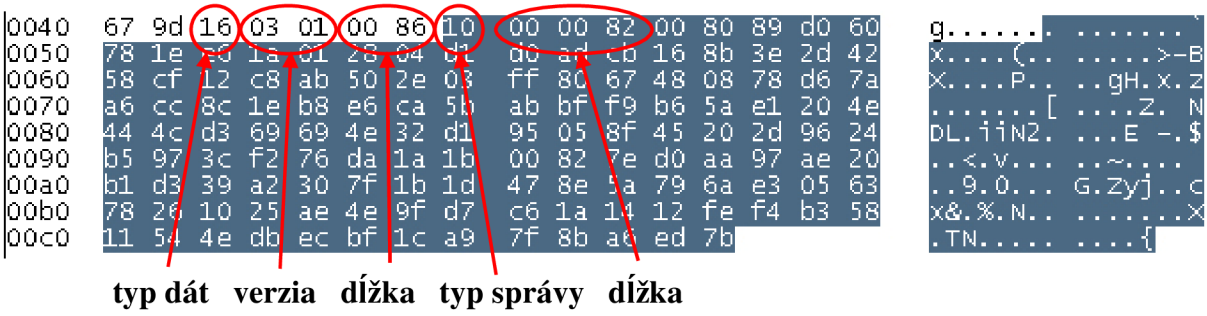
Obr. 3.12 Paket protokolu HP so správou *ServerHelloDone*

ClientCertificate

Správa sa posiela, ak server vyžaduje certifikát od klienta. Ak klient certifikát nemá, tak je poslaná výstražná správa *no_certificate.*

ClientKeyExchange

Slúži pre poslanie predbežného tajomstva (*PreMasterSecret*), z ktorého sa vypočíta hlavné tajomstvo (*MasterSecret*). Na obrázku 3.13 je znázornený paket, kde označená časť tvorí HP protokol. Pred ním sa nachádza záhlavie protokolu RLP (typ správy – *Handshake Protocol*, verzia – 3.1, dĺžka – 0086_{16}=134_{10}). Začiatok označenej časti tvorí záhlavie protokolu HP (typ správy 10 – *ClientKeyExchange*, dĺžka 000082_{16}=130_{10}). Obsah správy je v nasledujúcich bajtoch a obsahuje predbežné tajomstvo (*PreMasterSecret*), ktoré je zašifrované verejným kľúčom servera.



Obr. 3.13 Paket protokolu HP so správou *ClientKeyExchange*

CertificateVerify

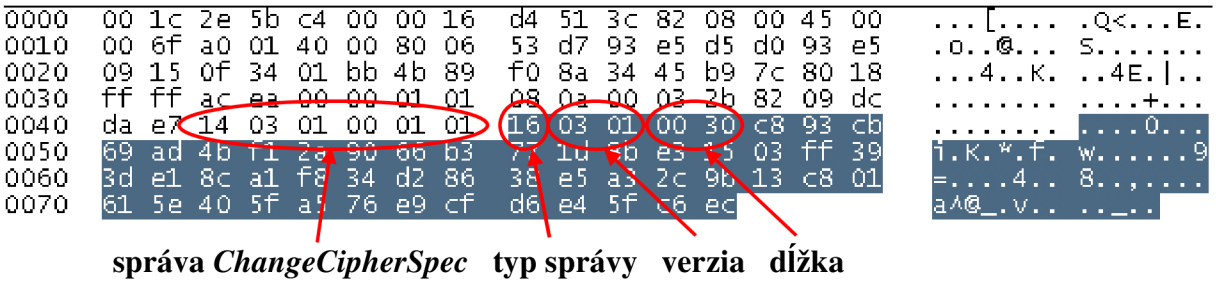
Táto správa sa posiela za účelom definitívneho overenia certifikátu klienta. Správa je poslaná za všetkými klientskymi certifikátmi okrem tých, ktoré zahrňujú parametre algoritmu Diffie-Hellman.

ChangeCipherSpecification

Správa je zhodná s protokolom CCSP, ktorý je podrobnejšie popísaný v kapitole 3.3 bakalárskej práce.

Finished

Táto zašifrovaná správa sa posiela ihneď po správe *ChangeCipherSpecification*. Server aj klient ňou ukončujú dialóg v HP protokole. Na obrázku 3.14 vidíme odchytený paket, kde označená časť tvorí HP protokol. Začiatok tvorí záhlavie protokolu HP (typ správy 16 – *Finished*, verzia 3.1 dĺžka $000030_{16}=48_{10}$). Potom v ďalšej šifrovanej časti je kontrolný súčet zo všetkých predchádzajúcich správ od správy *ClientHello*. Šifrovanie už ale prebieha symetrickým kľúčom.



Obr. 3.14 Paket protokolu HP so správou *Finished*

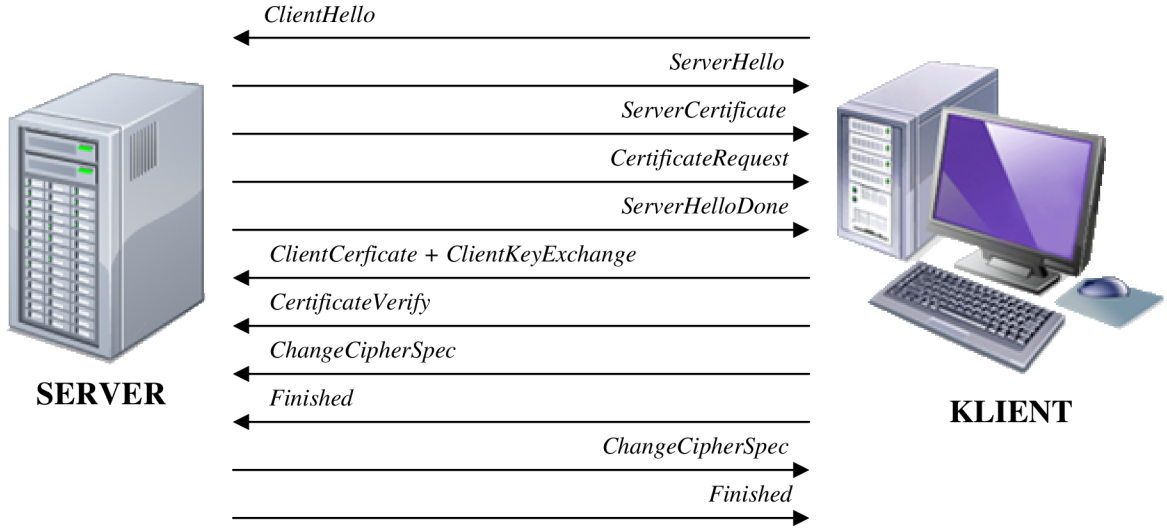
ServerKeyExchange

Používa sa iba v prípade, ak použité algoritmy neumožňujú pomocou správy *ClientKeyExchange* vybudovať blok kľúčov.

HelloRequest

Je prázdna správa, ktorou server upozorňuje klienta , že je na čase, aby klient odoslal správu *ClientHello,* pretože len klient môže začať výmenu správ vedúcu k ustanoveniu nových šifrovacích kľúčov.

Ak prebehnú všetky správy v poriadku, tak je HP protokol kompletný a dáta z aplikačnej vrstvy môžu byť prenesené cez privátny kanál, ktorý bol vytvorený. Celý proces HP protokolu je uvedený na obr. 3.15.



Obr. 3.15 Proces HP protokolu pri vytvorená novej relácie

3.5.2 Obnovenie relácie

Je možnosť klienta a servera znovu začať predchádzajúcu reláciu použitím jej *Session ID*. V tomto prípade nie je potrebný celý proces HP protokolu a sú poslané len správy *ClientHello, ServerHello, ChangeCipherSpecification a Finished.*

3.6 ROZDIELY MEDZI TLS A SSLV3

Pri použití blokovej šifry, by mali byť bloky dát násobkom bloku šifry. Preto dáta, ktoré majú byť zašifrované sú opatrené výplňou. Veľkosť výplne pri TLS môže byť rôzna (až do 255 bajtov), ale nesmie prekročiť maximálnu dĺžku bloku a zároveň musí byť konečný blok násobkom veľkosti bloku šifry. Napríklad, ak máme pred šifrovaním dáta o veľkosti 79 bajtov a blok šifry veľký 8 bajtov, tak veľkosť výplne môže byť 1, 9, 17 ..., až po 249.

V skorších verziách SSL, bola veľkosť výplne minimálna možná tak, aby dáta pred šifrovaním boli násobkom veľkosti bloku šifry. Pri použití vyššie uvedeného príkladu by výplň musela mať 1 bajt. Pri použití premennej veľkosti výplne je zložitejšie použitie útoku, pri ktorom sa analyzuje dĺžka prenesených správ.

Protokol TLS podporuje všetky výstražné správy protokolu SSLv3 okrem správy *no_certificate* a okrem toho zahrňuje aj nové výstražné správy. TLS podporuje taktiež všetky algoritmy výmeny kľúčov a šifrovania protokolu SSLv3 okrem algoritmu Fortezza pri výmene kľúčov a symetrického šifrovacieho algoritmu.

Menší rozdiel medzi protokolmi je aj vo výpočte kontrolného súčtu, ale úroveň bezpečnosti ostáva pre obe rovnaká.

3.7 POUŽITIE

SSL požíva mnoho protokolov z aplikačnej vrstvy, ale najčastejšie sa používa pre HTTP na vytvorenie HTTPS, čím sú zabezpečené webové aplikácie. Využíva sa aj pre protokoly

elektronickej pošty ako POP3, SMTP, IMAP. Tieto aplikácie využívajú k overeniu identity koncových bodov certifikáty s verejnými kľúčmi.

SSL sa môže taktiež použiť pre tunelovanie všetkých sieťových protokolov a vytvorenie virtuálnej privátnej siete (VPN). V porovnaní s tradičnými VPN technológiami na báze IPsec, má SSL zopár výhod pri prechádzaní cez firewall a NAT (*Network Address Translation*), ktoré uľahčujú správu pri väčšom počte vzdialených užívateľov.

3.7.1 Ukážka použitia

Pre praktickú ukážku sme zvolili prihlásenie sa na web-mailový účet na Internete. Najskôr sa prihlásime nezabezpečene a ukážeme, že nie je problém, pri sledovaní môjho *traffic*-u, nájsť prihlasovacie meno a heslo.

Obr. 3.16 Prihlásenie sa na web-mailový účet

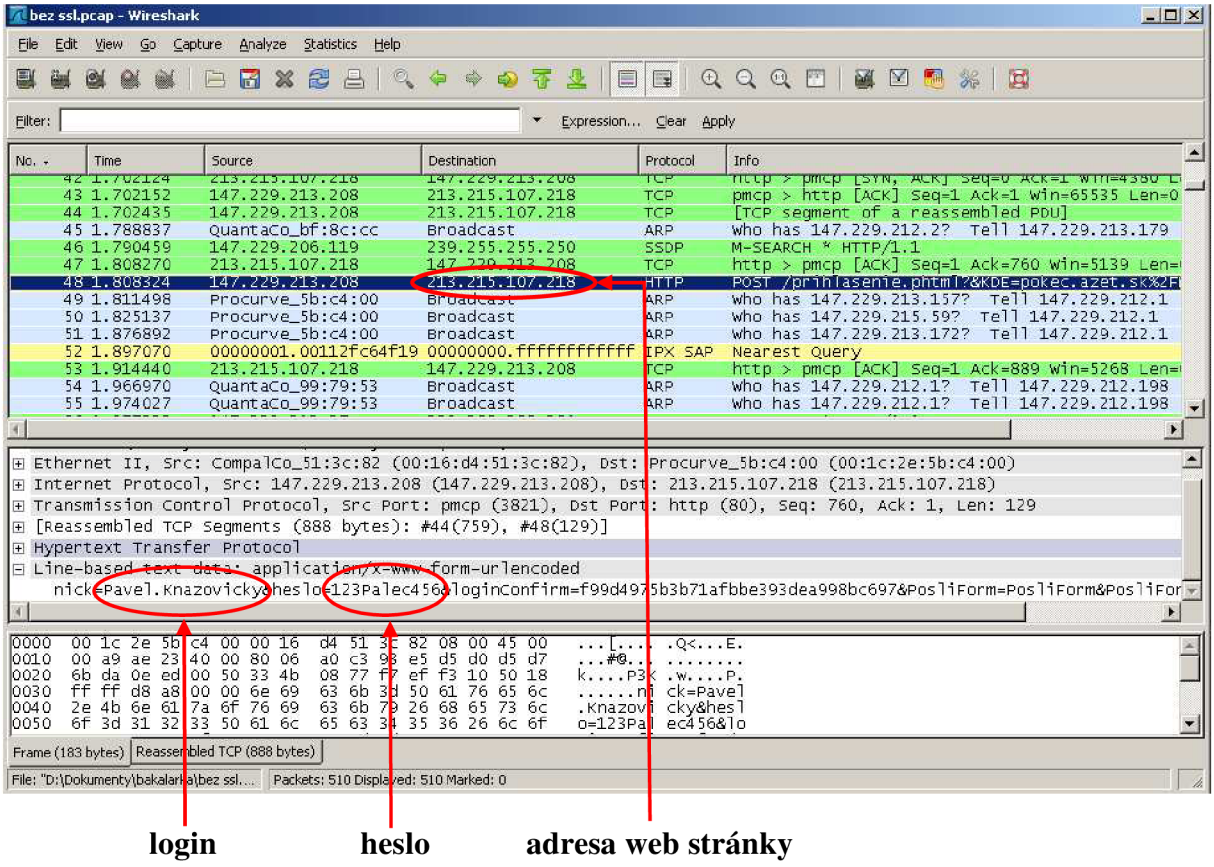
Prihlasovacie údaje sú:

- *login:* Pavel.Knazovicky
- *heslo:* 123Palec456

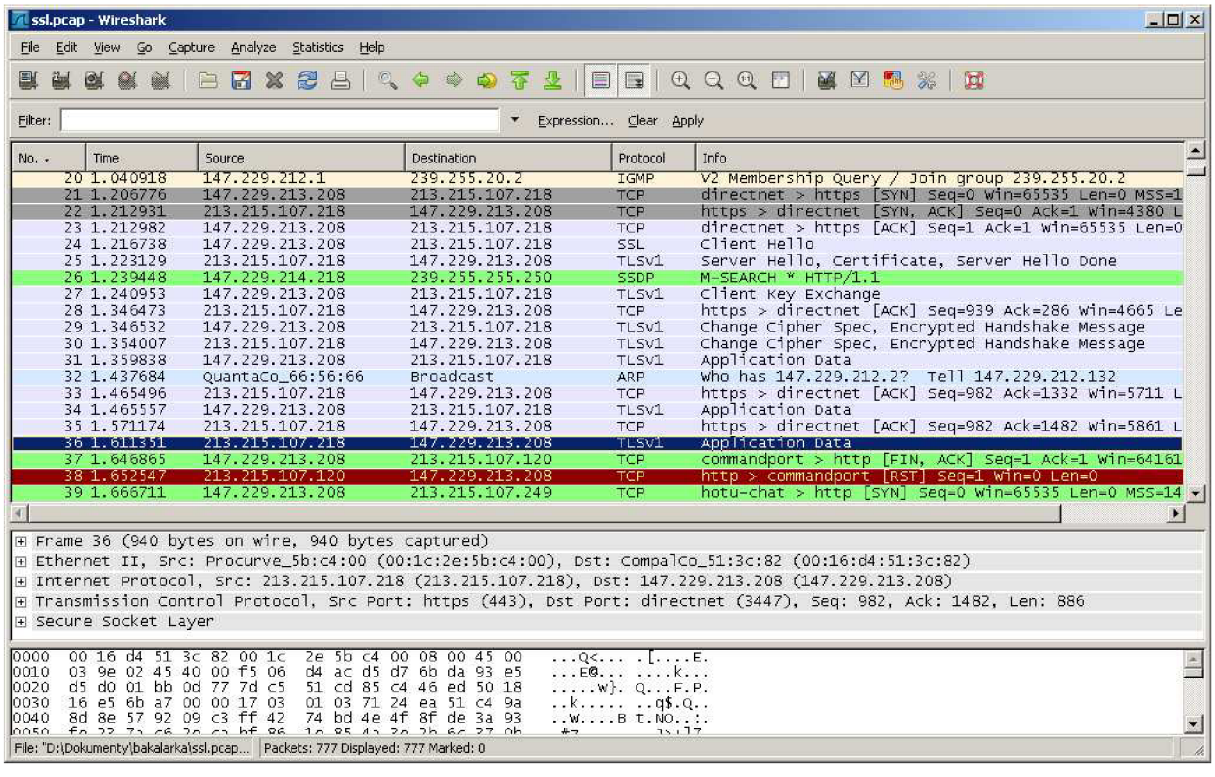
Pri zapnutom programe na sledovanie siete (*Wireshark, tcpdump* a iné) sme bez problémov našli HTTP paket s prihlasovacími údajmi (obr. 3.17).

Ak by bol počítač napadnutý útočníkom, ktorý by sledoval *traffic* nášho počítača z/do Internetu, tak pri krátkom hľadaní by nemal problém nájsť všetky nezašifrované autentizácie, ktoré sme podnikli.

Pri prihlásení pomocou SSL/TLS (obr. 3.18) prebehne HP protokol medzi klientom a serverom, kde si dohodnú všetky potrebné šifrovacie špecifikácie pre dané spojenie (pripravia si svitu na daný prenos). Po vymenení správ *ChangeCipherSpecification* a *finished* medzi klientom a serverom sú, v ďalších paketoch s označením "Aplication Data", prenášané zašifrované citlivé informácie od protokolu HTTP. Je vidieť, že po prihlásení prenos už nie je šifrovaný, takže na danom web-mailovom servery používajú protokol SSL/TLS pre vytvorenie HTTPS len na zabezpečenie prenosu prihlasovacích údajov.



Obr. 3.17 Odchytené pakety pri nezabezpečenom prihlásení



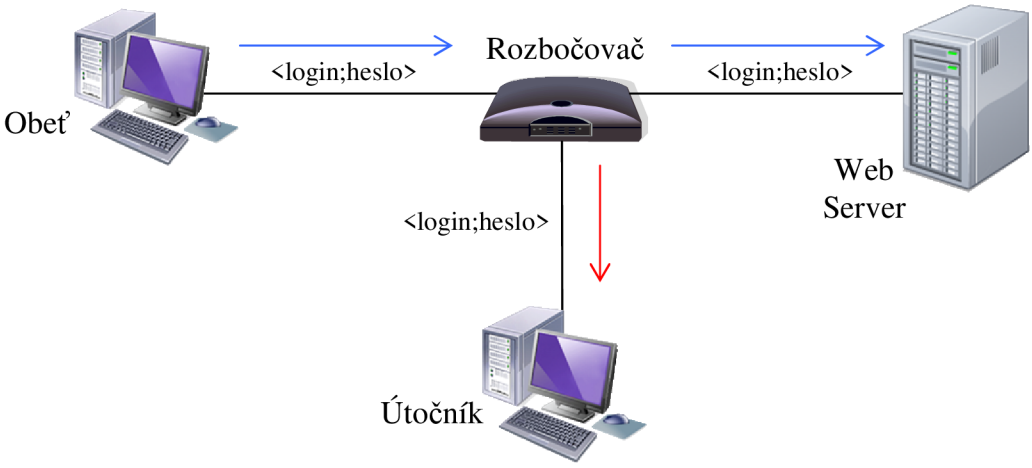
Obr. 3.18 Odchytené pakety pri zabezpečenom prihlásení pomocou SSL/TLS

4 ÚTOKY V POČÍTAČOVÝCH SIEŤACH

Útoky môžeme rozdeliť na **pasívne** a **aktívne**.

K *pasívnym útokom* patrí odpočúvanie komunikácie a analýza komunikácie v sieti. Pri odpočúvaní sa analyzuje obsah zasielaných správ. Ako obranu proti tomu použijeme šifrovanie dát. Analýza komunikácie sa používa aj bez ohľadu na možnosť analýzy obsahu správ. Používa sa pre získanie informácií o komunikujúcich stranách (typ komunikácie, topológia, uzly, atď).

Útok na obr. 4.1 je možný, ak útočník má zapnutý program na sledovanie siete (napr. *Wireshark, tcpdump* a iné) a sieťovú kartu má prepnutú do promiskuitného režimu, ktorá umožňuje príjmať aj dáta, ktoré jej nepatria (rozbočovač posiela dáta na všetky svoje pripojené porty, ale príjme ho len sieťové rozhranie, ktorého adresa je uvedená v hlavičke linkového rámca). Zabrániť tomu môžeme použitím prepínača, kde posielanie nefunguje ako broadcasting, ale dáta sú posielané len na daný port na ktorom má evidované sieťové rozhranie, ktorému dáta náležia. Ale ani táto metóda nie je spoľahlivá.



Obr. 4.1 Útok odpočúvaním na sieti LAN

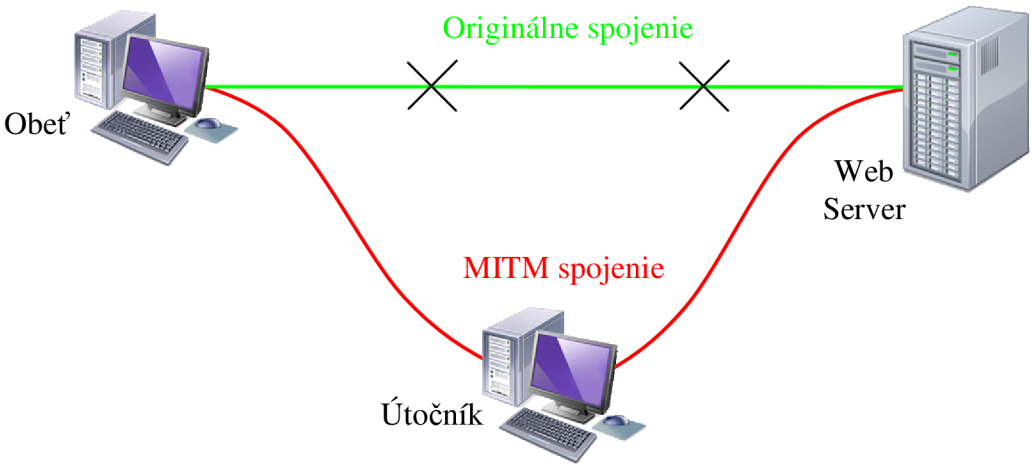
Aktívne útoky sú oveľa prepracovanejšie , ale často sú spojené aj s pasívnym útokom. Útočník aktívne vstupuje do komunikácie. Rozdeľujeme ich na 4 základné kategórie, konkrétne:

- podvrhnutie identity – útočník sa vydáva za legitímneho užívateľa siete;
- útok prehrávaním – najskôr je pasívne odpočúvanie dát, potom opätovné zasielanie dát, čím dochádza k neautorizovanej činnosti;
- modifikácia správy – zmenená správa, zmena poriadia správ, zámerne pozdržaná správa;
- DoS (*Denial of Service*) – zahltenie, vyradenie z činnosti, napr. aktívne zasielanie veľkého množsva správ útočníkom.

4.1 PRAKTICKÉ ÚTOKY

4.1.1 MITM

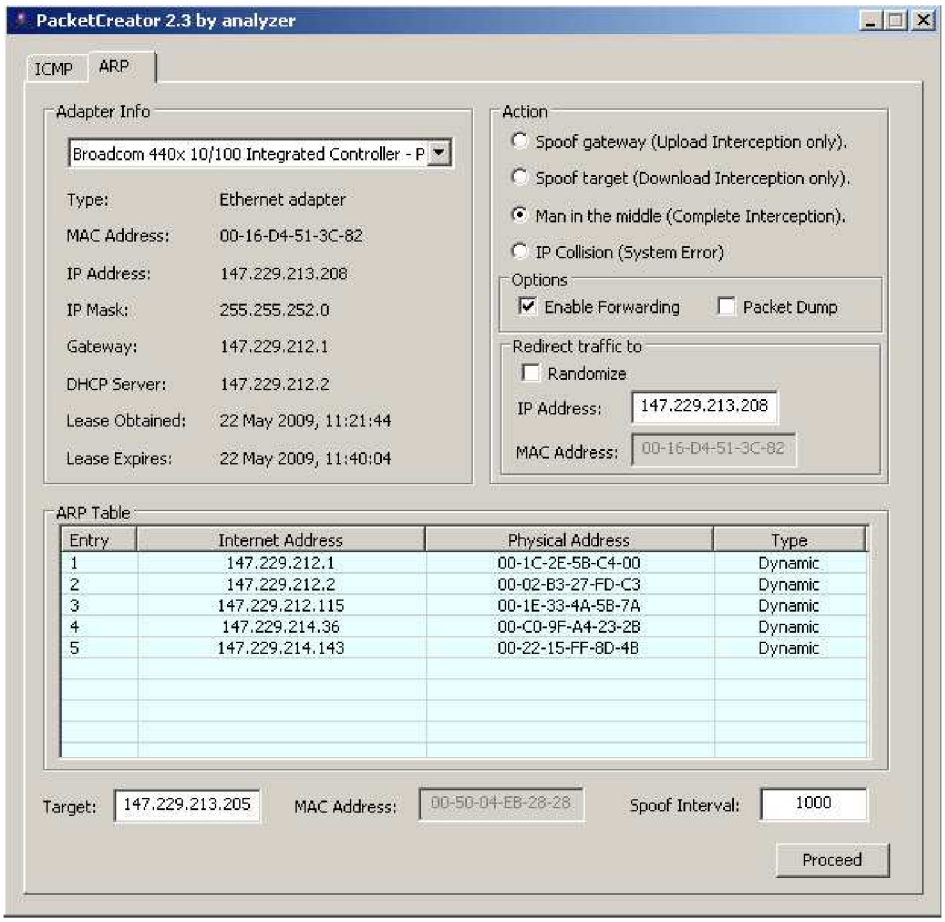
Ako z názvu (*Man-in-the-Middle*) vyplýva, jedná sa o útok, kde originálna trasa komunikácie medzi zariadeniami je presmerovaná cez útočníka (obr. 4.2). Tým môžeme docieliť napr. odpočúvanie komunikácie, útok prehrávaním, DoS, *phishing* a iné.



Obr. 4.2 Princíp útoku MITM

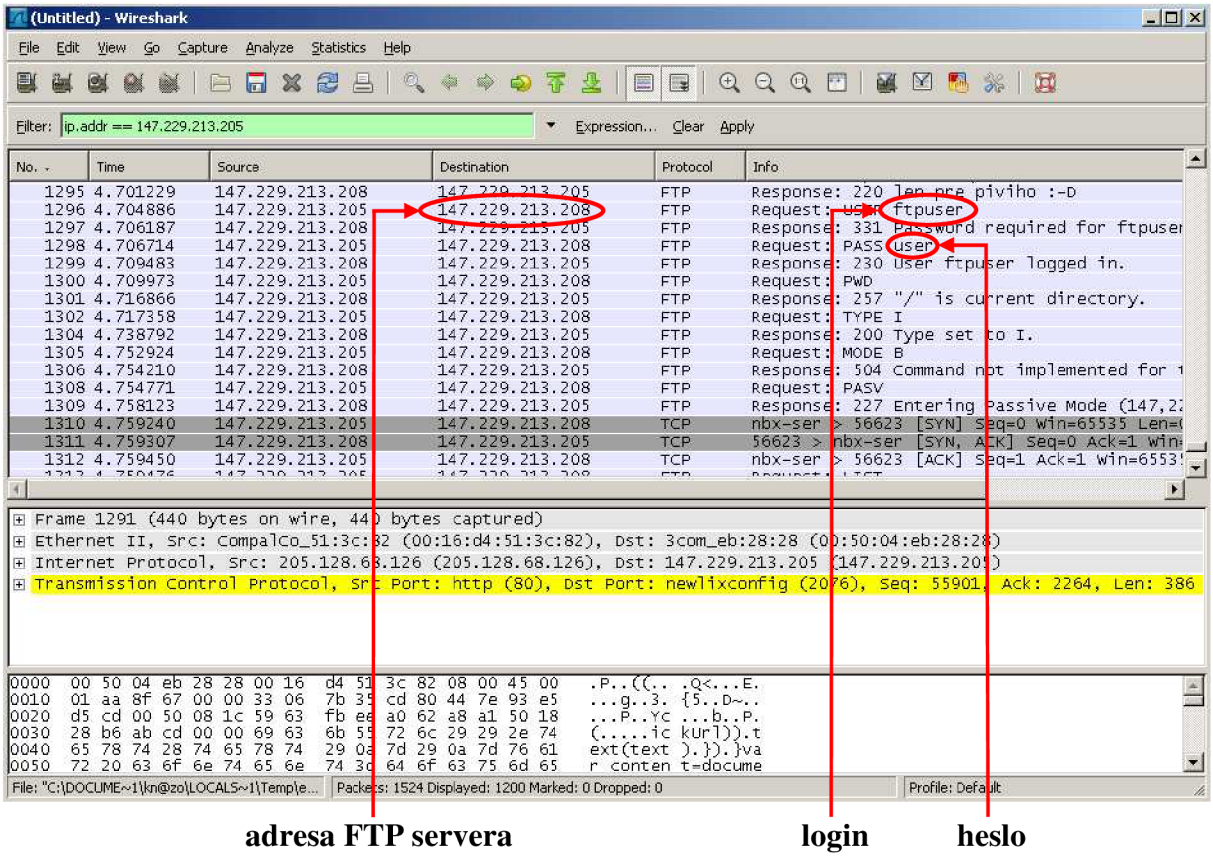
Ako obranu proti MITM útoku môžeme použiť PKI, silnú vzájomnú autentizáciu, biometriu a iné. Nástroj pre MITM útok je napr. *ettercap* (pre Linux) alebo *PacketCreator* (pre Windows).

Pre ukážku sme zvolili freeware program *PacketCreator*, ktorý bol vytvorený pre sieťových administrátorov na testovanie ich siete. Je veľmi jednoducho nastaviteľný (obr. 4.3). Pre útok MITM sa prepneme do záložky ARP, kde označíme *Action – Man in the middle*, v *Options* označíme *Enable Forwarding* (aby pakety boli preposielané pôvodnému adresátovi), do položky *Target* napíšeme IP adresu obete a nakoniec klikneme na tlačidlo *Proceed*.



Obr. 4.3 Nastavenie programu PacketCreator pre útok MITM

Ďalej si spustíme program na sledovanie siete (napr. *Wireshark*), napíšeme filter pre zobrazovanie paketov z IP adresy obete a môžeme sledovať, aké pakety sú odoslané/prijaté z daného počítača. Môžeme odchytávať napr. ICQ komunikáciu (ktorá prebieha nešifrovane), prihlasovacie mená a heslá pri nezabezpečenom prihlásení atď. Na obr. 4.4 je odchytené prihlásenie na FTP server.



Obr. 4.4 Odchytené pakety pri útoku MITM

4.1.2 Phishing

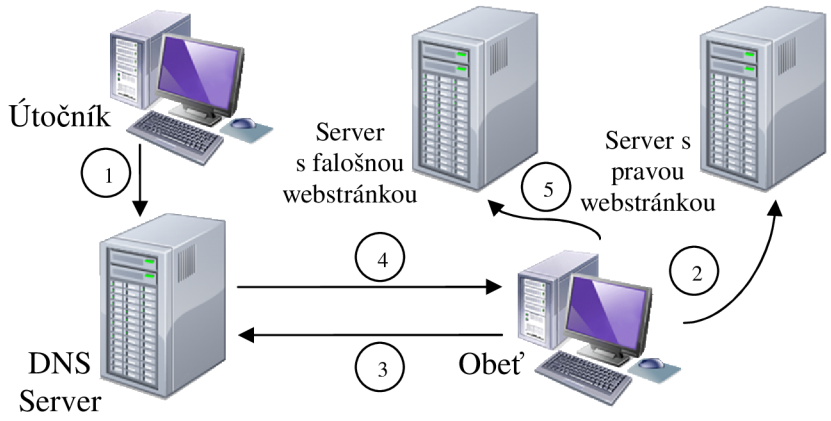
Jedná sa o podvrhnutie vzdialeného servera útočníkom. Väčšinou sa jedná o adresu webového servera a pripájajúci užívateľ si myslí, že pristupuje na pôvodný dôveryhodný server.

Často sa začína rozoslaním e-mailových správ užívateľom s výzvou o znovu pripojenie do systému. V skutočnosti sa však pripoja na útočníkov server, čím môžu predať útočníkovi svoje citlivé informácie (login, heslo, PIN, číslo kreditnej karty a iné).

Ochranou proti phishingu je dodržovanie bezpečnostných pravidiel a použitie aplikácií k detekciiín týchto útokov.

4.1.3 Pharming

Je novší a sofistikovanejší ako predchádzajúci phising. Využíva protokol DNS, ktorý slúži k preklade mien na IP a naopak, čiže sa jedná o útok na DNS serveri. Nová IP adresa je obvykle k nerozoznaniu od pôvodnej. Princíp útoku je na nasledujúcom obrázku.

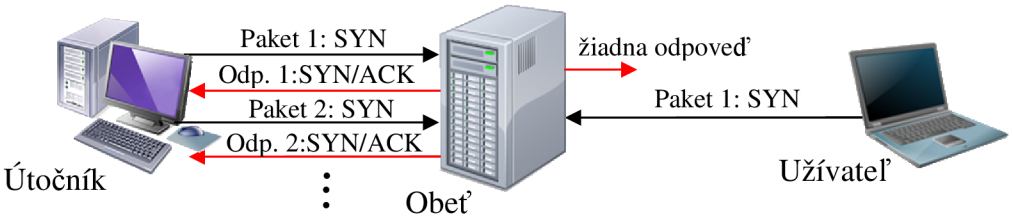


Obr. 4.5 Princíp útoku *pharming*

4.1.4 DoS

Pre útok Dos (*Denial of Service*) je typické zahltenie sieťového pásma, alebo odpojenie sieťových prostriedkov. Pre útok využíva toho, že vzdialené systémy musia uchovávať informácie o stavoch komunikácie a komunikujúce strany čakajú od konkrétnych aplikácií špecifické dáta. Dôsledkom útoku je zabránenie prístupu k prostriedkom alebo službám.

Jedným z mnohých DoS útokov je napr. vytvorenie veľa polovičných spojení (obr. 4.6). Útočník posiela SYN pakety oznamujúce naviazanie spojenia, počítač potvrdí prijatie paketu (SYN-ACK) a očakáva potvrdenie ACK paketom, ktorý však nepríde. Behom niekoľkých sekúnd dôjde k tisícom nových spojení, ktoré server nedokáže obslúžiť, čím prestane odpovedať na požiadavky legitímnych užívateľov alebo sa úplne zrúti.



Obr. 4.6 Princíp útoku DoS so SYN paketmi

Nástroj pre DoS útok je napr. *Tribe Flood Network* (pre Linux) alebo *Stacheldraht* (pre Windows).

4.1.5 Útoky v Ethernete

ARP spoofing

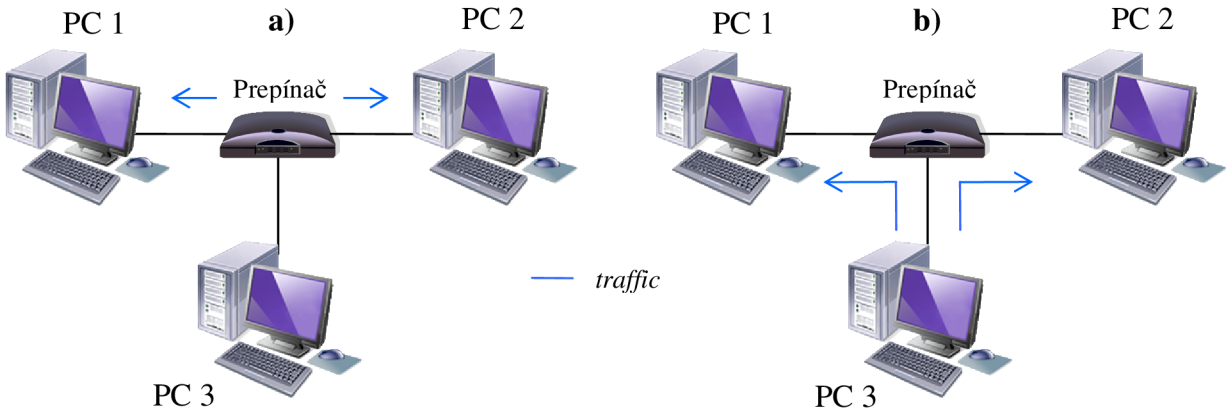
Pri tejto metóde sa útočník vydáva v miestnej sieti za iný počítač za pomoci zneužitia protokolu ARP (*Adress Resolution Protocol*), ktorý je používaný s protokolom IPv4 v rámci lokálnej siete k prekladu IP adries na fyzické adresy sieťových rozhraní. Princíp tohto útoku je v neustálom zasielaní podvrhnutej fyzickej adresy, ktorá nepatrí žiadnej IP adrese. Napadnutý počítač si zaznamená falošnú adresu do svojich ARP tabuliek a dáta bude následne posielať na ňu. Ak chceme odpočúvať komunikáciu medzi dvomi uzlami v rámci lokálnej siete, tak stačí obom účastníkom podstrčiť svoju fyzickú adresu. Dáta, ktoré by sme od nich dostali, stačí už len ďalej posielať adresátovi, ktorému boli určené. *ARP spoofing* sa môže aplikovať v lokálnych sieťach založených na technológii ethernet, kde sú počítače prepojené pomocou prepínača (*switch*).

Ako ochranu proti podvrhnutiu fyzickej adresy je možné použiť IPv6, tunelovanie spojenia, alebo statickú ARP tabuľku (veľmi neobvyklé).

Pre simuláciu tohto útoku sme si vytvorili testovaciu lokálnu sieť s tromi počítačmi, ktoré sú pripojené pomocou prepínača. Princíp útoku je graficky znázornený na obr. 4.7. V tabuľke 4.1 sú uvedené logické a fyzické adresy počítačov v testovacej sieti.

Tab. 4.1: Popis počítačov v testovacej sieti

	IP adresa/prefix	**Fyzická adresa**
PC 1	192.168.0.1/24	00:1F:1F:14:D3:D5
PC 2	192.168.0.2/24	00:50:04:EB:28:28
PC 3	192.168.0.3/24	00:16:D4:51:3C:82



Obr. 4.7 Komunikácia medzi PC 1 a PC 2
a) pri normálnej prevádzke b) pri použití útoku *ARP spoofing*

Z obrázku 4.7 je zrejmé, že úmyslom je odchytávať komunikáciu medzi PC 1 a PC 2 pomocou PC 3 (na počítačoch je použitý operačný systém Linux). Aby sa mohli poslať prijaté dáta k adresátovi, ktorému patria, tak musíme mať v PC 3 zakomponovanú podporu "IP: Advanced router". Ďalej potrebujeme mať na PC 3 inštalovaný balík *dsniff*.

Dsniff zahrňuje niekoľko programov pre testovanie sieťového prieniku a kontroly účtov. Utility ako *dsniff*, *filesnarf*, *mailsnarf*, *msgsnarf* a *webspy* pasívne kontrolujú sieť pre zachytenie citlivých dát ako napr. heslá, e-maily atď. *Arpspoof*, *dnsspoof*, *macof* pomáhajú zachytiť sieťový *traffic*, ktorý nie je normálne dostupný útočníkovi. Balík je podľa slov autora (Dug Song) vyvinutý s poctivými zámermi na preverenie jeho vlastnej siete a pre ukážku nezabezpečených sieťových aplikačných protokolov.

Na začiatku si zapneme v PC 3 podporu routra v jadre príkazom:
```
echo "1" > /proc/sys/net/ipv4/ip_forward
```

Vo iptables firewall-e si nastavíme základné nastavenie pre prístup:
```
iptables UP INPUT ACCEPT
iptables UP OUTPUT ACCEPT
iptables UP FORWARD ACCEPT
```

Sieťová komunikácia v rámci lokálnej siete je zabezpečená pomocou fyzických adries sieťových rozhraní, ktoré sú zaznamenané v ich ARP tabuľkách. Tieto tabuľky sa najčastejšie menia dynamicky na základe *traffic*-u alebo pomocou ARP paketov. Počítač, ktorý chce zistiť fyzickú adresu pomocou IP adresy, tak vyšle paket *ARP request* s danou IP adresou od ktorej dostáva paket *ARP reply* s požadovanou MAC adresou. Na posielanie falošných *ARP reply* paketov na PC 1 a PC 2 použijeme utilitu *arpspoof* z balíka *dsniff*.

Nasledujúcim príkazom povieme PC 1, že PC 2 má fyzickú adresu 00:16:D4:51:3C:82 a nie 00:50:04:EB:28:28:

```
arpspoof -t 192.168.0.1 192.168.0.2
```

A naopak PC 2 povieme, že PC 1 má fyzickú adresu 00:16:D4:51:3C:82 a nie 00:1F:1F:14:D3:D5:

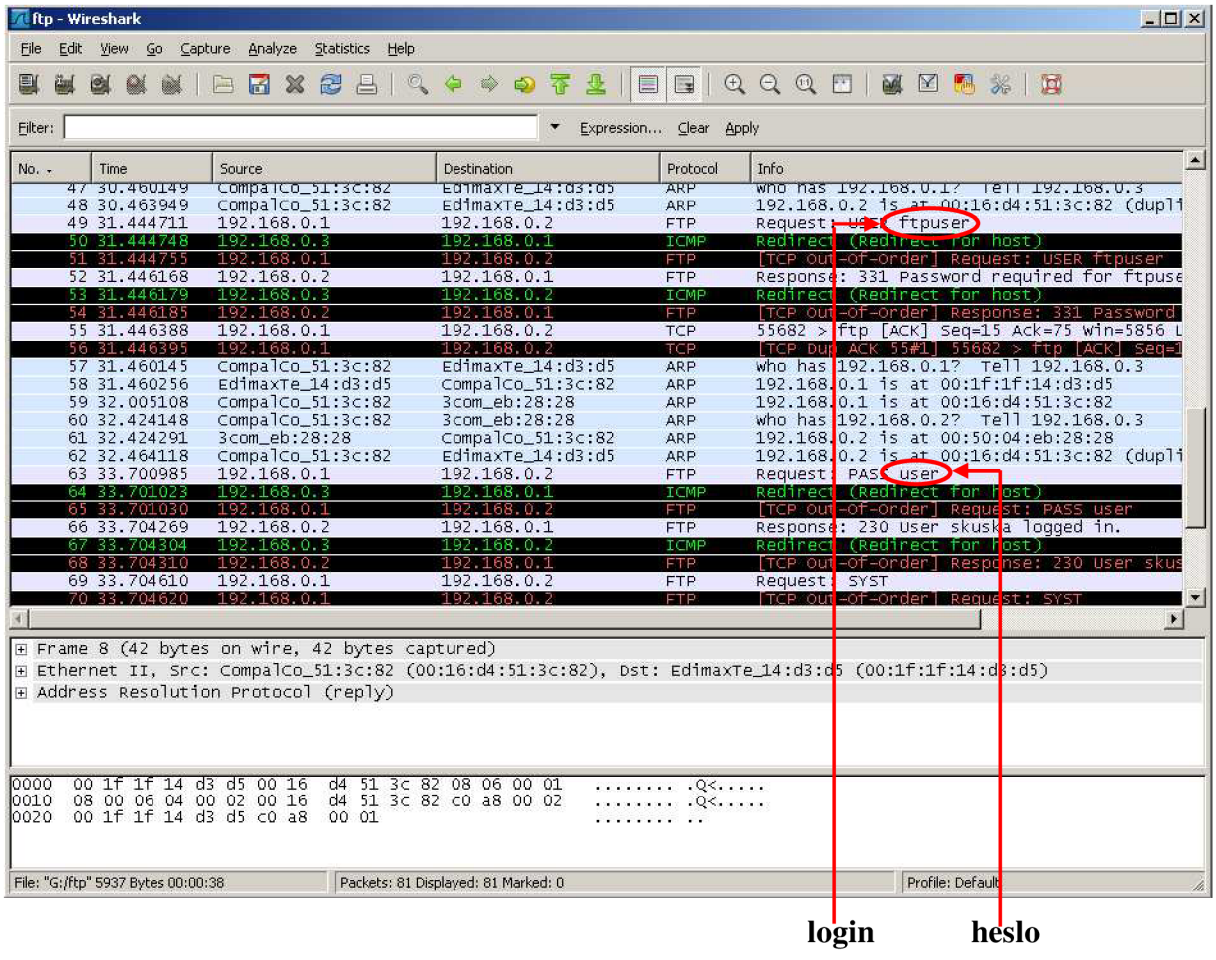
```
arpspoof -t 192.168.0.2 192.168.0.1
```

Takto je celý *traffic* medzi PC 1 a PC 2 presmerovaný cez PC 3 (môžeme ho sledovať cez rôzne programy na sledovanie siete ako *Wireshark, tcpdump* a iné), ktorý sa správa ako smerovač a pakety, ktoré mu nepatria, posiela podľa svojej ARP tabuľky správnemu adresátovi.

Na PC 2 je vytvorený FTP server s nasledujúcim prihlasovacím kontom:

- *login:* ftpuser
- *heslo:* user

Keďže protokol FTP je nezabezpečený, tak dáta smerované medzi klientom a serverom sú nešifrované. Ak sa prihlásime z PC 1 na tento FTP server (obr. 4.8), tak v odchytených paketoch na PC 3, nie je problém rozpoznať autentizáciu na tento server.



Obr. 4.8: Odchytené pakety pri prihlásení sa na FTP server

Z vyššie uvedených ochrán, proti tomuto útoku, sme si vybrali tunelovanie spojenia pomocou protokolu SSH. Protokol FTP pracuje na dvoch portoch 20 (FTP dáta) a 21 (FTP príkazy). Myslím, že dôležité je zašifrovanie autentizácie, tak nám bude postačovať port 21. Na PC 2

musíme mať zapnutý SSH-server. Na vytvorenie šifrovaného kanálu zadáme na PC 1 nasledujúci príkaz:

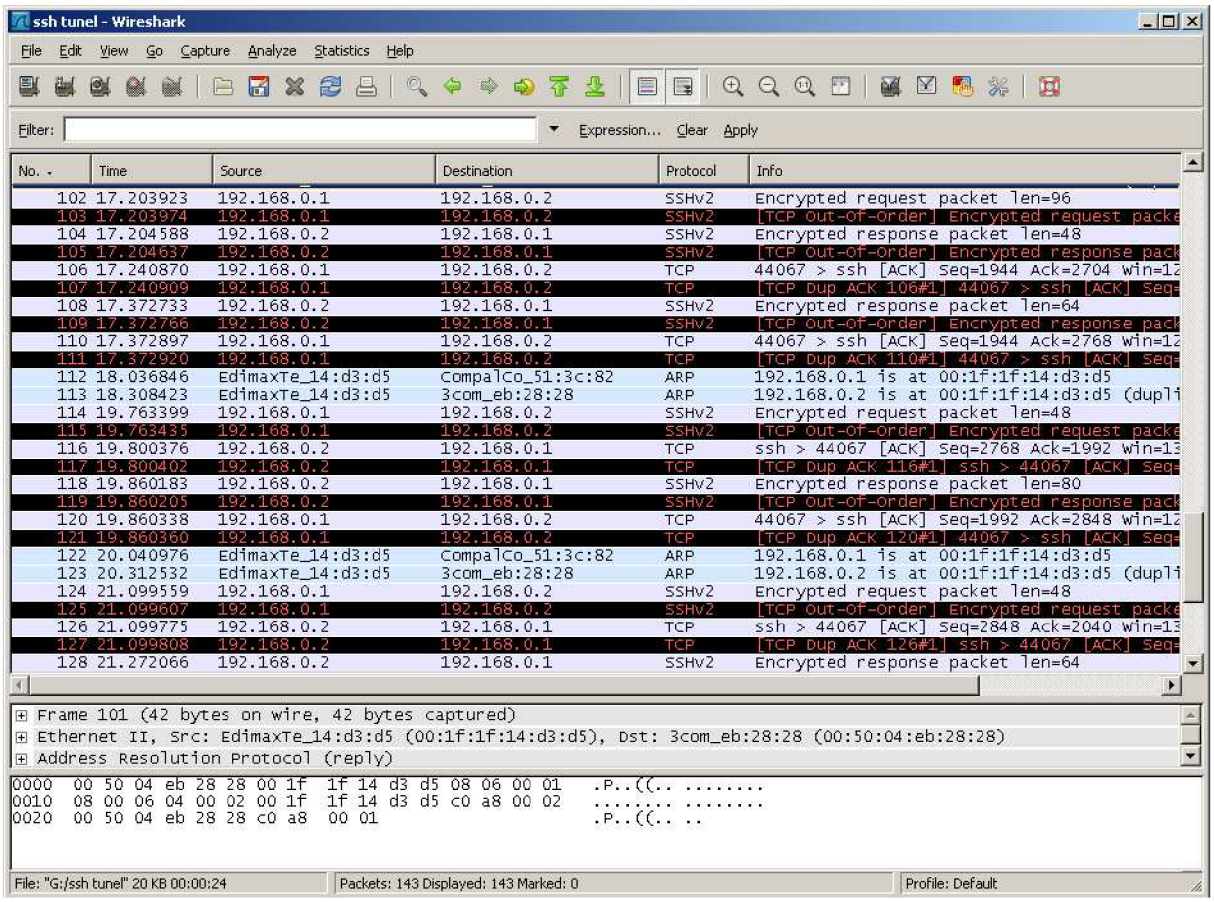
```
ssh -L 8021:localhost:21 user@192.168.0.2
```

Tým vytvoríme šifrovaný kanál medzi PC 1 a PC 2 a následne sa môžeme prihlásiť na FTP server takto:

```
ftp localhost 8021
```

Na obrázku 4.9 vidíme, že celá autentizácia je šifrovaná pomocou protokolu SSH, čím sú odchytené pakety pre útočníka bezcenné. Samozrejme existuje priamo protokol SFTP, ktorý na šifrovanie a autentizáciu používa protokol SSH, ale šifrovanie podstatne zaťažuje procesor a spomaľuje prenosy a tak vo veľa prípadoch je zašifrovanie autentizácie postačujúce.

Pomocou SSH môžeme vytvoriť šifrovaný kanál aj pre rôzne iné aplikačné protokoly pracujúce nad protokolom TCP.

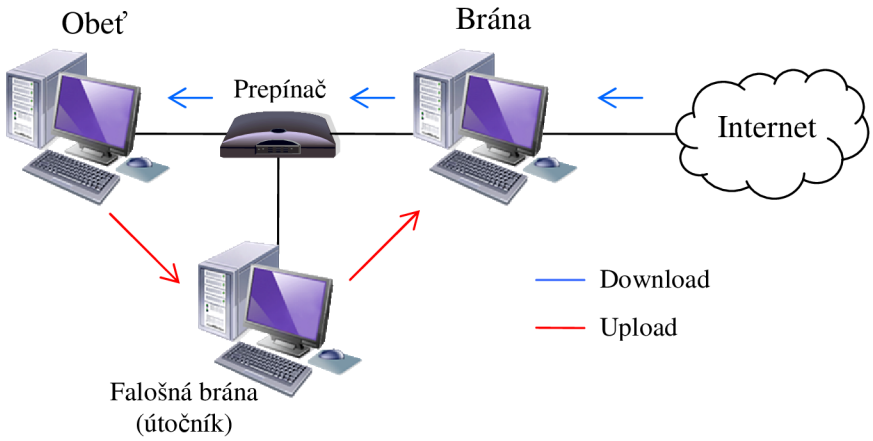


Obr. 4.9: Odchytené pakety pri prihlásení na FTP server cez SSH tunel

DHCP spoofing

V sieti LAN môže byť aj viacero DHCP serverov, čo práve využíva tento útok. Počítač obete, ktorý sa pripája do siete odošle paket *DHCP discover*. Dostáva správu *DHCP offer* od najrýchlejšieho servera, napr. aj útočníkovho. Útočník tak môže určiť DHCP parametre pre daný počítač, presmerovať komunikáciu na falošnú bránu, kedy ale musí preposielať pakety na pôvodnú bránu, aby predišiel prezradeniu.

Princíp útoku je znázornený na obrázku 4.10.



Obr. 4.10: Princíp útoku DHCP spoofing

MAC flooding

V prepínačoch sa fyzické adresy ukladajú do CAM tabuľky. Princíp útoku spočíva v zasielaní rámcov s náhodnými fyzickými adresami, až kým sa nenaplní CAM tabuľka a prepínač sa prepne do stavu *fail open* a chová sa ako rozbočovač.

Novšie zariadenia už sú väčšinou ochránené pred týmto útokom.

Port stealing

Ako s názvu vyplýva jedná sa o kradnutie portu na prepínači. Útočník si najskôr zistí fyzickú adresu adresáta a následne posiela upravené pakety, kde cieľová fyzická adresa je zhodná s adresou útočníka a cieľová fyzická adresa je zhodná s adresou adresáta. Prepínač následne priradí adresátovi nový port a komunikácia smerom k adresátovi je doručená útočníkovi, ktorý ju musí následne preposlať adresátovi, aby nebol odhalený.

Ako ochranu proti útoku je pravidelne poopravovať CAM tabuľky v prepínači (napr. poslaním paketu *ARP reply* na počítač adresáta).

DNS spoofing

Princíp je podobný útoku *ARP spoofing*. Útočník využíva slabiny DNS systému. Protokol DNS slúži na preklad mien na IP adresy a naopak. Komunikácia s DNS serverom je pomocou UDP protokolu, a tak je ľahko napadnuteľná. Útok spočíva v spustení programu, ktorý sleduje DNS požiadavky na sieti. Ak si nejaká stanica vyžiada službu DNS, program automaticky odošle odpoveď s falošnou IP adresou, ktorá smeruje na útočníkov počítač. Ak príde paket od útočníka skôr ako zo skutočného DNS, napadnutý počítač ho bude akceptovať a záznam z DNS bude ignorovať. Ďalšia komunikácia by mala byť ďalej smerovaná na skutočný cieľ, aby sa útočník vyhol odhaleniu.

Prevencia proti útoku spočíva v zavedení *DNS Security*, ktoré umožňuje každú odpoveď servera podpísať.

4.2 PREVENCIA

Prevencia pred útokmi je určite výhodnejšia ako následné odstraňovanie škôd. Medzi základné bezpečnostné prevencie zaraďujeme:

- **firewall** – pri správnej konfigurácii je veľmi účinný proti útokom a skenovaní vnútorných sietí;
- **informovanosť** – poznať všetky nové bezpečnostné hrozby (šíriace sa útoky na Internete sú často vopred hlásené, takže sa im môžeme vyhnúť);
- **systémový audit** – možnosť zapojenia automatických nástrojov, ktoré kontrolujú nezvyčajné udalosti alebo podozrivé aktivity;
- **antivírový softvér** – mať kvalitný antivírový program a pravidelne ho aktualizovať;
- **heslá** – používať netriviálne heslá, pravidelná zmena hesiel, jednorazové heslá;
- **zálohovanie dát** – pravidelné zálohovanie dát;
- **aktualizácie** – pravidelná inštalácia aktualizácií a záplat pri používaných softvéroch.

ZÁVER

Cieľom bakalárskej práce bola analýza protokolov zaisťujúcich zabezpečený prenos dát na jednotlivých vrstvách modelu ISO/OSI a zabezpečenie prenosu použitím týchto protokolov.

Prvá kapitola je zameraná na jednoduchý popis funkcií jednotlivých vrstiev referenčného modelu ISO/OSI. Ďalšia časť sa zaoberá už samotnými zabezpečovacími protokolmi, ktoré sú rozdelené podľa pôsobnosti v rámci modelu ISO/OSI. Na podrobný rozbor som si vybral protokol SSL/TLS, pretože je najčastejšie používaný pri šifrovanej komunikácii v prostredí Internetu. Posledná časť je venovaná útokom v počítačovej sieti, z ktorých sme dva prakticky vyskúšali pomocou testovacích utilít k tomu vyvinutých.

Behom práce sme zistili, že niektoré protokoly majú slabšiu ochranu a niektoré sú zasa odolnejšie voči aktívnym či pasívnym útokom. Je to dané tým, že u starších zabezpečovacích protokolov neboli dobre aplikované kryptografické ochrany, čím vznikli mnohé slabiny, ktoré mohol potenciálny útočník využiť. Pri vyvíjaní novších protokolov sa odborníci snažia všetky tieto slabiny odstrániť, napr. tým, že využívajú nové šifrovacie a autentizačné metódy, ktoré zabezpečenie zdokonaľujú. Tým vznikajú protokoly nové, alebo len novšie verzie protokolov starších.

Na počítačových sieťach striehne veľa nebezpečenstiev, práve preto by sme otázku bezpečnosti nemali brať na ľahkú váhu. Najväčším opatrením je počítač vôbec nepoužívať, ale to v dnešnej modernej dobe asi nie je možné. Určite by sme mali používať šifrovaný prenos informácií a mať paranoju na všetko neobvyklé. Návody na rôzne útoky sa dajú bez problémov nájsť na Internete, čím počet potenciálnych útočníkov určite rastie. Pre väčšinu bežných protokolov môžeme nájsť ich zabezpečenú alternatívu, poprípade ich môžeme zabezpečiť niektorým z nich. Ktoré oblasti by sme, teda, s použitím protokolov spomenutých v práci, mali zabezpečiť? Zapamätajme si päť hlavných oblastí, začínajúcich písmenom P, konkrétne:

- Pre prácu na vzdialenom počítači používať protokol SSH poprípade iné tunelovacie protokoly.
- Pre prenos súborov používať SFTP alebo SCP.
- Pre prácu s elektronickou poštou používať S/MIME alebo PGP.
- Pri webových serveroch využívať zabezpečenie pomocou SSL/TLS minimálne pri autentizácii.
- Pri zabezpečení bezdrôtových sietí používať protokoly WPA, WPA2 a nie protokol WEP, ktorý je v dnešnej dobe už ľahko prelomiteľný.

Súčasná kryptografia sa musí stále vyvíjať, pretože sa vyvíja aj výpočtová technika a útočníci sú čoraz vynaliezavejší. Zo zvýšeným rozvojom počítačových sietí narastá aj zvýšený prenos citlivých informácií. Žiaden zo zabezpečovacích protokolov nemôže poskytnúť stopercentnú ochranu, a tak v dnešnej dobe oblasť kryptografie prechádza búrlivým rozvojom.

Výsledkom bakalárskej práce je zhrnutie v dnešnej dobe využívaných zabezpečovacích protokolov, útokov v počítačovej sieti, základná prevencia proti nim a praktická ukážka dvoch z nich.

POUŽITÁ LITERATÚRA

[1] DOSTÁLEK, Libor, KABELOVÁ, Alena. *Velký průvodce protokoly TCP/IP a systémem DNS*, 2. akt. vyd., Computer Press, Praha, 2000, 425 s. ISBN: 80-7226-323-4

[2] THOMAS, Thomas M. *Zabezpečení počítačových sítí bez předchozích znalostí*, CP Books, a.s., Brno, 2005, 338 s. ISBN: 80-251-0417-6

[3] PETERKA, Jiří. *E-archiv Jiřího Peterky: Referenčný model ISO/OSI – jeho vznik* [online]. 1992, 1.12.1992 [cit. 2008-11-12]. Dostupné na Internete: http://www.earchiv.cz/a92/a212c110.php3

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

[5] DOSTÁLEK, Libor.: *Velký průvodce protokoly TCP/IP: Bezpečnost.* 2. aktualizované vydání, Computer Press, Praha, 2003, 571 s. ISBN: 80-7226-849-X

[6] MILOŠ, Jiří. *Kryptografické metody zabezpečení dat*. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, 2008. 43 s. Vedoucí bakalářské práce Ing. Petra Lambertová.

[7] BURDA, Karel: *Bezpečnosť informačných systémov*, VUT, Brno, 2005

[8] ZEMAN, Jaroslav, TANUŠKA, Pavol. *Niektoré problémy bezpečnosti počítačových sietí založených na technológii WiFi* [online], STU, Trnava, 2006 [cit. 2008-12-02]. Dostupné na Internete: http://www.mtf.stuba.sk/docs//internetovy_casopis/2006/2/tanuska.pdf

[9] SCHNEIER, Bruce. *Cryptoanalysis Microsoft's Point-to-Point Tunneling Protocol (PPTP)* [online]. 1998. [cit. 2009-04-30]. Dostupné na Internete: http://www.schneier.com/paper-pptp.pdf

[10] SCHNEIER, Bruce, FERGUSON, Niels. *A Cryptographic Evaluation of IPsec* [online]. 2003 [cit. 2009-05-02]. Dostupné na Internete: http://www.schneier.com/paper-ipsec.pdf

[11] SCHNEIER, Bruce, KATZ, Jonathan. *A Chosen Ciphertext Attack Against Several E-Mail Encryption Protocols* [online]. 2000, 23.06.2000 [cit. 2009-05-10]. Dostupné na Internete: http://www.schneier.com/paper-chotext.pdf

[12] SCHNEIER, Bruce, KATZ, Jonathan, JALLAD, Kahil. *Implememtation of Chosen-Ciphertext Attacks against PGP a GnuPG* [online]. 2002 [cit. 2009-05-12]. Dostupné na Internete: http://www.schneier.com/paper-pgp.pdf

[13] -: *SSL Tutorial* [online].2001, 31.08.2001 [cit. 2008-12-10]. Dostupné na Internete: http://www2.rad.com/networks/2001/ssl/index.htm

[14] -: *dsniff* [online]. Dostupné na Internete: http://monkey.org/~dugsong/dsniff/

ZOZNAM SKRATIEK

AES	Advanced Encryption Standard
AH	Authentication Header
AP	Alert Protocol
ARP	Address Resolution Protocol
CCITT	Comité Consultatif International Téléphonique et Télégraphique
CCSP	Change Cipher Specification Protocol
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name System
DoS	Denial of Service
EAP	Extensible Authentication Protocol
FTP	File Transfer Protocol
GRE	Generic Routing Encapsulation
HP	Handshake Protocol
HTTPS	Hyper Text Transport Protocol Secure
CHAP	Challenge-Handshake Authentication Protocol
IETF	Internet Engineering Task Force
IKE	Internet Key Exchange
IPsec	Internet Protocol security
ISAKMP	Internet Security Association and Key Management Protocol
ISO	International Organization for Standardization
L2TP	Layer 2 Tunneling Protocol
LAN	Local Area Network
MAC	Message Authentication Code
MD5	Message Digest 5
MITM	Man-in-the-Middle
OSI	Open Systems Interconnection
PAP	Password Authentication Protocol
PGP	Pretty Good Privacy
PKI	Public Key Infrastructure
PPP	Point-to-Point Protocol
PPTP	Point-to-Point Tunneling Protocol
QoS	Quality of Service
RADIUS	Remote Authentication Dial In User Service
RES	Rivest, Shamir, Adleman
RLP	Record Layer Protocol
S/MIME	Secure / Multipurpose Internet Mail Extensions
SA	Security Agreements
SCP	Secure Copy
SFTP	SSH File Transfer Protocol
SSH	Secure Shell
SSL	Secure Socket Layer
TCP	Transmission Control Protocol
TKIP	Temporal Key Integrity Protocol
TLS	Transport Layer Security
VPN	Virtual Private Network
WAN	Wide Area Network
WEP	Wired Equivalent Privacy
WPA	Wi-Fi Protected Access

ZOZNAM OBRÁZKOV

ZOZNAM TABULIEK

ZOZNAM PRÍLOH

Na priloženom CD nosiči sa nachádza elektronický text bakalárskej práce vo formáte pdf.