Česká zemědělská univerzita v Praze

Technická fakulta

Diplomová práce

Analýza možností komunikace prostřednictvím VPN protokolů s návazností na Subnetting

Bc. Karel Truneček

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Technická fakulta

ZADÁNÍ DIPLOMOVÉ PRÁCE

Bc. Karel Truneček

Informační a řídicí technika v agropotravinářském komplexu

Název práce

Analýza možností komunikace prostřednictvím VPN protokolů s návazností na subnetting

Název anglicky

Analysis of possibilities for communication via VPN protocols with connection to subnetting

---

Cíle práce

Cílem práce bude posoudit vhodnost a technické možnosti propojení koncových sítí globální sítě prostřednictvím různých VPN a porovnat reálné řešení. Diskutovány budou jednotlivé aplikační vrstvy a jejich limity, posouzena celková vhodnost použití. Zjištěné předpoklady budou ověřeny vlastními testy. Metodika testování bude podrobně zdůvodněna. Součástí práce bude i doporučení pro reálný provoz především z pohledu ceny a bezpečnosti.

Metodika

1. Úvod
2. Cíl práce
3. Metodika
4. Komunikační protokoly na globální síti
5. Routování
6. Virtual Private Network
7. Integrace služeb
8. Metodika testování
9. Praktické testy
10. Výsledky a zhodnocení
11. Závěr a doporučení

Doporučený rozsah práce

50 – 60 stánek včetně obrázků a grafů

Klíčová slova

počítačová síť, bezpečnost, VPN, subnetting

Doporučené zdroje informací

Barry L Williams: Information Security Policy Development for Compliance, ISBN: 1466580585, Taylor & Francis Ltd, 2013

Bartlett, Graham;Inamdar, Amjad: IKEv2 IPsec Virtual Private Networks, Pearson Education (US), 2016, ISBN: 9781587144608 / 9781587144608

COMER, D E. Computer networks and Internets : with Internet applications. Upper Saddle River: Prentice Hall, 2009. ISBN 0-13-091449-5.

James F. Kurose: Počítačové sítě, Computer Press, 2014, EAN: 9788025138250

Qiang Huang Jazib Frahim: SSL Remote Access VPNs, Cisco Press, 2008, ISBN 1587052423

Předběžný termín obhajoby

2020/2021 LS – TF

Vedoucí práce

Ing. Zdeněk Votruba, Ph.D.

Garantující pracoviště

Katedra technologických zařízení staveb

Elektronicky schváleno dne 3. 3. 2020

doc. Ing. Jan Malaťák, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 10. 2. 2021

doc. Ing. Jiří Mašek, Ph.D.

Děkan

V Praze dne 01. 03. 2021

Čestné prohlášení

Prohlašuji, že svou diplomovou práci *Analýza možností komunikace prostřednictvím VPN protokolů s návazností na Subnetting* jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne: …….. ……

Bc. Karel Truneček

Poděkování

Děkuji Ing. Zdeňku Votrubovi, Ph.D., vedoucímu bakalářské práce, za odborné vedení a cenné rady, čímž přispěl k vypracování této diplomové práce.

Analýza možností komunikace point-to-point prostřednictvím VPN IPsec s návazností na subnetting

Abstrakt: Cílem diplomové práce je zrealizovat propojení přes globální síť pomocí služby VPN s využitím protokolů PPTP, IPsec a L2TP včetně využití subnettingu na cílových uzlech. V práci je zdokumentována celá implementace a posouzení celkové vhodnosti použití v reálném provozu z pohledu konfigurace, rychlosti přenosu, ceny a bezpečnosti. V teoretické části je čtenář nejprve seznámen se základními principy komunikace po lokální a veřejné síti, včetně technologie VPN – PPTP, IPsec a L2TP a metody subnetting. V praktické části je následně rozebrána konfigurace všech zapojení včetně jednotlivých měření a testů, které budou zaměřeny především na ověření správnosti konfigurace, rychlosti přenosu a analýzy přenášených paketů.

Klíčová slova: VPN, PPTP, IPsec, L2TP, počítačová síť, subnetting, bezpečnost

Analysis of possibilities of point-to-point communication via VPN IPsec with connection to subnetting

Abstract: The aim of this master's thesis is to realize a connection over a global network using a VPN service with PPTP, IPsec and L2TP protocols, including the use of subnetting on target nodes. The work documents the whole implementation and assessment of the overall suitability of use in real operation in terms of configuration, transmission speed, price and security. In the theoretical part, the reader is first acquainted with the basic principles of communication over the local and public network, including VPN technology – PPTP, IPsec and L2TP and the subnetting method. The practical part then discusses the configuration of all connections, including individual measurements and tests, which will be focused primarily on verifying the correctness of the configuration, transmission speed and analysis of transmitted packets.

Keywords: VPN, PPTP, IPsec, L2TP, computer network, subnetting, security

Obsah

Seznam obrázků

Seznam tabulek

Seznam grafů

Seznam příloh

Seznam použitých zkratek

AES – Advanced Encryption Standard

ARPA – Advanced Research Projects Agency

ASCII – American Standard Code for Information Interchange

BGP – Border Gateway Protocol

CIDR – Classless Inter-Domain Routing

CRC – Cyclic redundancy check

DES – Data Encryption Standard

DHCP – Dynamic Host Configuration Protocol

DNS – Domain Name Systém

DPD – Dead Peer Detect

DSA – Digital Signature Algorithm

EAP – Extensible Authentication Protocol

FQDN – Fully Qualified Domain Name

FTP – File Transfer Protocol

HTTP – Hypertext Transfer Protocol

IETF – Internet Engineering Task Force

IMAP – Internet Message Access Protocol

IP – Internet Protocol

IPX/SPX – Internetwork Packet Exchange (IPX) / Sequenced Packet Exchange (SPX)

ISAKMP – The Internet Security Association and Key Management Protocol

ISO/OSI – International Organization for Standardization / Open System Interconnection

IS-IS – Intermediate System to Intermediate Systém

L2F – Layer 2 Forwarding

L2TP – Layer 2 Tunneling Protocol

LAN – Local Area Network

LCP – Link Control Protocol

MPPE – Microsoft Point-to-Point Encryption

MS-CHAPv2 – Microsoft-Challenge-Handshake Authentication Protocol

NAT – Network Address Translation

NAPT – Network Address Port Translation

NCP – Network Control Protocols

OSPF – Open Shortest Path First

POP3 – Post Office Protocol

PPTP – Point-to-Point Tunneling Protocol

QoS – Quality of Service

RSA – Rivest, Shamir, Adleman

RIP – Routing Information Protocol

SA – Security Association

SMTP – Simple Mail Transfer Protocol

SNMP – Simple Network Management Protocol

SSH – Secure Shell

SSL – Secure Sockets Layer

TCP/IP – Transmission Control Protocol / Internet Protocol

TLS – Transport Layer Security

UDP – User Datagram Protocol

VPN – Virtual Private Network

WAN – Wide Area Network

1. Úvod

Internet přinesl neuvěřitelné množství výhod a zjednodušení, které si ještě v minulém století neuměl nikdo ani představit. Bohužel je stejně užitečný jako je nebezpečný. Práce s ním vždy nese určité riziko, protože ne všichni uživatelé se zde chovají podle pravidel a mají dobré úmysly. Jak tedy lze chránit přenášená data přes tuto nepředstavitelně ohromnou a nezabezpečenou síť? Jedním z nezbytných kroků je bez debaty šifrování. Šifrovat komunikaci tak, aby zpráva měla smysl pouze pro oprávněnou osobu se lidé snaží už od pradávna. Samozřejmě dříve se šifrovaly dopisy a dnes se šifrují pakety. Ovšem princip zůstává pořád stejný. Pro zvýšení bezpečnosti přenosu se využívá právě služba VPN neboli virtuální privátní síť. VPN ve své podstatě vytváří zabezpečený tunel napříč nezabezpečenou sítí, přes který probíhá šifrovaný, a tedy zabezpečený přenos dat.

Bez pochyby jsou data v dnešní době pro mnohé společnosti právě tím nejcennějším, co mají. Právě proto je musí i náležitě zabezpečit. Zejména v poslední době, kdy se celý svět potýká s pandemií COVID-19 se řada společností přesvědčila o tom, jak je důležité pečlivě zabezpečit přenos dat. Zejména pro zaměstnance, kteří musejí pracovat z domova. Existuje celá řada druhů VPN a jejich správná volba a konfigurace je klíčová pro zabezpečení přenosu. V této práci jsou rozebrány a porovnány tři známé a široce používané VPN protokoly – PPTP, IPsec a L2TP.

S postupem času jsou útočníci čím dál tím více vynalézaví. Ruku v ruce s tím ovšem přichází celá řada vylepšení jak přenos a samotná data ještě lépe zabezpečit. Příkladem této technologie může být právě i subnetting. Jeho správnou konfigurací lze podstatně zvýšit bezpečnost i výkonnost sítě a usnadnit její spravování.

2. Cíl práce

Diplomová práce *Analýza možností komunikace prostřednictvím VPN protokolů s návazností na Subnetting* je zaměřena na realizaci propojení lokální sítě a vzdálené počítačové stanice skrze globální síť pomocí služby Virtual Private Network (VPN). Vzdálená počítačová stanice má navíc pomocí metody subnettigu omezený přístup pouze k vybraným zařízením. V této diplomové práci budou porovnávány a testovány tři široce známé a používané VPN protokoly – Point-to-Point Tunneling Protocol (PPTP), Internet Protocol Security (IPsec) a Layer 2 Tunneling Protocol (L2TP). Vybrané VPN protokoly budou podrobeny několika testům zaměřeným především na přenosovou rychlost, ztrátovost paketů, latenci, jitter a analýzu paketového přenosu. Závěrem práce budou tyto VPN protokoly navzájem porovnány z pohledu složitosti a flexibility konfigurace, bezpečnosti, naměřených výsledků a ceny.

3. Metodika

K realizaci VPN serveru bude využit router ER6020v1 od společnosti TP-Link. Na něm bude postupně nakonfigurována služba VPN s využitím všech tří VPN protokolů (PPTP, IPsec, L2TP). K lokální síti routeru ER6020v1 bude připojována vzdálená počítačová stanice, která se implicitně nachází v jiné síti. Tato vzdálená počítačová stanice bude vždy pomocí metody subnetting zařazena do konkrétní podsítě tak, aby měla přístup pouze k vybraným zařízením v lokální síti routeru ER6020v1.

Po ověření úspěšné realizace VPN bude skrz vzniklý VPN tunel testována přenosová rychlost (download a upload), ztrátovost UDP paketů, latence (ping) a jitter. Ke změření těchto hodnot bude využit freewarový program TamoSoft Throughput Test, příkazový řádek a webový portál speedtest.cesnet.cz. Na rychlosti přenosu se samozřejmě kromě samotné cesty přes globální síť od počítačové stanice k VPN serveru projeví i zdržení, kdy musí konkrétní VPN protokol každý datový paket před odesláním zašifrovat a na druhé straně tunelu zase dešifrovat. U každého testovaného VPN protokolu bude navíc pomocí programu Wiresahrk doložena analýza paketového přenosu při připojení a odpojení VPN.

4. Teoretická část

V teoretické části je nejprve popsán referenční model ISO/OSI, přenosové protokoly TCP a UDP, funkce IP adresy včetně masky a brány sítě a následně je vysvětlen princip samotného směrování (routování), jenž pomyslně završuje první půlku teoretické části, která má za úkol seznámit čtenáře se základními principy komunikace po lokální a veřejné síti. V druhé půlce teoretické části jsou rozebrány hlavní témata této práce, které pracují na výše pospaných principech – služba VPN, protokoly PPTP, IPsec, L2TP a metoda subnetting.

4.1. Referenční model ISO/OSI

ISO/OSI je zkratka, kde ISO představuje standardizační organizaci, která tuto normu vydala, tedy International Organization for Standardization a OSI z anglického Open System Interconnection, což se může přeložit jako „propojování otevřených systémů“. Úlohou referenčního modelu ISO/OSI je poskytnout základnu pro vypracování norem pro účely propojování systémů tak, aby zajistily spolehlivou a bezchybnou komunikaci. Norma tedy nespecifikuje realizaci systémů, ale uvádí všeobecné principy sedmivrstvé síťové architektury, která je zobrazena na Obrázku 1.

Referenční model OSI je tedy tvořen sedmi vrstvami a specifikuje protokoly na jednotlivých vrstvách a spolupráci mezi nimi. Jednotlivé vrstvy pro svou činnost využívají služeb své sousední nižší vrstvy a zároveň své služby pak poskytují sousední vyšší vrstvě. Podle referenčního modelu je zakázáno vynechávat vrstvy, ale některá vrstva nemusí být aktivní – takové vrstvě se říká nulová neboli transparentní. [1]

Obrázek 1 – Referenční model ISO/OSI

Zdroj: http://ijs.8u.cz/index.php/standardizace-v-pocitacovych-sitich/referencni-model-iso-osi

4.1.1. Fyzická vrstva

Nejnižší vrstva referenčního modelu specifikuje fyzickou komunikaci. Řeší se zde především technické parametry, jako jsou elektrické signály definující logickou nulu a jedničku, typy konektorů, druhy kabelů apod. Uskutečňuje přenos jednotlivých bitů komunikačním kanálem bez ohledu na jejich význam. Zároveň se zabývá také kódováním, modulací a synchronizací přenosu dat. Na této vrstvě pracuje například repeater, nebo HUB.

Mezi hlavní funkce fyzické vrstvy především patří řízení bitové rychlosti (počet odeslaných bitů za sekundu). V zásadě se dělí na tři typy: simplex, half-duplex a full-duplex, které definují, jakým způsobem budou data přenášena. Simplexní přenos umožnuje komunikaci pouze jedním směrem (od odesílatele k příjemci). Half-duplex slouží pro přenos dat oběma stranám, avšak nedochází k současnému přenosu (v jeden moment vždy data odesílá jedna strana). Full-duplex umožňuje současný přenos mezi oběma stranami. [1]

4.1.2. Linková vrstva

Tato vrstva je také někdy označována jako spojová vrstva, nebo vrstva datového spoje. Úkolem linkové vrstvy je spojení mezi dvěma sousedními systémy. Uspořádává data z fyzické vrstvy do logických celků známých jako rámce neboli frames o velikosti několika stovek bajtů. Fyzické rámce formátuje, opatřuje je fyzickou adresou a poskytuje synchronizaci pro fyzickou vrstvu. Pro tuto vrstvu je typický například bridge, nebo switch.

Linková vrstva kontroluje správnost přenosu pomoci CRC kontrolních součtů. Právě proto musí vždy přesně vědět kde který rámec začíná a kde končí. Další vlastnost linkové vrstvy je tzv. řízení toku. Jedná se o řízení rychlosti přenosu tak, aby příjemce bez problému stíhal odesílané rámce zpracovávat. [1]

4.1.3. Síťová vrstva

Tato vrstva zajišťuje adresaci a směrování dat mezi komunikujícími účastníky. Pod termínem směrování si můžeme představit nalezení vhodné cesty a zajištění správného předávání dat po této cestě. Od transportní vrstvy dostává data a adresu příjemce. Pro každý samostatný paket následně rozhoduje o směru odeslání. Nejznámějším protokolem na této vrstvě je pochopitelně IP (Internet Protocol). Na této vrstvě pracuje především router.

Pro tuto vrstvu je důležitá i funkce směrování neboli routing. Je to v podstatě rozhodování, jakým směrem budou data cestovat na základě znalosti topologie sítě. [1]

4.1.4. Transportní vrstva

Transportní vrstva uskutečňuje přenos dat mezi koncovými uzly. Rozděluje balíky odesílaných dat do paketů, které pak síťová vrstva posílá směrem k příjemci. Vrstva musí poskytnout právě takovou kvalitu přenosu, jakou požadují vyšší vrstvy. Zároveň tato vrstva zabezpečuje, aby se všechna data dostala k příjemci bezchybně. V případě chyby zajišťuje opakování zprávy a její opětovné sestavení. Nejznámějšími protokoly na této vrstvě jsou protokoly TCP (spojově orientovaný) a UDP (nespojově orientovaný). [1]

4.1.5. Relační vrstva

Smyslem této vrstvy je organizovat komunikaci mezi koncovými stanicemi. Realizuje zahájení i ukončení relačního spojení a zároveň zajišťuje práva, hesla, omezení i šifrování přenosu dat. Tato vrstva klade veliký důraz na synchronizaci – umožňuje vkládaní kontrolních (synchronizačních) bodů. Pomocí těchto bodů se zabraňuje ztrátám dat a zároveň se pomocí nich řídí dialog mezi zdrojovým a cílovým zařízením. Komunikace může být jednosměrná, obousměrná současná, nebo obousměrná střídavá. [1]

4.1.6. Prezentační vrstva

Prezentační vrstva má za úkol transformovat data z nejvyšší aplikační vrstvy tak, aby byla srozumitelná všem nižším vrstvám. Data pak zároveň na straně příjemce zpětně upravuje tak, aby je koncová zařízení dokázala bez problémů rozpoznat a předat konkrétním aplikacím. Proto je pochopitelně jednou z hlavních funkcí této vrstvy kódování neboli šifrování, o které se starají například protokoly SSL, TLS a ASCII. Můžeme si tedy tuto vrstvu představit jako takový překladač. Vrstva se nezajímá o význam dat, ale pouze o jejich strukturu. [1]

4.1.7. Aplikační vrstva

Sedmá a zároveň nejvyšší vrstva architektury vytváří rozhraní ke konkrétním programům. Definuje způsob, jakým aplikace (např. e-mail, databázové systémy, prohlížeče atd.) komunikují se sítí. Na této vrstvě pracuje celá řada protokolů, k těm nejznámějším patří například HTTP, FTP, DNS, DHCP, POP3, SMTP, SSH, Telnet. [1]

4.2. Transmission Control Protocol (TCP)

Známý spíše pod zkratkou TCP. Jedná se o protokol, který začal vznikat již na konci 60. let ve společnosti ARPA (Advanced Research Projects Agency) pod záštitou ministerstva obrany USA, která si tento a několik dalších protokolů nechávala vyvinout pro jejich nově vznikající síť zvanou ARPANET. Již tehdy se jednalo o zárodek vzniku něčeho daleko většího, něčeho, co po mnoha letech vývoje nazýváme Internet. Protokol TCP byl plně uveden do provozu až po více jak 14 letech, 1.1.1983. TCP je v dnešní době nejrozšířenější internetový přenosový protokol. Je na něm založena komunikace takřka všech zařízení připojených do veřejné sítě po celém světě. [2]

Protokol TCP používá ke spojení dvou uzlů protokol nižší síťové vrstvy – IP. Proto se velice často používá známá zkratka TCP/IP. S nadsázkou můžeme říct, že se jedná o „chytrý protokol na hloupé síti“, protože komunikace probíhá po aktivních zařízeních připojených do sítě. Ty se ovšem neustále mění (odpojují a zapojují), proto protokol musí být dostatečně „inteligentní“, aby vždy našel cestu do cílové adresy i na tak dynamicky proměnlivé síti, jako je Internet. Za opak tohoto protokolu lze považovat protokol IPX/SPX, který lze využívat výhradně v lokálních sítí, kde se aktivní zařízení nemění.

Koncept TCP se liší s jeho předchůdcem ISO/OSI především v tom, že tvůrci TCP považovali spolehlivost doručení datových paketů problémem koncových účastníků komunikace a mělo by být tedy řešeno až na úrovni transportní vrstvy. Z toho vyplívá řada výhod, ale bohužel i nevýhod tohoto protokolu. Za bezesporu největší výhodu lze požadovat garanci správnosti dat. TCP potvrzuje přijetí každého datového paketu a ověřuje, zda se po cestě žádný paket neztratil ani nepoškodil. Zároveň TCP zaručuje, že přenášená data přijdou vždy v pořadí, v jakém byla odeslána. Pokud TCP zjistí, že bylo některé z těchto zmíněných pravidel porušeno, neprodleně požaduje odeslání těchto dat znovu. Aby TCP tento standard dodržoval, musí do každé hlavičky paketu přidávat kontrolní informace. Ani TCP není dokonalý a má i několik nevýhod. První nevýhoda je, že každá hlavička paketu obsahuje mnoho řídících informací, se kterými se musí provádět řada kontrolních součtů, což přenos pochopitelně zpomaluje. Další nevýhodou je fakt, že TCP celkově vynakládá poměrně velkou zátěž na síť. Vzhledem k tomu, že je nutné potvrdit každý přijatý datový paket musí probíhat i kontrolní přenos v opačném směru ke zdroji. TCP na rozdíl od ISO/OSI počítá jen se čtyřmi vrstvami sítě. [3]

4.2.1. Vrstva síťového rozhraní

Vrstva síťového rozhraní (Network Interface Layer) je nejnižší vrstva, která má na starost vše, co je spojeno s přímým vysíláním a příjmem datových paketů. Příkladem této vrstvy je například síť Ethernet nebo Token ring. [3]

4.2.2. Síťová vrstva

Další v pořadí je Síťová vrstva (Network Layer). Tato vrstva má na starost zajištění komunikace mezi jednotlivými účastníky. Hlavním úkolem je tedy adresování a směrování. Typický protokol pro tuto vrstvu je opět IP. [3]

4.2.3. Transportní vrstva

Třetí vrstvou TCP je Transportní vrstva (Transport Layer). Vzhledem k tomu, že je nejčastěji realizována protokolem TCP, tak je také někdy nazývána jako TCP vrstva. Zajišťuje především spolehlivost přenosu (včetně kontroly dat). [3]

4.2.4. Aplikační vrstva

Poslední, a tedy nejvyšší vrstvou TCP je Aplikační vrstva (Application Layer). Pracuje především s protokoly, které komunikují s transportní vrstvou. Příkladem těchto protokolů může být například DNS, Telnet, DHCP, FTP, HTTP apod. Aplikační protokoly jsou rozlišeny pomocí tzv. portů – předem definovaného číselného označení aplikací. [3]

4.3. User Datagram Protocol (UDP)

User Datagram Protocol neboli UDP, jak se tento protokol běžně zkracuje, lze považovat za alternativu předchozího protokolu TCP. Protokol UDP je protokolem transportní vrstvy a stejně jako TCP slouží ke komunikaci po Internetové síti. Hlavní rozdíl mezi těmito dvěma protokoly je v záruce doručení dat. Jak bylo zmíněno výše, TCP zaručuje, že každý odeslaný paket dojde ke svému cíli, a navíc i ve správném pořadí. UDP ovšem tyto zásady nemá. Z toho pochopitelně vyplývá jeho největší nevýhoda. Při použití UDP v podstatě ztrácíme nad přenášenými daty do jisté míry kontrolu. UDP nijak nekontroluje doručení dat, a ani nijak nekontroluje pořadí doručených dat. Příjemce těchto datových paketů tedy nemá žádnou jistotu, že data dorazila v pořádku. Po těchto několika prvních větách se může zdát, že protokol UDP se v porovnání se spolehlivým protokolem

TCP nevyplatí nikdy upřednostňovat. Opak je však pravdou… TCP je sice bezpochyby globálně nejvyužívanější přenosový protokol, ale UDP má řadu svých výhod, které TCP poskytnout nemůže. UDP oproti TCP má daleko menší hlavičku (není třeba posílat kontrolní informace) a nepotřebuje odesílat zpětné potvrzení o přijetí každého datového paketu. Díky těmto vlastnostem lze považovat UDP za rychlejší a podstatně méně zatěžující síť.

UDP tedy své uplatnění nachází především u „real-time“ aplikací, u kterých není nutno doručit každý odeslaný paket. To jsou aplikace využívané například pro on-line hovory, videokonference, počítačové hry apod. UDP využívá samozřejmě i řada klíčových protokolů vyšších vrstev (jako například DNS, DHCP, RIP, IKE atd.). [4] [5]

4.4. IP adresa

Všechna zařízení, která mezi sebou chtějí komunikovat, musejí být jednoznačně identifikovatelná, což na tak ohromné síti jako je Internet, není vůbec jednoduché. K této identifikaci slouží právě Internet Protocol. Při každém přenosu dat po internetu je nutné znát IP adresu odesílatele a příjemce. O adresování a směrování datových paketů ke svému konkrétnímu příjemci se pak starají především směrovače (routery).

V současnosti je nejrozšířenější model IPv4, který vznikl už v roce 1981. Počet IP adres je samozřejmě omezený (u verze IPv4 je 2^{32} = 4 294 967 296 jedinečných IP adres). Tady narážíme na jeden z největších problémů Internetu současnosti. IPv4 adresy v určitých regionech už došly nebo rychle docházejí.

IPv4 je adresa tvořena 32bitovým číslem rozděleným tečkou po čtyřech oktetech. V zásadě se dělí na tři základní části. Prvních osm bitů je číslo sítě, druhých osm bitů je číslo podsítě a poslední dvě osmice bitů tvoří číslo síťového rozhraní. Podle toho, jak jsou jednotlivé sítě rozsáhlé, rozlišujeme tři hlavní třídy IP adres – A, B a C. Třídy jsou rozdělovány pomocí masky sítě. Je důležité si uvědomit, že počítač nepracuje s dekadickou soustavou, nýbrž s binární. Takže například IP adresa 192.168.1.5 vypadá pro počítač následovně: 11000000.10101000.00000001.00000101 (každé dekadické číslo oddělené tečkou se samostatně převede do binárního tvaru). [2] [6]

4.4.1. Maska sítě

Slouží k rozdělení počítačové sítě na podsítě. V podstatě definuje, jaká část IP adresy je síťová a jaká síťového rozhraní. V binárním tvaru nese jedničky tam, kde se v adrese nachází číslo sítě a podsítě, a nuly tam, kde je číslo síťového rozhraní. Běžně se setkáme

se zápisem masky sítě: 255.255.255.0, kde je pro adresu síťového rozhraní určena poslední osmice bitů IP adresy. I zde je důležité si uvědomit, že počítač pracuje s binární soustavou. Převod je zde obdobný jako v předchozí kapitole. [7]

4.4.2. Brána sítě

Takto je označován uzel, který má v síti nejvyšší postavení. Tvoří ho zpravidla aktivní hardwarový prvek, nejčastěji router, který přeposílá datové pakety do koncových zařízení. Přes toto zařízení pak probíhá veškerá komunikace v dané síti. Router má dvě adresy, jednu vnější (veřejnou) přes kterou komunikuje s ostatními aktivními prvky ve veřejné síti a druhou vnitřní tzv. implicitní bránu (default gateway), přes kterou komunikuje se všemi zařízeními, které jsou do něj zapojené v LAN síti. [8]

4.4.3. Pravidla přidělování IP adres

Při přidělování IP adres musí být dodrženy dvě základní pravidla. Pokud se dvě a více síťových rozhraní nachází ve stejné síti, musí mít jejich IP adresy totožnou síťovou část (Network ID) a pochopitelně jedinečnou klientskou část (Host ID) pro každé rozhraní. Pokud se dvě síťová rozhraní nachází v různých sítích, musí mít jejich IP adresy odlišnou síťovou část a klientská část může i nemusí být stejná. [9]

Pro automatické přidělování IP adres slouží DHCP (Dynamic Host Configuration Protocol). Na hardwarovém zařízení (nejčastěji routeru), je spuštěný DHCP server, který všem zařízením, jenž jsou do něj zapojená, propůjčuje na určitou dobu jedinečnou IP adresu a v případě potřeby zapůjčení prodlužuje. Zároveň přiděluje i další parametry: masku sítě, primární a sekundární DNS server a výchozí bránu. DHCP je protokol z rodiny TCP/IP, díky tomu může jednotlivým zařízením přidělovat automaticky potřebné parametry, čímž podstatně centralizuje a usnadňuje správu sítě. Na routeru je vždy i možnost vypnutí DHPC serveru a jednotlivým zařízením přidělit IP adresu a ostatní parametry ručně, tzv. staticky. [10]

4.5. Směrování

Dnes si velikost Internetu už nedokáže ani zdaleka nikdo představit. Je tvořen nespočetným množstvím zařízení, které denně využívají miliardy uživatelů a přes tuto nepředstavitelně komplexní síť mohou mezi sebou komunikovat i na vzdálenost tisíců kilometrů ve zlomku vteřiny. Řadu uživatelů už jistě napadlo, jak si jimi odeslaná zpráva

dokáže najít cestu takřka kamkoliv, a navíc během pár vteřin. K tomu, aby si odeslaný datový paket vždy našel cestu ke svému příjemci přes neustále se měnící síť jako je Internet slouží právě směrování neboli routování.

Nedílnou součástí tohoto procesu je zařízení zvané router. Ten má za úkol zpracovat každý příchozí datový paket a poslat ho na správnou cestu. Aby se datový paket vydal vždy právě na tu správnou cestu (nejkratší a nejefektivnější) se využívá tzv. Routing protokol. Routing protokol zároveň posílá okolním zařízením routovací informace na jejichž základě si pak ostatní routery v okolí udržují směrovací tabulku (routing table), která pak představuje zjednodušený obraz topologie sítě.

Dnes už se pro routování nevyužívá výhradně jen router, je sice stále nejrozšířenější, ale k tomuto procesu se dále využívají například i L3 switche, firewally, koncová zařízení nebo i servery.

Celý proces routování musí probíhat co nejrychleji, aby nezpomaloval provoz sítě. Při vývoji směrovacích protokolů byl proto kladen velký důraz nejen na algoritmy výpočtu správné a nejrychlejší cesty, ale také na uchování dostatečného množství informací, které jsou ukládány a nadále udržovány v těchto směrovacích tabulkách.

Pro celkové zrychlení routování se routing rozděluje na statický a dynamický. Statické routování se využívá především na malých sítích, kde nedochází k častým změnám zapojených zařízeních. Díky tomu se do směrovací tabulky nemusí často zasahovat a měnit jí. To samozřejmě podstatně zkracuje rychlost vyhledávání cesty k adresátovi. Naopak u velkých sítí, kde často dochází ke změnám aktivních zařízení se využívá dynamické routování. To spočívá v komunikaci se svými sousedními zařízeními, které si pomocí přenosových protokolů vyměňují informace o svých dalších „sousedech“ a vytváří si svou propojovací tabulku. Je na administrátorovi, aby zvážil velikost spravované lokální sítě a posoudil, jestli se vyplatí využít spíše statické nebo dynamické routování. [11]

4.5.1. Směrovací protokoly

Routery si díky směrovacím protokolům udržují přehled o topologii sítě. V zásadě tyto protokoly definují, jak si routery mezi sebou vyměňují informace a díky těmto informacím pak routery dokážou směrovat datové pakety ke svému příjemci. Každý router má při startu znalost pouze o síti, ke které je přímo připojený. Směrovací protokol sdílí tyto informace nejprve mezi bezprostředními sousedy a následně i v celé síti. Ačkoli existuje mnoho typů směrovacích protokolů, v sítích IP se široce používají tyto hlavní tři třídy:

- **Interior gateway protocols type 1:**

Jsou založen na stavu linek (příklad protokolů: OSPF, IS-IS).

- **Interior gateway protocols type 2:**

Jsou založen na vektoru vzdálenosti (příklad protokolů: RIP, RIPv2, IGRP).

- **Exterior gateway protocol:**

Výměna informací mezi autonomními systémy (příklad protokolu: BGP).

Jak je zřejmé existuje celá řada konkrétních typů směrovacích protokolů, které mají své nejrůznější uplatnění. Vzhledem k narůstajícím požadavkům na výkon a velikost sítě se samozřejmě tyto protokoly v průběhu let museli podstatně zdokonalovat a přizpůsobovat vývoji Internetu. [11] [12]

4.5.2. Směrovací tabulka

Směrovací tabulka obsahuje základní informace o topologii sítě v blízkosti daného zařízení. Tabulka v podstatě představuje datovou strukturu, která je uložena v paměti routeru nebo v operační paměti počítače. Záznamy ve směrovací tabulce jsou buď statické (zapsané správcem počítače), nebo dynamické. Dynamické záznamy jsou zapisovány nebo odebírány ze směrovací tabulky pomocí směrovacích protokolů, jejichž funkce je popsána v předchozí kapitole. U některých velkých sítích se může stát, že k jedné IP adrese odkazují dvě nebo i více rozhraní. V tomto případě samozřejmě nejsou v tabulce zaneseny všechny možné cesty, ale pouze ty nejkratší (podle největší masky sítě). V tabulce se musí minimalizovat množství irelevantních informací, aby se maximalizovala přehlednost a rychlost vyhledávání. Základní informace o směrovací tabulce se dají snadno dohledat pomoc příkazového řádku (CMD) a příkazu „*route print*“.

Záznamy jsou v tabulce seřazeny od nejkonkrétnějších po nejobecnější. Jak moc je daný záznam konkrétní je určeno právě pomocí délky masky sítě, která rozděluje IP adresu na dvě části – adresu sítě a adresu uzlu. Takže na prvních místech jsou IP adresy s největší maskou sítě (tj. 255.255.255.255) a na konci je záznam s nejmenší maskou sítě (tj. 0.0.0.0) pro tzv. implicitní bránu.

Celý proces započne v momentě, kdy router přijme datový paket. V hlavičce paketu zjistí, kam paket směřuje a snaží se dohledat shodu adresáta ve své směrovací tabulce. Pomocí těchto informací se router rozhoduje, kterým portem pošle datový paket dál. V případě, že IP adresa adresáta datagramu neodpovídá žádnému konkrétnímu záznamu,

tak datagram odešle právě na implicitní bránu, což je obecný název pro nadřazenou počítačovou síť. [11] [13]

4.5.3. Směrovací algoritmus

Informace ze směrovací tabulky musejí být zpracovány a vyhodnoceny tak, aby výsledná cesta pro každý datový paket byla co nejefektivnější. Obecně vzato tuto cestu musí určit síťová vrstva, na které právě pracuje směrovací algoritmus. Metoda výpočtu je volena na základě směrovacích protokolů. Směrovací algoritmy se v zásadě dělí na dvě hlavní skupiny: neadaptivní a adaptivní.

Neadaptivní směrování typicky používají routery nebo koncové stanice v malých počítačových sítích, protože nevyužívají žádné informace dynamického charakteru. Nevyužívají tedy údaje o okamžitém zatížení jednotlivých přenosových cest, výpadcích aktivních zařízeních, délkách čekacích front v uzlech atd. Předpokladem je tedy to, že záznamy ve směrovací tabulce se v průběhu činnosti nemusejí nutně měnit. Naopak adaptivní směrování má své uplatnění na větších sítích, protože se dynamicky přizpůsobuje jak topologii sítě, tak i síťovému provozu. Adaptivní algoritmy se pak rozdělují do několika menších celků, jejichž principy jsou zde velice stručně popsány: [12]

- **Centralizované směrování:**

  Informace o aktuálním stavu sítě se shromažďují v tzv. RCC – Routing Control Center (Směrovacím centru), které na základě těchto informací sestavuje mapu sítě, vytváří směrovací tabulky a zasílá je směrovačům. [12]

- **Izolované směrování:**

  O nejvhodnější cestě rozhoduje samostatně každý uzel, na základě informací, které získá sám, bez spolupráce s ostatními uzly. [12]

- **Záplavové směrování:**

  Vychází z izolovaného směrování. Každý přijatý paket je odeslán všemi směry, kromě toho, odkud sám přišel. Touto metodou jsou zkoušeny v podstatě všechny cesty a tak se vždy najde i ta nejefektivnější. [12]

- **Zpětné učení:**

  Na základě příchozích a odchozích paketů se algortimy „učí“, ve kterém směru se jednotlivé uzly nalézají. [12]

- **Distribuované směrování:**

  Jednotlivé uzly si mezi sebou vyměňují informace o stavu sítě a na základě těchto informacích se rozhodují, jakým směrem každý datový paket odešlou. [12]

4.6. Virtual private network (VPN)

Virtual Private Network lze přeložit jako virtuální privátní síť, kterou zná většina lidí pod běžně používanou zkratkou VPN. Jedná se o službu, pomocí které se lze bezpečně připojit v podstatě z jakéhokoliv místa na světě k vzdálené lokální síti, zařízení či serveru. Službu VPN dříve využívaly pouze velké společnosti, aby se jejich zaměstnanci mohli z domovu připojit bezpečně do pracovní sítě, ale dnes se tato služba používá i v běžných domácnostech.

Pro zjednodušení si můžeme představit, že služba VPN vytváří jakýsi „tunel“ přes veřejnou nedůvěryhodnou počítačovou síť (Internet), do lokální sítě. Lze takto dosáhnout stavu, kdy spojené počítače budou mezi sebou moci komunikovat, jako kdyby byly propojeny v rámci jediné uzavřené privátní sítě. Veškerý přenos dat probíhá šifrovaně, proto lze tuto komunikaci považovat za bezpečnou. VPN se dělí na tři základní způsoby připojení: [6]

- **Bod – Bod (Point–to–Point):**

  Dnes pravděpodobně nejběžnější případ využití VPN, kdy se uživatel připojuje z jednoho uzlu v síti na jiný vzdálený uzel v jiné síti. Příkladem může být i běžné připojení klienta přes Internet k zabezpečené webové bankovní aplikaci. [6]

- **Bod – Síť (Point–to–Site):**

  Tento typ připojení se nejběžněji používá v podnicích, kdy se zaměstnanec pracující z domova připojuje do vzdálené lokální sítě společnosti. Díky VPN se počítač tváří jako by byl připojený přímo v kanceláři a tím umožní přístup například do zabezpečeného intranetu podniku, nebo k dokumentům v síti. [6]

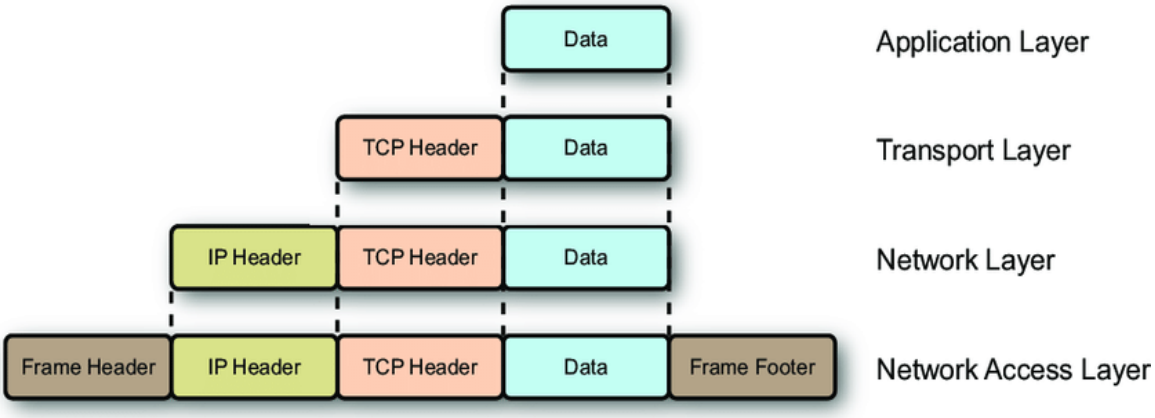
- **Síť – Síť (Site–to–Site):**

  Propojení dvou vzdálených lokálních sítí. Počítače zapojené v obou sítích se pak tváří jako by byly společně v jedné velké síti. To má své uplatnění zejména pokud podnik má dvě vzdálené pobočky a potřebuje, aby všechny počítače byly společně v jedné velké síti. [6]

4.6.1. Síťové tunelování

Se službou VPN je neodmyslitelně spojen i termín „tunelování“. Jak bylo zmíněno výše „VPN vytváří zabezpečený tunel napříč nezabezpečenou sítí“. Jak ale takový tunel vypadá? Po technické stránce tunel vůbec nepřipomíná. Tunelování se provádí zapouzdřením datového paketu přenášeným jedním protokolem do jiného protokolu. Paket tvoří dvě hlavní části: záhlaví a uživatelská data. Řídící data jsou nejčastěji umístěna v hlavičkách paketů (záhlaví), které mohou být na začátku i na konci datového paketu. Uživatelská data jsou pak samozřejmě mezi nimi. Tunelování pak v praxi vypadá tak, že je před původní paket přidána nová hlavička protokolu, do kterého se zapouzdřuje. Jak mohou vypadat zapouzdřená data napříč vrstvami je znázorněno na Obrázku 2. To má své uplatnění především v sítích, které přijmou pouze určité druhy protokolů. V případě šifrovacích protokolů je obsah původního paketu navíc i zašifrován. [14]

Obrázek 2 – Tunelování – Zapouzdření paketu



Zdroj: https://www.researchgate.net/figure/

4.6.2. Zabezpečení VPN

Nejdůležitějším úkolem VPN je zajištění bezpečného přenosu dat k oprávněnému subjektu. Existuje celá řada zabezpečovacích protokolů, které právě toto mají na starosti. V zásadě lze shrnout požadavky na bezpečnost VPN v těchto čtyřech bodech: [6]

- **Autentizace:**

  Tento proces slouží k ověření totožnosti koncových uživatelů využívajících spojení VPN. Mezi nejběžnější autentizační metody patří využití tajného klíče, bezpečnostního certifikátu, nebo hesla. Autentizace může probíhat obousměrně nebo jednosměrně,

při čemž se prokazuje jenom jedna strana. K autentizaci lze pro větší míru bezpečnosti využít i třetí strana, která sděluje tajné heslo nebo klíč koncovému uživateli. [6]

- **Autorizace:**

  Po úspěšné autentizaci následuje autorizace. Autorizace určuje práva přihlášeného uživatele. Omezuje jeho přístupy, práci s daty, provádění specifických příkazů atd. Pro zajištění autorizace se například využívají protokoly TACACS nebo RADIUS. [6]

- **Integrita:**

  Záruka bezchybného přenesení všech odesílaných dat. Jinak řečeno: „Data musejí dorazit ke svému příjemci v takovém stavu, v jakém byla odeslána“. Integritu dat běžně zajišťují šifrovací algoritmy, hashovací funkce a nejrůznější typy opravných kódů. Porušení integrity může být i následek pokusu o neoprávněný přístup potenciálního útočníka. [6]

- **Kryptografie:**

  Hlavním úkolem kryptografie neboli šifrování je znemožnit potenciálním útočníkům odposlouchávat odesílané informace v průběhu přenosu (tzv. man in the middle). Respektive musí přetvořit odesílané informace takovým způsobem, aby v průběhu cesty přes veřejnou nezabezpečenou síť neměly pro potenciálního útočníka žádnou hodnotu. Toto téma je podrobněji rozebráno v následující kapitole. [6]

4.7. Kryptografie

Šifrování je tedy metoda, která utajuje smysl zprávy, tak aby jí bylo možné přečíst pouze s nějakou speciální znalostí – klíčem. Tuto znalost by měla mít pouze oprávněná osoba, jinak šifrování zcela ztrácí smysl. Mezi neodbornou veřejností je zřejmě nejznámější dnes už poměrně primitivní Caesarova šifra, kterou využíval pro komunikaci už Julius Ceasar kolem 50. roku př.n.l v průběhu Galských válek. Ceasarova šifra zamění všechna písmena ve zprávě za jiné písmeno, jenž se nachází v abecedě o pevně určený počet míst dále. Další bezpochyby známý šifrovací model byl například kód enigmy, kterým za 2. Světové války nacisti šifrovali zprávy. Nedílnou součástí této metody byl samotný stroj Enigma. Tento stroj ve všech možných nastaveních umožňoval až 159×10^{18} možných řešení. Jak je zřejmé, snaha šifrovat informace tu je už od pradávna a během těchto mnoha let se také značně zdokonalila. S dokonalejším šifrováním vždy přicházejí i dokonalejší metody jak šifrování naopak prolomit. Za nejbezpečnější šifrovací protokol lze pravděpodobně dnes

považovat AES–256 (Advanced Encryption Standard), který používá více šifrovacích kombinací, než je hvězd ve známém vesmíru. [6] [15]

4.7.1. Symetrické a asymetrické šifrování

Šifrování lze rozdělit do dvou velkých skupin, a to na symetrické šifrování a asymetrické šifrování. Rozdíl mezi těmito dvěma skupinami je v počtu používaných klíčů.

Symetrické šifrování využívá pouze jeden jediný klíč, jak pro šifrování a dešifrování tak i pro autentizaci. Tento klíč musí znát pouze a jedině oprávněný subjekt. Proto si musí obě strany před začátkem komunikace tento tajný klíč vyměnit i s informacemi o použitém algoritmu. Velkou výhodou symetrického šifrování je především nízká výpočetní náročnost a s tím související rychlost šifrování a dešifrování. Další výhodou je fakt, že i při relativně krátké délce klíče je i pro současnou techniku obtížné klíč uhádnout. Se vzrůstající délkou klíče exponenciálně vzrůstá i náročnost na jeho odhalení. Naopak nevýhodou je právě nutnost domluvy obou stran na totožném tajném klíči. Tento klíč nelze zasílat v nezašifrované formě! Kdyby se k němu útočník dostal, mohl by bez problémů dešifrovat veškerou následující komunikaci. Symetrické šifrování se ještě dál dělí na proudové a blokové šifry. Proudové šifry šifrují a dešifrují zprávu pomocí klíče bit po bitu. Oproti tomu blokové šifry šifrují a dešifrují zprávy po stejně velkých bitových blocích (nejčastěji 64, 128 nebo 256 bitů). Mezi typické algoritmy blokových šifer patří například AES, 3DES, BLOWFISH, KERBEROS atd.

Asymetrické šifrování na rozdíl od předchozí metody využívá dva klíče – veřejný a soukromý. Oba tyto klíče jsou propojeny a tvoří tak klíčový pár. Jak už názvy napovídají, veřejný klíč každého uživatele je dostupný v podstatě všem (existují specifické případy asymetrického šifrování, kdy i veřejný klíč musí zůstat v tajnosti) a slouží k šifrování zpráv. Soukromý klíč má každý uživatel svůj vlastní a musí ho držet v tajnosti, protože se naopak využívá k dešifrování zpráv. V praxi pro zajištění bezpečné komunikace mezi dvěma koncovými uživateli pak strana autora zašifruje zprávu pomocí veřejného klíče adresáta. Adresát pak tuto zprávu dešifruje svým soukromým klíčem. Největší výhodou asymetrického šifrování je absence vzájemného přeposílání klíče na začátku komunikace. Zprávy jsou šifrovány veřejným klíčem, ale ten nelze použít pro dešifrování. Zároveň tato metoda díky veřejnému klíči umožňuje i autenticitu – identifikaci uživatelů. Nevýhodou této metody je vysoká výpočetní náročnost (oproti symetrickému šifrování může být až tisíckrát

náročnější). Mezi nejznámější asymetrické algoritmy patří například RSA, DSA, Diffie-Hellman atd.

Často se tyto dvě šifry používají ve společné kombinaci. V takovém případě si pak obě strany pomocí asymetrického šifrování vymění klíč pro symetrické šifrování. Obě strany pak bezpečně získají klíč a komunikace může probíhat pomocí symetrického šifrování, a tudíž podstatně rychleji s menší výpočetní náročností. Takovéto metodě, kde je využita rychlost symetrického šifrování a bezpečnost výměny klíčů asymetrického šifrování se říká hybridní šifrování. [16]

4.7.1.1. Triple Data Encryption Standard (3DES)

Pod zkratkou DES je v kryptografii označována bloková symetrická šifra s 56bitovým klíčem vyvinutá v 70. letech dvacátého století. Samotný DES ovšem v dnešní době vzhledem k moderním kryptoanalytickým technikám a počítačovému výkonu nelze považovat za spolehlivý. Aby nebylo třeba přejít ke zcela novému algoritmu, začala se prosazovat jeho vylepšená verze 3DES (někdy taky označována jako TDES – Triple DES), která díky většímu klíči poskytuje mnohem lepší zabezpečení. 3DES je v podstatě jen trojnásobnou aplikací šifry DES, kdy jsou data několikrát zašifrována a dešifrována pomocí několika různých klíčů. Tyto tři klíče pak dohromady tvoří tzv. TDES klíč. V praxi se nejčastěji používá šifrování typu EDE (Encrypt–Decrypt–Encrypt). Při této metodě jsou data nejprve zašifrována prvním klíčem, následně dešifrována druhým klíčem a v posledním kroku znovu zašifrována třetím klíčem. Varianty používaných klíčů se dělí na tři způsoby:

- **Varianta 1** – všechny tři klíče jsou na sobě nezávislé (168bitů)
- **Varianta 2** – první klíč je stejný jako třetí, ale druhý klíč je na nich nezávislý (112bitů)
- **Varianta 3** – všechny tři klíče jsou stejné (56bitů)

První varianta s nejdelší délkou klíče 3x56=168 je samozřejmě považována za nejbezpečnější, a proto je i nejčastěji využívána. Oproti tomu je třetí varianta nejméně bezpečná. Při metodě šifrování EDE se první a druhá operace vzájemně vyruší. Čímž se v podstatě vrátí k šifrování DES, které nelze považovat za bezpečné. Nutno ovšem podotknout, že oproti novým algoritmům je 3DES vzhledem k jeho stáří považován za pomalý, a tak se od jeho používání pomalu ustupuje. Přeci jen, od vytvoření první verze DES uplynulo už necelých 50 let. [17]

4.7.1.2. Advanced Encryption Standard (AES)

AES je kryptografický algoritmus, který lze použít k ochraně elektronických dat. Jedná se o podmnožinou blokové šifry Rijndael. Název Rijndael je přesmyčka dvou belgických autorů Joana Daemena a Vincenta Rijmena, kteří algoritmus publikovali v roce 1997 ve veřejné soutěži institutu NIST. V témže roce NIST zahájil i samotný vývoj AES, protože pomalu, ale jistě potřebovali alternativu k šifrovacímu algoritmu DES, který se začínal stávat zranitelným. V roce 2001 byl AES publikován jako federální (USA) norma pro zpracování informací – FIPS 197.

AES je stejně jako DES bloková šifra, a data tedy před šifrováním rozděluje do bloků o pevně dané délce 128bitů. Ovšem na rozdíl od DES se jedná o symetrický algoritmus, což znamená, že k šifrování i dešifrování používá pouze jeden klíč o délce 128, 196 nebo 256 bitů. Samotné šifrování algoritmu je založeno na principu známém jako substitučně–permutační síť a je velice efektivní jak pro software, tak hardware.

Na začátku procesu šifrování jsou data nejprve rozdělena do bloků. Velikost bloku je 128 bitů, takže data jsou rozdělena do sloupců čtyři ku čtyřem o šestnácti bajtech (16 x 8 = 128 [bajt = 8 bitů]). Do stejného bloku 4x4 je posléze převeden i používaný klíč. AES šifruje data jak pomocí tohoto klíče, tak později i pomocí tzv. Round Key, který vznikne přetvořením původního klíče na základě Rijandaelova klíčového plánu.

Následně probíhá celá řada matematických operací, které přetváření původní bajty v jednotlivých sloupcích a řádcích každého bloku. Tyto operace jsou rozděleny do několika příslušně pojmenovaných sekcí:

- **Add Round Key** – šifrování pomocí aditivní operace XOR a příslušného klíče
- **Substitute bytes** – nahrazení bajtů podle stanovené tabulky
- **Shift rows** – každý řádek je posunut o určitý počet míst
- **Mix columns** – kombinace čtyř bytů v každém sloupci

Jednotlivé sekce se několikrát opakují v závislosti na délce používaného klíče (128 bitů – 9 kol, 196 bitů – 11 kol, 256 bitů – 13 kol).

AES je při správné implementaci do dnešního dne bezpečnostními experty považován za bezpečný. Existuje sice několik úspěšných útoků, ale nikdy se nejednalo přímo o chybu šifry AES. K prolomení totiž došlo pomocí tzv. útoku postranním kanálem. Tento druh útoku nenapadá šifru samotnou, ale útočí na její implementaci na daném systému. S bezpečností šifry tedy nemá nic společného. [18] [19]

4.7.1.3. RSA (Rivest, Shamir, Adleman)

Název tohoto šifrovacího algoritmu je složen z iniciály tří hlavních autorů Ron Rivest, Adi Shamir a Leonard Adleman, kteří algoritmus veřejně publikovali v roce 1978. Algoritmus je založen na Eulerově větě a na již zmiňovaném asymetrickém šifrování. Pro svou funkci tedy využívá jeden veřejný (šifrovací) klíč, který je volně dostupný pro ostatní uživatele a druhý soukromý (dešifrovací) klíč, který je udržován v tajnosti.

Algoritmus pak zjednodušeně funguje tak, že uživatel vytvoří a zároveň publikuje veřejný klíč [z], jenž je výsledkem součinu dvou velkých prvočísel [x, y] (spolu s pomocnou hodnotou). Prvočísla jsou udržována v tajnosti jakožto soukromý klíč. Pro rychlejší zpracování se v praxi klíče uchovávají v mírně upravené formě. Zprávy tedy mohou být šifrovány kýmkoli prostřednictvím veřejného klíče [z], ale mohou být dekódovány pouze někým, kdo zná obě prvočísla [x, y]. Je velmi důležité, aby nikdy nebyla zvolena příliš krátká prvočísla, jinak by celý systém byl náchylný na tzv. útok hrubou silou, při kterém se útočník snaží zjistit soukromé klíče pomocí faktorizace veřejného klíče uživatele.

Jednalo se o první algoritmus, který byl vhodný jak pro šifrování, tak pro podepisování. Používá se až do dnes, což z něj dělá jeden z nejstarších a nejpoužívanějších šifrovacích algoritmů vůbec. Jeho úroveň zabezpečení je v první řadě založena na délce klíče. Při použití dostatečně dlouhého klíče neexistují žádné publikované metody, jak tento systém prolomit. RSA lze ovšem považovat za relativně pomalý algoritmus. Proto se běžně nepoužívá k přímému šifrování dat, ale spíše k přenosu sdílených symetrických klíčů, které se pak používají pro hromadné šifrování a dešifrování. [20]

4.7.2. Metody šifrování VPN datagramů

VPN obecně využívá tří metody jak datagramy šifrovat: Payload encryption, IP-in-IP tunneling a IP-in-TCP tunneling. [6]

- **Payload encryption:**

  Aby komunikace mezi koncovými zařízeními byla zabezpečená musí dojít k šifrování datagramů. Při této metodě dochází k zašifrování pouze těla datagramu a hlavička zůstává nezašifrována, čímž nedochází k zabezpečení zdrojové adresy, cílové adresy ani čísla portu. [6]

- **IP-in-IP Tunneling:**

  Některé VPN používají technologii IP-in-IP Tunneling, která odchozí datagramy zašifruje celé, včetně hlavičky a informace o přenosu pak vloží do jiného datagramu. Tento postup je o něco bezpečnější než u předchozí metody. [6]

- **IP-in-TCP Tunneling:**

  Třetí alternativa, používána k zabezpečení přenosu, zahrnuje využití tzv. TCP tunelu. Obě koncová zařízení navážou připojení TCP a pak přes toto připojení odesílají zašifrované datagramy. K datagramům je přidána malá hlavička, která označuje hranici mezi jednotlivými datagramy. Hlavička se obvykle skládá z dvoubajtového čísla, které určuje délku datagramu. Koncové zařízení pomocí VPN softwaru přečte hlavičku datagramu a zjistí počet zadaných bajtů. Pomocí tohoto čísla sestaví původní datagram, který následně může dešifrovat. Hlavní výhoda této metody oproti IP-in-IP Tunneling je spolehlivost. TCP zajišťuje, že všechny datagramy dorazí spolehlivě do cílové adresy. [6]

4.8. Point–to–Point Tunneling (PPTP)

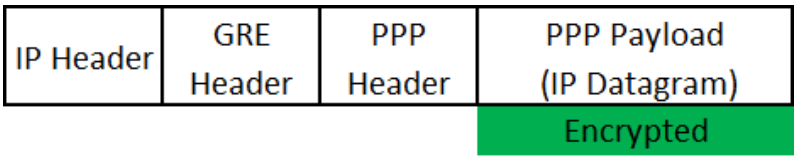
Jednou z nejstarších realizací VPN je Point–to–Point Tunneling Protocol (PPTP). Protokol původně nebyl schválen jako standard IETF. Na jeho vývoji pracovalo hned několik firem, jako například Microsoft, 3COM, US Robotics a další. Stal se velice populární právě díky společnosti Microsoft, protože to byl jejich první podporovaný VPN protokol. Právě díky tomu a jednoduchosti konfigurace se stává velice často primární volbou při výběru VPN. PPTP používá TCP port 1723. Ovšem nutno zdůraznit, že tento protokol v dnešní době nelze považovat za zcela bezpečný. K zabezpečení komunikace totiž primárně využívá zastaralý autentizační protokol MS–CHAPv2. [21]

4.8.1. PPTP – Princip

PPTP ke své činnosti využívá dva důležité protokoly: PPP (Point–to–Point Protocol) k vytvoření relace prostřednictvím sítě IP a vylepšenou verzi GRE (Generic Routing Encapsulation) k zapouzdření paketů (tunelování). Pomocí PPP se přes GRE tunel připojí k přístupovému serveru. Následně se spustí PPTP klient, který vytvoří zabezpečené spojení s cílovým PPTP serverem, ke kterému má klient přístupová práva a zároveň je dosažitelný v rámci směrovacích informací (ty jsou obsaženy v poslední přidané IP hlavičce paketu). Viz Obrázek 3, kde je znázorněno PPTP zapouzdření paketu. Pro autentizaci využívá

již zmiňovaný protokol MS-CHAPv2. PPTP navíc umožňuje (nepovinně) šifrovat přenos pomocí MPPE (Microsoft Point-to-Point Encryption), který využívá šifrovací algoritmus RSA (Rivest-Shamir-Aldeman) RC4. [21]

Obrázek 3 – Zapouzdření PPTP paketu



Zdroj: Vlastní

4.8.1.1. Point–to–Point Protocol (PPP)

Jedná se o relativně starý komunikační protokol navržený již v roce 1994. Protokol poskytuje spojení a přenos datagramů mezi dvěma síťovými uzly. PPP je vrstvený protokol tvořený třemi složkami:

- Metoda pro zapouzdření datagramů s více protokoly.
- Link Control Protocol (LCP) pro navázání, konfiguraci a testování připojení datového spoje.
- Rodina protokolů Network Control Protocols (NCP) pro vytváření a konfiguraci protokolů síťové vrstvy.

PPP zároveň podporuje protokoly PAP (Password Authentication Protocol), MPPE (Microsoft Point-to-Point Encryption) a CHAP / MS–CHAP / MS–CHAPv2 (Challenge Handshake Authentication Protocol), které po ověření PPTP spojení využívá. [22]

4.8.1.2. Generic Routing Encapsulation (GRE):

GRE je tunelovací protokol vytvořený rovněž v roce 1994 společností Cisco Systems. Protokol poskytuje přenos paketů jednoho protokolu zapouzdřeného přes jiný protokol. Nejprve je paket zapouzdřen v paketu GRE (před paket je vložena GRE hlavička) a následně je celý tento paket zapouzdřen v doručovacím IP protokolu (před GRE hlavičku je vložena IP hlavička). [14] [23]

4.8.1.3. MS–CHAPv2

CHAP (Challenge-Handshake Authentication Protocol) je autentizační protokol společnosti Microsoft (MS) především využívaný v sítích VPN. Existuje ve dvou verzích,

ale v rámci PPTP se používá pouze verze 2. Právě tento protokol představuje nejzranitelnější místo PPTP. CHAP je totiž mimo jiné velice zranitelný proti tzv. slovníkovým útokům. Princip těchto útoků spočívá ve snaze uhodnout heslo z předem připraveného seznamu často používaných hesel. Výběr hesla zde tedy hraje klíčovou roli. Ovšem dnes existují metody, které dokážou tento protokol prolomit v řádu několika hodin. Microsoft samotné použití tohoto protokolu na nezabezpečené síti nedoporučuje. Pro zajištění další ochrany (především u bezdrátových sítí) lze použít EAP (Extensible Authentication Protocol). [24]

4.8.1.4. Microsoft Point-to-Point Encryption (MPPE)

MPPE poskytuje zabezpečení dat pro připojení PPTP, které je mezi klientem a serverem VPN. Nejedná se přímo o šifrovací algoritmus, nýbrž o metodu šifrování dat přenášených přes vytáčené připojení založené na protokolu PPP nebo VPN. MPPE pro šifrování dat využívá šifrovací algoritmus RSA s podporou 40bitových a 128bitových klíčů relace. Ty by se kvůli zajištění vyšší bezpečnosti měly často měnit. Maximální délka MPPE datagramu, přenášeného přes PPP spojení je stejná jako maximální délka zapouzdřeného paketu informačního pole PPP. Jsou šifrovány pouze pakety s čísly protokolu v určitém rozsahu (0x0021 až 00FA).

Samotný MPPE nijak nekomprimuje ani nerozšiřuje data, ale protokol se často používá ve spojení s Microsoft Point-to-Point Compression, který data komprimuje. [25]

4.9. Internet Protocol Security (IPsec)

Internet protocol security neboli zkráceně IPsec je protokol pracující na třetí vrstvě modelu ISO/OSI využívající UDP porty 500 a 4500. Slouží k zabezpečení komunikace na základě autentizace a šifrování každého datagramu. IPsec byl původně navržen jako součást tenkrát nově navrhovaného síťového protokolu IPv6. Vývoj IPv6 se ovšem opozdil, a tak byl nakonec IPsec vydán předčasně. Díky kvalitnímu řešení a vysoké úrovni zabezpečení kterou poskytoval, se IPsec velice rychle uplatnil právě pro zabezpečování VPN tunelů. K realizaci IPsecu point–to–site na Windows platformách je nutné využít klientský software třetí strany. [26]

4.9.1. IPsec – Princip

Před začátkem komunikace se pomocí protokolu IKE koncová zařízení předem domluví na formě šifrování a tajném klíči. Následně je každý datagram napříč celou

komunikací podle toho zašifrován. IPsec zároveň ověřuje i původ každého přijatého datagramu. Díky těmto zabezpečovacím prvkům představuje IPsec jednu z nejspolehlivějších šifrovacích metod pro zabezpečení VPN. Zároveň je ale také poměrně náročný na správou implementaci. IPsec navíc pracuje v jednom ze dvou režimů, které se odlišují podle způsobu šifrování hlavičky datagramu. Těmto dvěma režimům se říká transportní a tunelovací mód. [27]

- **Transportní mód:**

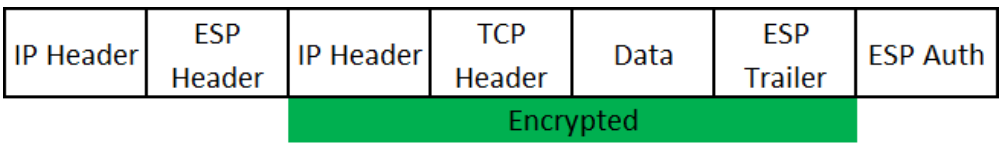
  V tomto režimu IPsec zašifruje pouze obsah zprávy datagramu a hlavička zůstává takřka vůbec nezměněna, aby informace o cestě paketu v lokální síti zůstaly zachovány. Tento typ přenosu má své uplatnění především při autentizaci vzdálených klientů VPN. Vzhledem k tomu, že jsou informace o cílovém zařízení dopředu známé, podporuje tento mód i nadstandartní funkce, jako je například QoS. [27]

- **Tunelovací mód:**

  Tunelovací mód na rozdíl od předchozí metody šifruje kompletně celý datový paket včetně hlavičky. Paket je pak v podstatě zabalen do nového IP paketu i s novou hlavičkou. V této podobě zůstává po celou dobu cesty až k cílovému zařízení, kde je rozbalen do původní podoby. VPN tento mód využívá především při komunikaci mezi dvěma sítěmi (host-network a host-to-host). Tato poměrně složitá režie zároveň zvyšuje nárok na výkon koncových zařízeních. [27]

IPsec využívá dva důležité protokoly, které tvoří základ této technologie. Jedná se o protokoly AH a ESP. Výsledné zapouzdření šifrovaného paketu s využitím protokolu ESP je na Obrázku 4.

Obrázek 4 – Zapouzdření IPsec paketu



Zdroj: Vlastní

4.9.1.1. Authentication Header (AH)

Protokol AH byl vytvořen v US Naval Research Laboratory na počátku 90.let dvacátého století. Je částečně odvozen z předchozích standardů IETF používaných pro autentizaci protokolu SNMP. Jako součást protokolu IPsec má na starost především

zajišťování integrity dat v hlavičce a autentizaci koncových uživatelů komunikace. Data chrání před zneužitím a nežádoucími změnami, ale před přenosem je nijak nešifruje. Zároveň umožňuje ochranu před útokem typu Replay Attack, kdy se útočník snaží zachytávat pakety a vyřadit příjemce z provozu. Protokol AH vytváří pouhý otisk dat pomocí jednosměrného hashe a algoritmů SHA1 (pro VPN typu site-to-site) nebo MD5 (pro vzdálený přístup), při čemž používá bezpečnostní kontrolní součet HMAC (Hash Message Authentication Code). [28]

4.9.1.2. Encapsulating Security Payload (ESP)

Protokol ESP byl stejně jako protokol AH vytvořen na počátku 90.let minulého století ve stejnojmenné společnosti jako součást projektu sponzorovaného společností DARPA. Protokol ESP je samozřejmě také členem sady protokolů IPsec, který poskytuje integritu dat, autentizaci uživatelů a na rozdíl od protokolu AH také navíc data šifruje. ESP podporuje konfigurace pouze pro autentizaci nebo pouze pro šifrování, ale použití šifrování bez autentizace se důrazně nedoporučuje, protože je nezabezpečené. Protokol ESP standardně šifruje data pomocí symetrické šifry DES, ale protože se tento algoritmus již nepovažuje za zcela bezpečný, využívá se častěji jeho vylepšená verze 3DES. Pro šifrování lze ovšem využít i jiné algoritmy (např. AES). [29]

4.9.1.3. Message-Digest algorithm (MD5)

MD5 je kryptografický hashovací algoritmus vytvořený v roce 1991 Ronaldem Rivestem. MD5 byl původně vytvořen jako náhrada MD4, která již tehdy neumožňovala dostatečné zabezpečení. Obecně vzato funkce hash vytvoří z řetězce znaků libovolné délky řetězec s předem určenou fixní délkou. U tohoto procesu je důležité, že sebemenší změna v datech má za následek velké a nepředvídatelné změny v hodnotě hash. MD5 tedy vezme jako vstup zprávu libovolné délky a vytvoří její 128bitový otisk (hash). Algoritmus je určen pro aplikace digitálního podpisu. Velké soubory musí být ještě před komprimací bezpečně zašifrovány soukromým klíčem!

Již v roce 1996 ovšem byly v modelu MD5 nalezeny první vady, které bohužel nebyly poslední. Od roku 2010 CMU Software Engineering Institute dokonce považuje MD5 za rozbitý a nevhodný pro další použití. I přes řadu varování a výstrah je MD5 stále široce používán. Sice už ne tolik jako kryptografická hash funkce, k čemuž byl původně vytvořen, ale spíše pro kontrolní součet ověření integrity dat (proti neúmyslnému

poškození). Také se stále používá v několika světových bezpečnostních protokolech a aplikacích, jako je například SSH, SSL, nebo právě IPSec. [30]

4.9.1.4. Secure Hash Algorithm (SHA)

SHA je další kryptografická hashovací funkce, která dokáže převést libovolně dlouhý řetězec dat na otisk fixní velikosti. SHA představuje rodinu celkem pěti protokolů, které jsou dohromady rozděleny do dvou skupin SHA-1 a SHA-2. SHA-2 pak souhrnně představuje zbylé čtyři protokoly (SHA-224, SHA-256, SHA-384, SHA-512).

SHA-1 vydal již roce 1995 americký vládní institut National Institute of Standards and Technology (NIST). Výstupem SHA-1 je 160 bitů dlouhý otisk vstupní zprávy, který se obvykle zobrazuje jako šestnáctkové číslo o délce 40 znaků. SHA-1 lze v podstatě považovat za nástupce MD5. Po roce 2004, kdy byly v MD5 nalezeny první vady, začala řada společností přecházen právě na SHA-1. Ovšem ani SHA-1 dnes nelze považovat za bezpečný. Již v roce 2005 byl zaznamenán útok na SHA-1 a od roku 2010 samotný institut NIST upřednostňuje použití protokolu SHA-2.

SHA-2 byl vydán jako oficiální standart v roce 2002. Jednotlivé protokoly SHA-2 byly pojmenovány podle výstupní bitové délky otisku zprávy (výstupem SHA-256 je 256 bitů dlouhý otisk – kód o 64 znacích). SHA-2 v roce 2015 nahradil SHA-1 v řadě bezpečnostních protokolech, jako je například SSL /TLS. SHA-2 je zatím oproti SHA-1 považován za bezpečný. Jako u většiny těchto protokolů, se ale neptáme, jestli je bezpečný, ale jak dlouho bude bezpečný. Už od roku 2012 existuje nástupce – SHA-3, který byl prvním protokolem NIST vyvinutým pomocí veřejné soutěže. [31] [32]

4.9.1.5. Security Association (SA)

K rozbalení paketů, které byly zapouzdřeny pomocí protokolu AH nebo ESP je potřeba znát tajný klíč a použitý algoritmus. Tyto informace jsou uloženy v tzv. SA (Security Association). SA definuje parametry navazovaného VPN spojení včetně přenosových pravidel. SA tedy neobsahuje jen informace o tajném klíči a používaném algoritmu, ale také informace o IP adrese autora i adresáta (IPv4 nebo IPv6), identifikátorem bezpečnostního protokolu (AH nebo ESP) a také tzv. SPI (Security parameter index), což je jednoznačná identifikace SA. Jinými slovy, SA lze považovat za skupinu zabezpečujících parametrů, která umožňuje sdílet informace s jinou entitou. Dále využívá protokoly jako například ISAKMP, který poskytuje rámec pro vytváření SA, nebo protokol IKE, jenž poskytuje informace o autentizaci a klíčích. [33]

4.9.1.6. Protokol Internet Key Exchange (IKE)

Výměna tajných klíčů mezi účastníky je možná dvěma způsoby. Buď poměrně nepohodlnou ruční konfigurací nebo s využitím protokolu IKE (Internet Key Exchange). Tento protokol je součástí sady protokolů IPsec a zajišťuje především správu klíče včetně jeho distribuce. Jeho úkolem je tedy poskytovat bezpečné přeposlání tajného klíče, mezi uživateli po nezabezpečené síti. V současné době existuje IKE ve dvou verzích – IKEv1 a IKEv2. Vylepšená verze IKEv2 má několik rozšíření a výhod ovšem funkce zůstává stejná.

IKEv1 pracuje ve dvou fázích. V první fázi dojde k vytvoření zabezpečeného komunikačního kanálu mezi dvěma účastníky, kterému se všeobecně říká IKE SA. K tomu využívá Diffie-Hellmanova protokol, jenž má za úkol vytvoření a přenesení symetrického šifrovacího klíče, který je následně použit pro šifrování zbytku komunikace. První fáze může navíc pracovat v jednom ze dvou módů – Main Mode (Hlavní mód) a Aggressive Mode (Agresivní mód). Hlavní mód využívá šesticestný handshake, parametry jsou zde vyměňovány ve více kolech a jsou vždy zašifrovány. Naopak Agresivní mód využívá třícestný handshake, kdy zahashované parametry odesílá v jedné nezašifrované zprávě. Tato metoda je obvykle používaná pro VPN se vzdáleným přístupem, nebo v situacích, kdy oba uživatelé mají dynamické externí IP adresy. Fáze dva (někdy se nazývá Quick Mode) využívá již vzniklý klíč a komunikační kanál z fáze jedna pro ustanovení SA výše zmíněných protokolů AH a ESP. Očekává se, že veškerá komunikace v tomto okamžiku bude zabezpečena na základě ověřování, ke kterému došlo v první fázi.

IKEv2 na rozdíl od první verze neprobíhá ve fázích, ale metodou dvojitých zpráv – požadavek / odpověď. Vždy čeká na dokončení předchozí relace těchto dvojitých zpráv a pak řeší další. V prvních dvou zprávách se domluví na IKE SA, a navíc také na Child SA, což je první bezpečnostní asociace pro konkrétní protokol. [34]

4.10. Layer 2 Tunel Protocol (L2TP)

Poslední a zároveň zde nejmodernější VPN protokol je Layer 2 Tunel Protocol – L2TP. Protokol, jak už název napovídá, vytváří tunel na druhé vrstvě modelu ISO/OSI. L2TP vznikl vzájemnou spoluprací společnosti Microsoft a Cisco Systems, a tudíž základ této technologie tvoří dva jejich protokoly. Jedná se o specifikaci již zmiňovaného PPTP od Microsoftu a protokolu L2F (Layer 2 Forwarding) od Cisco Systems. Díky tomu je podpora Windows od verze 2000 samozřejmostí. [35]

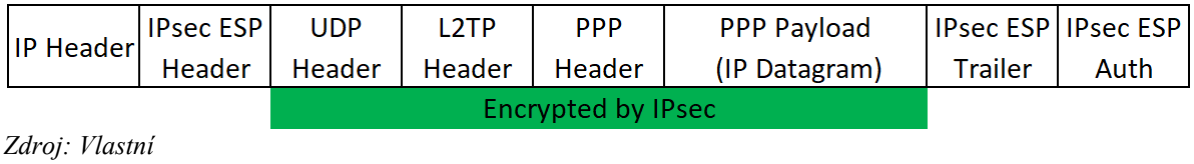
4.10.1. L2TP – Princip

Z teoretického hlediska L2TP představuje jakousi kombinaci obou předchozích VPN protokolů. Obdobně jako u PPTP je zahájeno spojení s přístupovým serverem pomocí PPP v kombinaci s L2F protokolem. Pokud je ověření úspěšné, je navázáno VPN spojení s cílovým zařízením s komunikační relací L2TP. L2TP ovšem následnou komunikaci nijak nešifruje. V šifrování totiž spoléhá na IPsec, který je s L2TP typicky implementován. Této kombinaci se běžně říká L2TP/IPsec (tunelování tedy probíhá na druhé vrstvě a šifrování až na třetí). Výměna SA informací opět probíhá pomocí protokolu IKE. Pro vytvoření tunelu a následné řízení přenosu používá UDP spojení na portu 1701. Defaultní nastavení L2TP/IPsec je následovné:

- Transport Mode
- Encapsulating Security Payload (ESP)
- Triple Data Encryption Standard (3DES)
- Secure Hash Algorithm (SHA-1)
- Diffie-hellman Medium (DH-2)

Výsledné zapouzdření paketu vypadá následovně. Před PPP rámec je vložena L2TP a UDP hlavička. Takto zapouzdřený paket je posléze zašifrován pomocí protokolu IPsec ESP a jeho hlavička je vložena na začátek paketu. V posledním kroku je paket zapouzdřen v doručovacím IP protokolu. Na Obrázku 5 je zobrazena finální podoba zapouzdřeného paketu. [36]

Obrázek 5 – Zapouzdření L2TP paketu – IPsec šifrování



Zdroj: Vlastní

4.10.1.1. Layer 2 Forwarding (L2F)

Jedná se o další tunelovací protokol, tentokrát od společnosti Cisco Systems. L2F vytváří zabezpečený tunel napříč veřejnou infrastrukturou. L2F byl speciálně navržen pro tunelování protokolu PPP. Protokol tedy vytváří síťová a uživatelská připojení point–to–point a umožňuje protokolům na vyšší úrovni vytvářet tunely prostřednictvím síťové vrstvy. [26]

4.11. Subnetting

Když se poprvé začaly přidělovat IP adresy vznikla tehdy prvotní myšlenka, že IP adresy budou přidělovány poměrně nešetrně po celých „blocích“. Tehdy nikdo nemohl očekávat tak dynamický rozvoj internetu do takového rozsahu, jak ho známe dnes. To znamenalo, že úspěšnému žadateli o IP adresy byl přidělen nejbližší vyšší počet IP adres. Takže, když si podnikatel zažádal o 400 IP adres, dostal automaticky IP adresy třídy B, což znamenalo 65 536 IP adres. Samozřejmě netrvalo dlouho a tímto způsobem IP adresy začaly nekompromisně docházet. Další myšlenkou ve snaze šetření IP adresami bylo přidělit žadateli více menších bloků. Takže by tento podnikatel tentokrát dostal dva bloky IP adres třídy C, tedy 2x 255 IP adres. Ani tato metoda už tehdy nebyla dostačující, a tak se začalo nasazovat několik úsporných řešení, které se používají až do dnes. [9]

Jednou z prvotních technologii bylo dynamické přidělování IP adres pomocí DHCP serveru. Došlo sice k úspoře, ale i tak se hledaly jiné, lepší způsoby. Jedním z dalších řešení bylo zavedení právě tzv. subnettingu, který dělil bloky adres na více částí pro více sítí. Další variantou byl CIDR, který umožnoval ještě podrobnější dělení síťové a klientské části.

Tyto metody byly později v této problematice zastíněny technologií vnějších a vnitřních (resp. privátních a veřejných) sítí, které se společně s technologií NAT využívají až do dnes. V dnešní době je i tato metoda nedostačující a jediné, co může pomoci je zvětšení adresního prostoru čili přechod na IPv6 (zhruba $3{,}4\times10^{38}$ volných IP adres). [37]

Subnetting byl první metodou, jak rozdělovat síť na podsítě. Jeho největší výhodou byla jednoduchost a fakt, že rozdělení probíhalo pouze softwarově za pomoci vhodně nastavené masky sítě a IP adres jednotlivých zařízení. K rozdělení není třeba žádný další aktivní prvek. Naopak největší nevýhodou subnettingu je, že dokáže rozdělit síť jen na omezený počet podsítí. Konkrétně jen násobky dvou. To znamená, že IP adresu třídy C dokáže rozdělit jen na 2, 4, 8, 16, 32, 64 podsítí. [9]

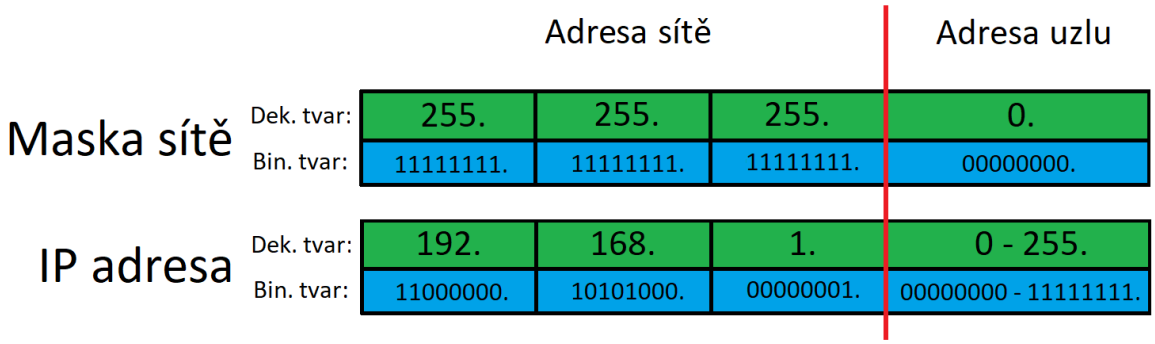
4.11.1. Subnetting – Princip

Jak je zmíněno výše, IP adresa je rozdělena do čtyř částí, oddělených tečkou, po osmi bitech. Tyto čtyři části IP adresy jsou pomocí třídy adres pevně rozděleny na dvě části, a sice na adresu sítě a adresu uzlu. Subnetting pomocí masky sítě umožňuje změnit kolik bitů v rámci jedné části IP adresy bude přiděleno pro adresu sítě a kolik pro adresu uzlu. Jelikož je samozřejmě počet bitů v IP adrese neměnný, tak z logiky věci vyplývá, že pokud bude odebráno několik bitů, které byly původně přiděleny adrese uzlu, a budou použity pro adresu

podsítě, tak se počet bitů určený pro adresy uzlů sníží. Jinak řečeno, zvětšuje se adresa sítě na úkor adresy uzlu (čím více bude podsítí, tím méně síťových rozhraní v nich může být zapojeno). Jelikož je subnetting vázán rozdělováním sítě po celých blocích (8 bitech), lze síť rozdělovat velice omezeně. IP adresu třídy C, kde je pro adresu uzlů k dispozici posledních 8 bitů (254 dostupných IP adres pro síťová rozhraní v jedné síti), je možno rozdělit pouze na 2, 4, 8, 16, 32, 64 podsítí. Protože každý bit masky sítě, přísluší stejnému bitu v IP adrese, a právě maska sítě definuje rozdělení IP adresy na část síťovou a část uzlovou. Na pozici, kde jsou jedničky je část síťová a na zbylých pozicích, kde jsou nuly, je část uzlová. [9]

Viz Obrázek 6, kde síť není rozdělena na podsítě. Maska sítě je tedy 255.255.255.0 a IP adresy pro jednotlivá zařízení mohou být v rozsahu 192.168.1.0-255. S takto nastavenou sítí se lze zcela běžně setkat v domácnostech.

Obrázek 6 – Subnetting – Maska sítě a IP adresa



Zdroj: Vlastní

4.11.1.1. Rozdělení sítě na podsítě pomocí subnettingu

Prvním krokem při rozdělení sítě na podsítě je změna masky sítě. Pomocí vzorce $[2^N = počet\ podsítí]$ je možné dopočítat počet bitů, které budou přiděleny z uzlové části do síťové. V tomto vzorci v levé části konstanta čísla 2 představuje počet možných stavů, které mohou nastat na dané pozici (hodnota 1 a 0) a mocnina N je proměnná, kterou je číslo umocněno, tak aby výsledek byl počet potřebných podsítí. Toto číslo N je shodné s potřebným počtem bitů. Tyto bity představují v masce sítě jedničky od nejvyšší bitové pozice po nejnižší. Zbytek masky sítě je potom samozřejmě doplněn nulami. Maska sítě je spočítána jako převod tohoto binárního čísla do dekadické soustavy.

Bity, které se v IP adrese shodují s jedničkovými bity masky sítě (jenž v podstatě definují počet nově vzniklých podsítí) mohou nabývat všech možných kombinací 1 a 0. Počet kombinací jedniček a nul exponenciálně vzrůstá s počtem používaných bitů

pro síťovou část. Aby bylo jednoznačně určeno, jaké zařízení bude zapojeno, do které podsítě, musí mít statickou IP adresu z rozsahu IP adres dané podsítě.

Rozsah IP adres v jednotlivých podsítích je dán převodem adresy podsítě a zároveň uzlové části z binární soustavy do dekadické, kde jsou bity, které tvoří síťovou část doplněny o nuly (nejnižší možná IP adresa v podsíti) a pak o jedničky (nejvyšší možná IP adresa v podsíti). Od maximálního možného počtu IP adres je nutnost vždy odečíst dvě, protože nultá (00000000) a poslední (11111111) IP adresa se nikdy nepoužívá pro vlastní účely! To platí pro každou podsíť. V Tabulce 1 jsou zobrazeny dostupné IP adresy třídy C v závislosti na nastavené masce sítě.

Tabulka 1 – Subnetting – Počet dostupných IP adres pro danou podsíť

Počet podsítí	Maska sítě	Počet dostupných IP adres
1	255.255.255.0	254
2	255.255.255.127	126
4	255.255.255.192	62
8	255.255.255.224	30
16	255.255.255.240	14
32	255.255.255.248	6
64	255.255.255.252	2

Zdroj: Vlastní

4.11.1.2. Příklad implementace subnettingu

Zadání:

Je dána IP adresa třídy C: 192.168.1.X, kterou je třeba rozdělit na 4 podsítě.

Maska sítě:

Počet bitů pro síťovou a uzlovou část je spočítán pomocí výše zmiňovaného vzorce:

$[2^N = \textit{počet podsítí}] = [2^{\mathbf{2}}=\mathbf{4}]$

Pro síťovou část jsou tedy zapotřebí 2 bity a pro uzlovou část zbude 6 bitů:

$[\mathbf{11}000000_{BIN} = 192_{10}]$

Maska sítě je tedy stanovena na 255.255.255.192 jak je zobrazeno na Obrázku 7.

Obrázek 7 – Subnetting – Maska sítě pro rozdělení na 4 podsítě

Zdroj: Vlastní

Rozsah IP adres:

První dva bity IP adresy, které se shodují s prvními dvěma bity masky sítě určují adresu podsítě. Všechny možné stavy, které mohou nastat kombinací 1 a 0 v těchto dvou bitech určují adresy jednotlivých podsítí. Adresa první podsítě je tedy [00], druhé [01], třetí [10] a čtvrté [11]. Rozsah IP adres pro danou podsíť se vypočítá převodem do dekadické soustavy, kde je zbytek bitů doplněn o nuly pro nejnižší IP adresu a o jedničky pro nejvyšší IP adresu. Jak je uvedeno v Tabulce 2.

Tabulka 2 – Subnetting – Rozsah IP adres pro 4 podsítě

Maska sítě:	255.255.255.	11	000000		Rozsah IP adres pro podsítě
IP adresa:	192.168.1.	00	000000 - 111111	=	192.168.1.0 - 192.168.1.63
		01	000000 - 111111	=	192.168.1.64 - 192.168.1.127
		10	000000 - 111111	=	192.168.1.128 - 192.168.1.191
		11	000000 - 111111	=	192.168.1.192 - 192.168.1.255

Zdroj: Vlastní

V každé podsítí není možné využít nejnižší a nejvyšší IP adresu! Tyto dvě adresy jsou určeny pro adresu podsítě a broadcoast. Zařízení, která budou zapojena do první podsítě musí mít staticky přidělenou IP adresu z rozsahu 192.168.1.1 až 192.168.1.62, v druhé podsítí 192.168.1.65 až 192.168.1.126, ve třetí 192.168.1.129 až 192.168.1.190 a ve čtvrté 192.168.1.193 až 192.168.1.254.

4.12. Classless Inter–Domain Routing (CIDR)

S metodou subnetting se velice často spojuje i metoda CIDR. CIDR vznikal právě na základě subnettingu s cílem eliminovat jeho největší nevýhodu, a to sice značně omezené rozdělování podsítí. CIDR tedy stejně jako subnetting umožnuje softwarové rozdělení sítě na podsítě, ale daleko podrobněji. Při využití CIDRu v podstatě úplně zaniká význam tříd IP adres. CIDR už totiž není vázán na jednotlivé bloky IP adresy. Je ovšem podstatně náročnější na výpočet než subnetting, a právě proto se mu v praxi dává přednost pouze, když je podrobnější rozdělení sítě (než na 2, 4, 8, 16, 32 atd. podsítí) nevyhnutelné. [9] [38]

Za adresu sítě se přidává lomítko se zápisem buďto celé masky sítě, nebo jednoho čísla (tzv. CIDR Notation), avšak oba zápisy jsou totožné. CIDR Notation prostě a jednoduše určuje počet bitů v masce sítě, které jsou od nejvyšší pozice jedničky (zbytek jsou nuly). Zápis IP adresy 192.168.1.1/28 je totožný jako 192.168.1.1/255.255.255.240. Protože maska sítě je v binárního tvaru: 1111111.1111111.1111111.11110000. Poslední jednička je zleva přesně 28. bit a číslo $11110000_2 = 240_{10}$. CIDR Notation oddělený za IP adresou lomítkem tedy určuje, do jaké podsítě daná IP adresa připadá. [39]

CIDR v kombinaci s adresou sítě se často také používá pro definování IP rozsahu. IP rozsah využívají routery při specifických operacích, kdy potřebují určit možný rozsah IP adres, který například mohou přidělit konkrétním zařízením. Například IP adresa sítě 192.168.1.0/26 ($26_2 = 192_{10}$) definuje rozsah IP adres od 192.168.1.0 až 192.168.1.63. V takovémto případě je pak IP adresa další sítě 192.168.1.64 a v kombinaci se CIDRem „26“ definuje rozsah IP adres od 192.168.1.64 až 192.168.127.

5. Praktická část

V praktické části práce je čtenář nejprve seznámen se všemi komponenty a programy, které jsou využity při konfiguraci a měření. Následně je vysvětlena konfigurace LAN sítě, konfigurace routerů ER6020v1 a ER6020v2 a popsány všechny prováděné testy a měření. Každá konfigurace VPN serveru na routeru ER6020v1 je v následujících kapitolách detailně zdokumentována. U každé konfigurace jsou zároveň doloženy všechny výsledky prováděných testů. V předposlední kapitole jsou zhodnoceny naměřené výsledky a závěrem práce jsou diskutovány slabé a silné stránky jednotlivých VPN protokolů.

5.1. Využité komponenty a programy

Konfigurace VPN serveru probíhá ve školní laboratoři Technické fakulty České zemědělské univerzity. V laboratoři jsou k dispozici dvě totožné počítačové stanice („A“ a „B“), které jsou pro implementaci a testování využity. VPN server je realizován na routeru ER6020v1 od společnosti TP-Link. Pomocí služby VPN je k tomuto routeru připojena další počítačová stanice („C“), která se nachází v jiné vzdálené síti. Tato počítačová stanice je fyzicky umístěna ve městě Kladno (vzdušnou čarou necelých 18 km od Technické fakulty ČZU).

K měření přenosové rychlosti a ztrátovosti paketů mezi vzdálenými počítačovými stanicemi je využita freewarová aplikace TamoSoft Throughput Test. K měření latence uvnitř sítě je použit příkazový řádek (CMD) a funkce Ping. Pro změření hodnot ping (latence), jitter, download a upload do Internetu je využit webový portál „speedtest.cesnet.cz/“. Zachycení jednotlivých paketů po dobu přenosu probíhá pomocí programu Wireshark (záznamy jsou doloženy v příloze). Pro realizaci IPsecu na Windows platformě je navíc využit klientský software TheGreenBow VPN. V ostatních případech probíhá přihlášení přes Windows rozhraní.

- **Latence/Ping/Odezva:**

  Vyjadřuje časovou prodlevu (zpoždění) mezi požadavkem na provedení akce a okamžikem, kdy je požadavek vyřízen. Rychlost odezvy se měří v milisekundách. Samozřejmě čím je hodnota nižší, tím je komunikace rychlejší.

- **Jitter:**

  Představuje kolísání velikosti zpoždění paketů při průchodu sítí. Tento parametr pomáhá odhalit, jak je internetové připojení stabilní. Rychlost se měří také v milisekundách. Čím je hodnota jitter nižší, tím je stabilita lepší.

- **Download:**

  Široce známý pojem, který v počítačových sítích představuje rychlost příjmu dat stažených ze vzdáleného zařízení, systému nebo serveru (například z webového serveru, FTP serveru, e-mailového serveru apod.). Nejčastěji se udává v jednotkách Mbps (Megabits per second), což jak název napovídá, je počet megabitů stažených za vteřinu.

- **Upload:**

  Upload představuje pravý opak downloadu. Definuje množství odeslaných dat ze zařízení na jiné vzdálené zařízení, systém nebo server. Stejně jako download se udává v jednotkách Mbps.

5.1.1. Počítačové stanice

K měření rychlosti uploadu a downloadu je využit webový portál „speedtest.cesnet.cz/“. Test je spuštěn celkem 10x. Výsledky z počítačových stanic technické fakulty („A“) jsou zapsány v Tabulce 3 a výsledky ze vzdálené počítačové stanice („C“) jsou v Tabulce 4.

Počítačové stanice na technické fakultě ČZU („A“ a „B“):

- Operační systém: Windows 10 Enterprise (verze 1803)
- Typ systému: 64bitový operační systém
- Procesor: Intel(R) Core(TM) i5-8400 CPU @ 2.80GHz 2.81GHz
- Operační paměť: 16 GB
- Grafická karta: NVIDIA Quadro P620
- ID produktu: 00329–10180–22135–AA445
- Internet provider: T-Mobile

Tabulka 3 – PC:A – CESNET – Ping, Jitter, Download, Upload

	Ping [ms]	Jitter [ms]	Download [Mbps]	Upload [Mbps]
1.	4,66	1,37	93,01	85,62
2.	6,31	3,42	99,20	87,23
3.	4,00	1,02	94,35	86,50
4.	4,64	1,65	95,81	84,37
5.	3,72	0,29	93,80	84,43
6.	3,94	8,04	96,93	81,34
7.	3,89	3,68	94,47	85,27
8.	4,49	1,43	94,20	83,68
9.	4,23	0,24	93,32	86,85
10.	4,04	1,56	93,06	81,85
Průměr:	**4,39**	**2,27**	**94,82**	**84,71**

Zdroj: Vlastní

Vzdálená počítačová stanice („C“):

- Operační systém: Windows 10 Home (verze 2004)
- Typ systému: 64bitový operační systém
- Procesor: Intel (R) Core(TM) i5-4460 CPU @ 3.20 GHz 3.20 GHz
- Operační paměť: 16 GB
- Grafická karta: NVIDIA GeForce GTX 1060 3 GB
- ID produktu: 00326–10000–00000–AA521
- Internet provider: UPC internet

Tabulka 4 – PC:C – CESNET – Ping, Jitter, Download, Upload

	Ping [ms]	Jitter [ms]	Download [Mbps]	Upload [Mbps]
1.	20,25	5,28	297,92	20,05
2.	20,23	2,28	295,64	20,00
3.	20,43	2,92	292,23	19,98
4.	20,97	4,72	278,82	20,00
5.	19,70	3,44	301,12	21,10
6.	20,65	1,93	299,80	20,17
7.	19,61	2,22	289,60	20,92
8.	21,02	15,98	299,17	19,95
9.	21,17	8,78	288,89	20,13
10.	20,75	6,81	293,34	20,03
11.	21,24	8,59	292,26	20,17
Průměr:	**20,55**	**5,72**	**293,53**	**20,23**

Zdroj: Vlastní

5.1.2. Router TL-ER6020

K nastavení VPN serveru je využit router ER6020 (Gigabitový router SafeStream™ Dual-WAN VPN) od společnosti TP-Link s 64bitovým dvoujádrovým procesorem. Router podporuje všechny tři profilové VPN protokoly (PPTP, IPsec, L2TP). Možné parametry

nastavení těchto VPN jsou v Tabulce 5. Konfigurace routeru je možná po přihlášení do uživatelského prostředí ve webovém browseru. Do URL adresy je třeba zadat lokální adresu (default gateway) routeru. Tu lze nejsnáze dohledat přes příkazový řádek pomocí příkazu „*ipconfig*“.

Tabulka 5 – Router ER6020v1 – parametry VPN

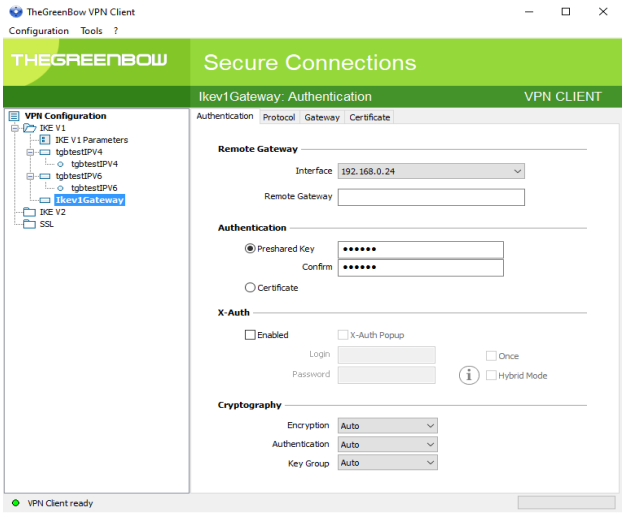
VPN Protokol	Parametry
PPTP	podporuje až 16 PPTP VPN tunelů zároveň PPTP VPN Server/Client PPTP šfirování s MPPE
IPsec	podporuje až 64 IPsec VPN tunelů zároveň LAN-to-LAN, Client-to-LAN Main, Aggressive Negotiation Mode Šifrovací algoritmy: DES, 3DES, AES128, AES192, AES256 Autentizační protokoly: MD5, SHA1 IPsec NAT Traversal (NAT-T) Dead Peer Detection (DPD) Perfect Forward Secrecy (PFS)
L2TP	podporuje až 16 L2TP VPN tunelů zároveň L2TP VPN Server/Client L2TP šifrování s IPsec

Zdroj: https://www.tp-link.com/cz/business-networking/vpn-router/tl-er6020/

5.1.3. TheGreenBow VPN

Společnost TheGreenBow poskytuje od roku 1998 nejrůznější řešení VPN. Jejich hlavní produkt je klientský software VPN (s totožným jménem TheGreenBow VPN). Software je schopen pracovat na všech médiích (WiFi, Ethernet, ADSL, 3G, GPRS, satelit atd.) a podporuje hned několik VPN protokolů. TheGreenBow není freewarový software. K jeho použití je třeba předplacená licence. V této práci je proto využita pouze jeho 30denní trial verze. Uživatelské rozhraní aplikace je zobrazeno na Obrázku 8. Pro zvýšení bezpečnosti (například ve firemním prostředí) aplikace umožňuje konfiguraci VPN tunelu zabezpečit heslem.

Obrázek 8 – TheGreenBow VPN – Uživatelské rozhraní



Zdroj: Vlastní

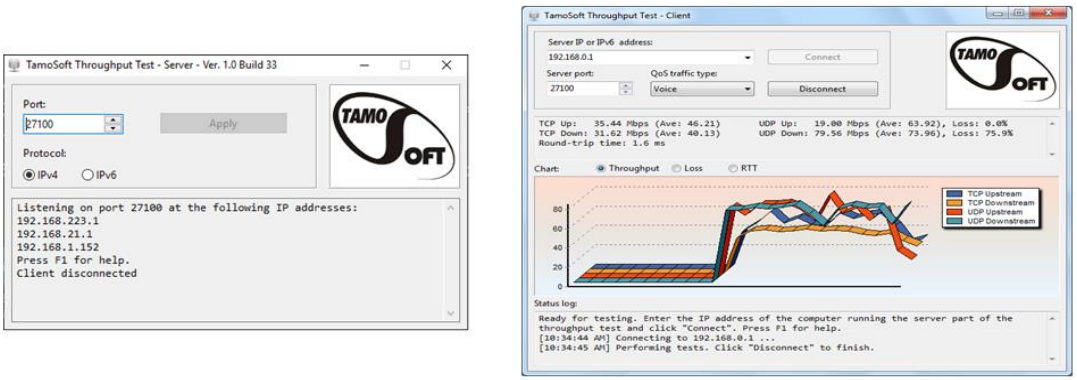
5.1.4. TamoSoft Throughput Test

TamoSoft Throughput Test je freewarový software pro testování výkonu kabelové i bezdrátové sítě od společnosti TamoSoft. Program průběžně odesílá TCP a UDP datové toky v rámci lokální sítě a měří přenosovou rychlost včetně ztrátovosti UDP paketů. Tyto výsledky jsou zobrazeny v numerických i grafických formátech.

Program má dvě části – server a klient. Na obou koncových zařízeních musí být spuštěna právě jedna z nich. Serverové a klientské rozhraní je na Obrázku 9. Počítač, na kterém je spuštěn klient se připojí k počítači se serverem. Po navázání spojení odesílají klient i server data v obou směrech. Klientská část aplikace pak vypočítává a zobrazuje výsledky.

Konfigurace je velice snadná. Na klientské části stačí zadat IP adresu počítače se serverem a na obou počítačových stanicích se musí v aplikaci shodovat server port.

Obrázek 9 – TamoSoft Throughput Test – Server (vlevo) / Klient (vpravo)

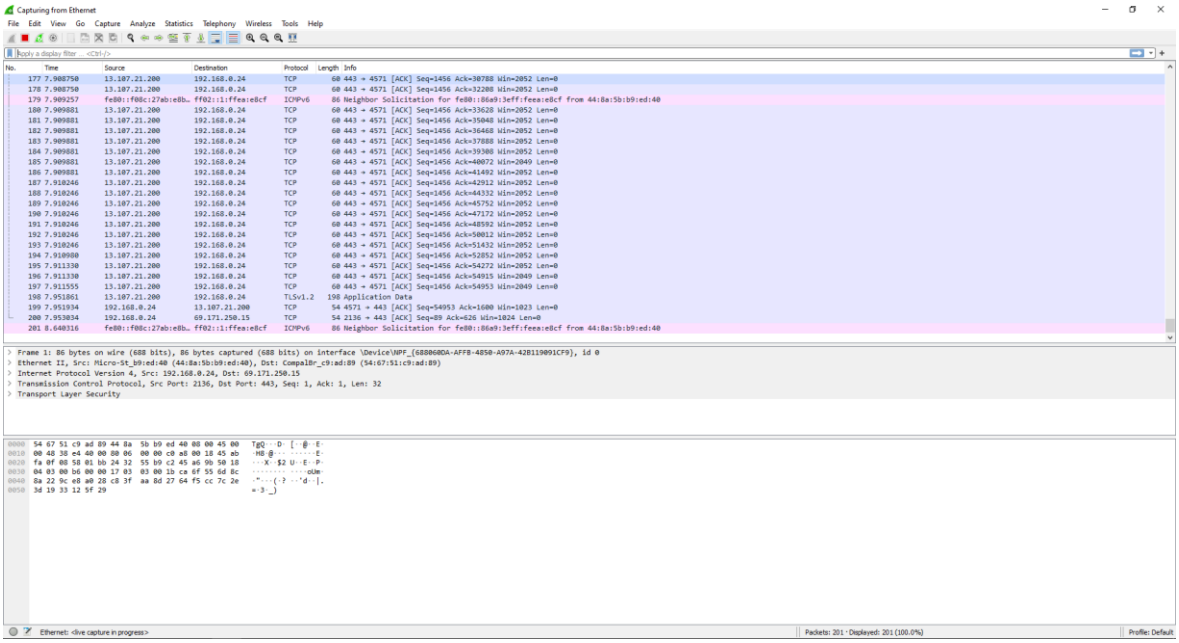


Zdroj: https://www.tamos.com/products/throughput-test/

5.1.5. Wireshark

Program Wireshark od stejnojmenné společnosti slouží jako analyzátor síťových protokolů. Umožňuje velice detailně zachytit co se děje v kabelové i bezdrátové síti v průběhu přenosu. Zpracovaná data lze procházet pomocí grafického uživatelského rozhraní nebo pomocí terminálové verze utility TShark. Jak je vidět na Obrázku 10, Wireshark přehledně zobrazuje veškeré přijaté a odeslané pakety (včetně oddělení jednotlivých protokolů). V aplikaci je dostupný i filtr pro usnadnění vyhledávání konkrétních dat. Zároveň je možné data vyexportovat v souborech XML, CSV nebo prostém textu. Wireshark používá pro zpracování paketů knihovnu zvanou „pcap“, takže zachytí pouze pakety, které tato knihovna podporuje.

Obrázek 10 – Wireshark – Uživatelské rozhraní



Zdroj: Vlastní

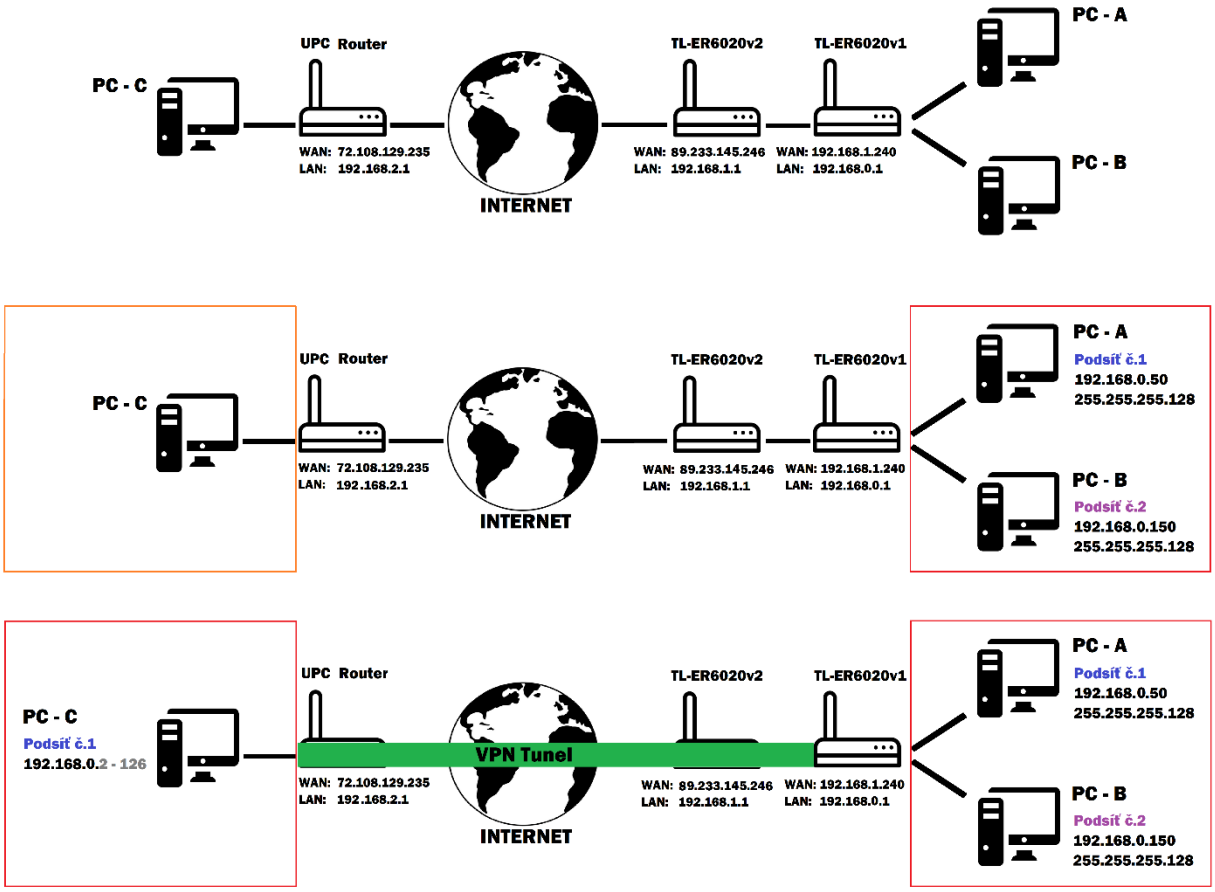
5.2. Konfigurace LAN

Pro konfiguraci a měření celého zapojení je vytvořena pomocí routeru ER6020v1 malá lokální síť o dvou počítačových stanicích („A“ a „B“). Před routerem ER6020v1 se nachází ještě jeden hlavní router ER6020v2, který má přístup k veřejné síti. Počítačové stanice uvnitř lokální sítě routeru ER6020v1 jsou rozděleny do dvou menších podsítí. Počítačová stanice „A“ je v podsíti číslo 1 a počítačová stanice „B“ je v podsíti číslo 2. Síť je do podsítí rozdělena metodou subnetting (tzn. pouze softwarově pomocí vhodně nastavené statické IP adresy a masky sítě). Stanice „A“ může mít přidělenou IP adresu z rozsahu

192.168.1.2 až 192.168.1.126 a stanice „B“ z rozsahu 192.168.1.129 až 192.168.1.254. V této práci je počítačové stanici „A“ přidělena statická IP adresa 192.168.1.50 a počítačové stanici „B“ IP adresa 191.168.1.150. Maska sítě je u obou počítačových stanic totožná 255.255.255.128.

Na routeru je následně spuštěn VPN server. Konfigurace PPTP, IPsec a L2TP je zdokumentována v samostatných kapitolách níže. K této lokální síti routeru ER6020v1 je pomocí služby VPN připojena třetí počítačová stanice „C“, která se implicitně nachází v oddělené vzdálené síti. Tato počítačová stanice se nachází v podsíti číslo 1 a musí jí být tedy přidělena IP adresa z prvního rozsahu. Počítačová stanice „C“ má tedy v rámci této sítě přístup k počítačové stanici „A“ a naopak nemá přístup k počítačové stanici „B“. Správnost konfigurace je u každého zapojení ověřena pomocí funkce Ping. Předpokládá se, že z počítačové stanice „C“ bude úspěšně volána stanice „A“ a neúspěšně stanice „B“. Celé schéma zapojení je demonstrováno na následujícím Obrázku 11.

Obrázek 11 – Schéma zapojení



Zdroj: Vlastní

5.3. Konfigurace routeru ER6020v2

Realizace VPN serveru probíhá na routeru ER6020v1. V topologii sítě laboratoře na technické fakultě ČZU se ale před routerem ER6020v1 nachází ještě jeden hlavní router laboratoře ER6020v2 (další zařízení NAT).

Aby k VPN serveru na routeru ER6020v1 byl možný přístup z veřejné sítě je na výše postaveném routeru ER6020v2 nastavena služba Virtual Servers. Pro jednotlivá čísla portů VPN protokolů je touto službou nastaveno přesměrování na IP adresu 192.168.1.240. Pod touto IP adresou se skrývá právě router ER6020v1 – VPN server. Toto nastavení je možné na routeru ER6020v2 v záložce „Transmission => NAT => Virtual Servers“. Níže zapsané porty jsou přesměrovány na WAN adresu VPN serveru. Kdyby k tomuto nastavení nedošlo, tak by router ER6020v2 veškerou tuto komunikaci na těchto portech blokoval a přístup ze vzdálené sítě by k VPN serveru nebyl možný.

- PPTP port: 1723 TCP port
- L2TP port: 1701 UDP port
- IPsec port č.1: 500 UDP port
- IPsec port č.2 4500 UDP port

Pro následnou konfiguraci VPN je třeba dohledat i WAN IP adresu hlavního routeru laboratoře ER6020v2 přes kterou probíhá veškerá komunikace do veřejné sítě. Tu lze dohledat na jakékoliv počítačové stanici umístěné za tímto routerem. Ve webovém prohlížeči stačí do URL adresy zadat odkaz: „www.whatismyip.cz/“.

- WAN IP adresa routeru ER6020v2: 89.233.145.246

5.4. Konfigurace routeru ER6020v1

Vzhledem k nastavení na výše postaveném routeru je zde v záložce „Network => WAN1“ nakonfigurována WAN IP adresa podle následujících parametrů:

- IP address: *192.168.1.240*
- Subnet Mask: *255.255.255.0*
- Default Gateway: *192.168.1.1*
- Primary DNS: *8.8.8.8*

IP adress je tedy nastavena na 192.168.1.240 (adresa, na kterou se z routeru ER6020v2 přesměrovává komunikace na vybraných portech), Subnet Mask je nastaven jako 255.255.255.0, Default Gateway jako 192.168.1.1 (vnitřní brána routeru ER6020v2) a DNS je nastaven na adresu Googlu 8.8.8.8. Zbytek nastavení je ponecháno v předdefinovaných stavech.

Pro vytvoření VPN serveru je dále nutné přednastavit tzv. IP pool, který bude potřebný při pozdější konfiguraci PPTP a L2TP. Zařízení, které se připojí k tomuto VPN serveru je pak možné přidělovat IP adresy právě z tohoto připraveného rozsahu. Jinými slovy, vytvořený virtuální VPN adaptér na vzdáleném zařízení bude mít vždy přidělenou IP adresu právě z tohoto rozsahu. Je tedy důležité, aby byl IP pool nastaven v rozsahu první podsítě. Počítačová stanice „C“ má potom přístup k počítačové stanici „A“ a naopak nemá přístup k počítačové stanici „B“. IP pool je možný nastavit v záložce „VPN => PPTP/L2TP => IP Address Pool“. V rámci této práce je na routeru vytvořen IP pool s tímto rozsahem adres:

- IP pool name: *VPN pool*
- Starting IP adress: *192.168.0.20*
- Ending IP adress: *192.168.0.40*

Další konfigurace každého VPN protokolu je odlišná. Proto je další postup uveden v samostatných kapitolách PPTP konfigurace a testování, IPsec konfigurace a testování a L2TP konfigurace a testování.

5.5. Popis prováděných testů

Důležité je zdůraznit, že každý test přes veřejnou síť je proměnlivý a zkreslený současným zatížením sítě, které bohužel nelze nijak ovlivnit nebo stabilizovat. Zároveň jsou testy závislé na výchozích parametrech sítě (ping, jiter, download, upload) dané konkrétním providerem. Pro co možná nejobsáhlejší výsledky je každý test pro PPTP, IPsec a L2TP měřen ve dvou dnech. Prvních pět testů PPTP, IPsec a L2TP přes TamoSoft Throughput Test a CMD je měřeno v jeden den a druhý den je měřeno dalších pět testů. Na stejném principu je měřen i test do veřejné sítě pomocí webového portálu „speedtest.cesnet.cz/“.

Test mezi počítačovou stanicí „A“ a „C“: U každého testovaného VPN protokolu je mezi počítačovou stanicí („A“) umístěnou fyzicky uvnitř lokální sítě routeru a vzdálenou počítačovou stanicí („C“) připojenou pomocí VPN proveden test přenosové rychlosti TCP

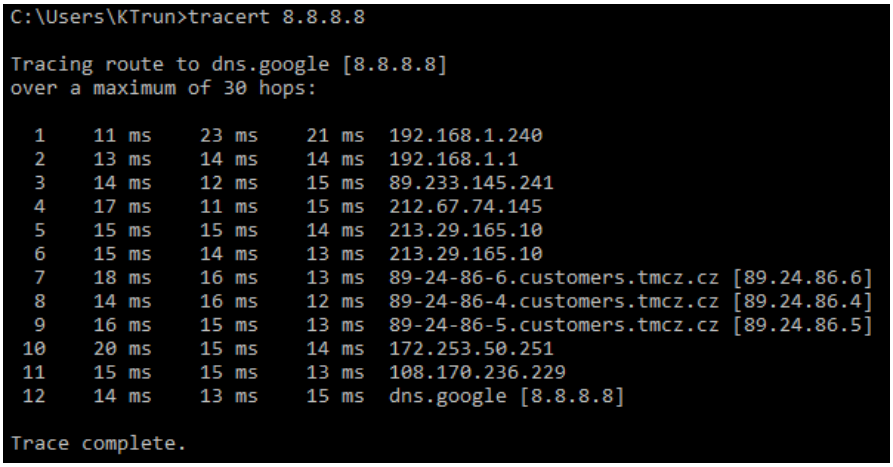
a UDP paketů, ztrátovosti UDP paketů a latence pomocí programu TamoSoft Throughput Test a příkazového řádku (CMD).

Test z počítačové stanice „C“ do Internetu (pouze PPTP a L2TP): VPN adaptér na vzdálené počítačové stanici je při využití protokolu PPTP a L2TP vždy nastaven tak, aby přístup do Internetu byl přes vzdálenou bránu routeru ER6020v1. Router ER6020v1 v případě IPsecu toto nastavení bohužel neumožňuje, a tak z tohoto testu musí být vynechán. U PPTP a L2TP datové pakety odeslané z počítačové stanice „C“ musí první dorazit skrz VPN tunel k routeru ER6020v1, který posléze odešle tyto datové pakety na svou cestu do Internetu. Správnost tohoto nastavení lze ověřit pomocí příkazového řádku a příkazu „*tracert 8.8.8.8*“ – veřejná IP adresa Googlu umístěná ve veřejné síti. Při správné konfiguraci lze pozorovat, jak datové pakety cestují nejprve k routeru ER6020v1 (192.168.1.240) a posléze přes vnitřní bránu routeru ER6020v2 (192.168.1.1) pokračují do Internetu. Viz Obrázek 12, kde je zobrazen záznam z příkazového řádku. Samozřejmě, z logiky věci vyplývá, že zpomalení, ke kterému dojde při zašifrování a odeslání paketů skrz VPN tunel se projeví na všech testovaných hodnotách (ping, jitter, download a upload). Výsledky tohoto testu jsou porovnány s naměřenými výsledky stejného testu, který probíhal na počítačových stanicích, které jsou fyzicky zapojeny za routerem ER6020v1 („A“ a „B“).

Záznam přenášených paketů: V závěru každé kapitoly VPN protokolu je v příloze doložen záznam přenosu zachycený programem Wireshark na počítačové stanici „C“. Záznam přenosu dokumentuje připojení a po 5 vteřinách odpojení konkrétní VPN. Vzhledem k množství odeslaných paketů (i za těchto pár vteřin) je ze všech seznamů vždy smazána opakující se komunikace, tak aby počet záznamů nepřekročil 65. Takovéto množství záznamů dostatečně pokryje důležité pakety odeslané při připojení a odpojení VPN. V každém doloženém záznamu tak lze velice dobře pozorovat očekávanou komunikaci protokolů popsanou v teoretické části. Zároveň, kvůli celkovému množství ostatních přenášených paketů (nezávislých na připojení nebo odpojení VPN), jsou všechny záznamy ve Wiresharku vyfiltrovány pomocí příkazu „*ip.addr == 89.233.145.246*“. Díky tomu jsou v historii přenosu zobrazeny pouze pakety, které směřují k veřejné bráně routeru ER6020v2 a samozřejmě posléze k routeru ER6020v1 (VPN server).

Vzhledem k faktu, že na tomto zapojení nejsou prováděny žádné penetrační testy (testy bezpečnosti), tak při konfiguraci VPN jsou použity triviální hesla i předsdílené klíče (Pre-Share Key). To samé platí i o nastavených autentizačních a šifrovacích protokolech. Bezpečnost jednotlivých VPN je zde komentována pouze z teoretických znalostí.

Obrázek 12 – CMD – Traceroute



Zdroj: Vlastní

5.6. PPTP konfigurace a testování

Prvním testovaným VPN protokolem je protokol PPTP, jehož konfigurace je považována za nejsnazší. Na routeru je spuštěn PPTP server, na který je možné se ze vzdálené počítačové stanice „C“ připojit. Na novějších routerech od společnosti TP-Link (jako například ER6020v2) je možné poměrně jednoduše vytvořit jednotlivé uživatelské účty s přihlašovacími údaji. Tuto možnost bohužel ER6020v1 neumožňuje. V rámci této práce je ovšem jeden účet dostačující.

5.6.1. Nastavení PPTP serveru

Na routeru je záložka „VPN => L2TP/PPTP“ rozdělena na sekci General a L2TP/PPTP Tunnel. Pro spuštění PPTP serveru je přidáno následující nastavení:

General:

- Hello Interval: *60 Sec*
- Primary DNS: *192.168.1.1*
- Secondary DNS: *8.8.8.8*
- Net BIOS Pass.: *Enable*

Hello Interval je časový interval, po jehož uplynutí se odesílají PPP pakety. Zde je ponecháno výchozích 60 sec. Primary DNS je nastavena na bránu routeru ER6020v2 – 192.168.1.1 a Secondary DNS je nastavena na IP adresu Googlu – 8.8.8.8. NetBIOS Passthrough umožňuje vysílat prostřednictvím VPN tunelu NetBIOS pakety a je zde nastaven na Enable.

L2TP/PPTP Tunnel:

- Protocol: *PPTP*
- Mode: *Server*
- Account Name: *PPTPuzivatel*
- Password: *Heslo123*
- Tunnel: *Client-to-LAN*
- Max Connections: *1*
- Encryption: *Enable*
- Pre-Share Key: –
- Client IP: –
- IP Adress Pool: *VPN Pool*
- Remote Subnett: –
- Status: *Active*

Jako Protocol je pochopitelně vybrán testovaný PPTP. Po vybrání tohoto protokolu se zároveň znemožní použití předsdíleného klíče (Pre-Shared Key) a Client IP. Tyto parametry lze nastavovat pouze s použitím protokolu L2TP. Account Name je pro pořádek zvolen jako „PPTPuzivatel“ a Password pro zjednodušení „Heslo123“. Jako Mode musí být vybrán Server, aby se na něj ostatní vzdálená zařízení mohla připojovat. Tunnel je nastaven na Client-to-LAN a pro Max Connections (maximum současně přihlášených uživatelů pod tímto účtem) bude stačit „1“. S vybráním možnosti Client-to-LAN se znemožní i vyplnění Remote Subnett, jenž se používá pouze pro typ spojení LAN-to-LAN. Encryption je nastaveno na Enable, aby komunikace napříč VPN tunelem byla šifrována. Jako IP Adress Pool je zvolen předdefinovaný IP rozsah „VPN pool“. Tím je zaručeno, že připojené zařízení dostane vždy IP adresu z prvního rozsahu.

Po dokončení tohoto nastavení lze pomocí tlačítka Add tento nově vytvořený PPTP Tunel přidat. Tunel lze pak nadále spravovat níže v List of Configuration.

5.6.2. Nastavení PPTP klienta

Vzhledem k tomu, že Windows podporuje protokol PPTP je přihlášení možné přímo ve Windows rozhraní. V nabídce start je třeba vyhledat „Nastavení sítě VPN“. Zde je v záložce „VPN => Přidat připojení VPN“ přidáno nové cílové připojení VPN. Přihlašovací údaje a informace jsou následovné:

- Poskytovatel připojení VPN: *Windows*
- Název připojení: *PPTP*
- Název nebo adresa serveru: *89.233.145.246*
- Typ sítě VPN: *PPTP*
- Typ přihlašovacích údajů: *Uživatelské jméno a heslo*
- Uživatelské jméno: *PPTPuzivatel*
- Heslo: *Heslo123*

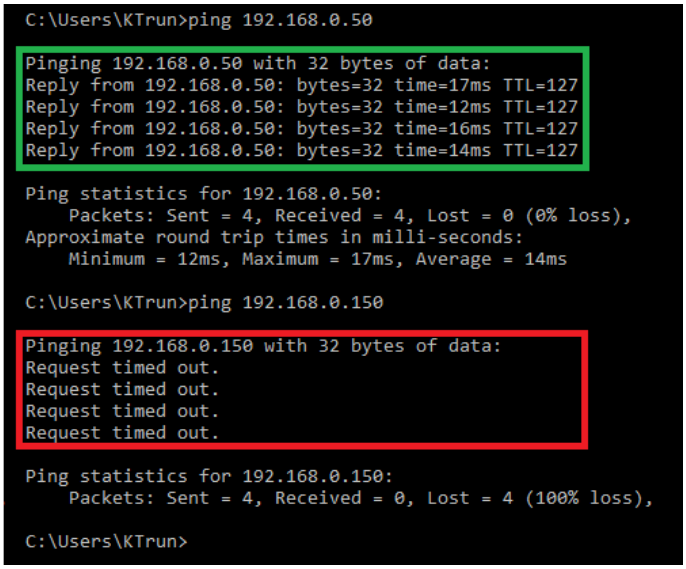
Poskytovatel připojení VPN je předdefinován jako Windows. Název připojení je vlastní název uživatele počítačové stanice „C“, pod kterým se toto připojení zobrazuje v seznamu dostupných VPN. Zde je vyplněno „PPTP“. Název nebo adresa serveru je vyplněna jako veřejná IP adresa routeru ER6020v2 – 89.233.145.246 a typ sítě VPN je pochopitelně PPTP. K přihlášení dojde pomocí příslušného uživatelského jména a hesla účtu. Toto nově nastavené připojení se posléze objeví v seznamu dostupných VPN.

Ještě před připojením je třeba upřesnit nastavení adaptéru. Ve stávajícím okně VPN je ve sloupci vpravo odkaz „Změnit možnosti adaptéru“. Po kliknutí na něj se otevře seznam síťových připojení. Zde se nachází i nově vytvořené připojení PPTP. Po pravém kliknutím myši je vybrána možnost „Vlastnosti“. V nově otevřeném okně je pod záložkou „Zabezpečení“ zvolen Typ VPN jako „Point to Point Tunneling Protocol (PPTP)“ a v sekci Autentifikace zaškrtnuto „Povolit tyto protokoly“, „Challnage Handshake Authentication Protocol (CHAP)“ a „Microsoft CHAP version 2 (MS-CHAP v2)“. V posledním kroku je ještě třeba v záložce „Síť“ upravit nastavení IPv4, aby vzdálený hostitel přistupoval k internetu přes vzdálenou bránu sítě. Veřejná IP adresa počítačové stanice „C“ bude po připojení stejná jako veřejná IP adresa routeru ER6020v2, tedy 89.233.145.246. Ve vlastnostech „Protokolu IP verze 4 (TCP/IPv4)“ je po rozkliknutí tlačítka „Upřesnit“ v nově otevřeném okně zaškrtnuta možnost „Používat výchozí bránu vzdálené sítě“. Po uložení tohoto nastavení je možné PPTP připojit.

Po úspěšném připojení je u této VPN přepnut profil sítě z Veřejné na Privátní. Díky tomu mají počítačové stanice, které jsou fyzicky uvnitř lokální sítě routeru ER6020v1 přístup ke vzdálené počítačové stanici „C“. Nastavení profilu sítě je možné v „Upřesnit možnosti“ v přehledu VPN.

Po ověření testu ping, kdy je úspěšně volána počítačová stanice „A“ (192.168.0.50) a neúspěšně počítačová stanice „B“ (192.168.0.150) lze shledat připojení za kompletní a postoupit k dalšímu testu. Záznam z příkazového řádku je na Obrázku 13.

Obrázek 13 – PPTP – CMD – Test Ping



Zdroj: Vlastní

5.6.3. Test mezi počítačovou stanicí „A“ a „C“

Mezi vzdálenými počítačovými stanicemi „A“ a „C“, které jsou propojeny pomocí VPN PPTP je měřena přenosová rychlost, ztrátovost UDP paketů a latence. Pro měření přenosové rychlosti a ztrátovosti UDP paketů je využit program TamoSoft Throughput Test. Počítačová stanice „A“ bude pro tento program složit jako server a počítačová stanice „C“ jako klient. Na obou stanicích je nastaven server port na 27100 a z klientské části je volána IP adresa serveru, tedy 192.168.0.50. Každý test je po minutovém intervalu resetován, a spuštěn znovu. Pro změření latence mezi počítačovou stanicí „A“ a „C“ je využit příkazový řádek a funkce Ping. Na počítačové stanici „C“ je do příkazového řádku zadán příkaz „*ping 192.168.0.50*“, díky čemuž je čtyřikrát volána počítačová stanice „A“ a zároveň příkazový řádek vypíše i dobu odezvy (latenci). Každý test byl proveden celkem 10x. Naměřené výsledky jsou zobrazeny v Tabulce 6.

Tabulka 6 – PPTP – PC:A / PC:C – TamoSoft, CMD

	TCP [Mbps]		UDP [Mbps]		Lost packets [%]		Ping [ms]
	UP	DOWN	UP	DOWN	UP	DOWN	
1	8,79	8,76	/	/	/	/	16
2	8,58	5,42	/	/	/	/	20
3	7,56	13,13	/	/	/	/	15
4	7,46	11,69	/	/	/	/	16
5	8,75	12,19	/	/	/	/	30
6	7,94	13,70	/	/	/	/	17
7	7,72	12,96	/	/	/	/	15
8	7,96	12,93	/	/	/	/	16
9	7,41	12,07	/	/	/	/	20
10	8,50	13,50	/	/	/	/	19
Průměr:	**8,07**	**11,64**	/	/	/	/	**18,40**

Zdroj: Vlastní

Při odesílání TCP paketů se upload rychlost pohybuje v minimálním rozmezí 7,41 Mbps až 8,79 Mbps. Z deseti prováděných měření je průměrná hodnota 8,07 Mbps. U downloadu je rozmezí rychlosti o něco větší, a to sice od 5,42 Mbps až do 13,7 Mbp. Průměrná rychlost downloadu je tak 11,64 Mbps. Test downloadu, uploadu a ztrátovosti UDP paketů musí být vynechán. V aplikaci TamoSoft je zaškrtnuto políčko „TPC only“, které omezí odesílání datových paketů pouze na typ TCP. Jinak při pokusu přenášet datové pakety typu UDP dochází k okamžitému odpojení od VPN. Dle samotného vyjádření společnosti TamoSoft (Milla Bren – Certified Wireless Network Professional) je tento problém způsoben vysokým zatížením provozu UDP přes PPTP tunel, který vyžaduje tak velkou šířku pásma, že některé potřebné pakety nedosahují na server / klienta a dochází k odpojení VPN. Latence mezi počítačovou stanici „A“ a „C“ je poměrně vysoká. V těchto deseti prováděných testech se pohybuje mezi 15ms až 30ms, což ve výsledku činí průměrnou hodnotu 18,4ms.

5.6.4. Test z počítačové stanice „C“ do Internetu

Na počítačové stanici „C“ jsou pomocí webového portálu „speedtest.cesnet.cz/“ měřeny hodnoty ping, jitter, download a upload. Adaptér VPN je nastaven tak, aby do Internetu přistupoval přes vzdálenou bránu routeru ER6020v1 (dále pak přes bránu routeru ER6020v2). Před začátkem měření je pomocí příkazu „*tracert 8.8.8.8*“ v příkazovém

řádku zkontrolována správnost konfigurace. Test byl proveden celkem 10x. Změřené hodnoty jsou v Tabulce 7.

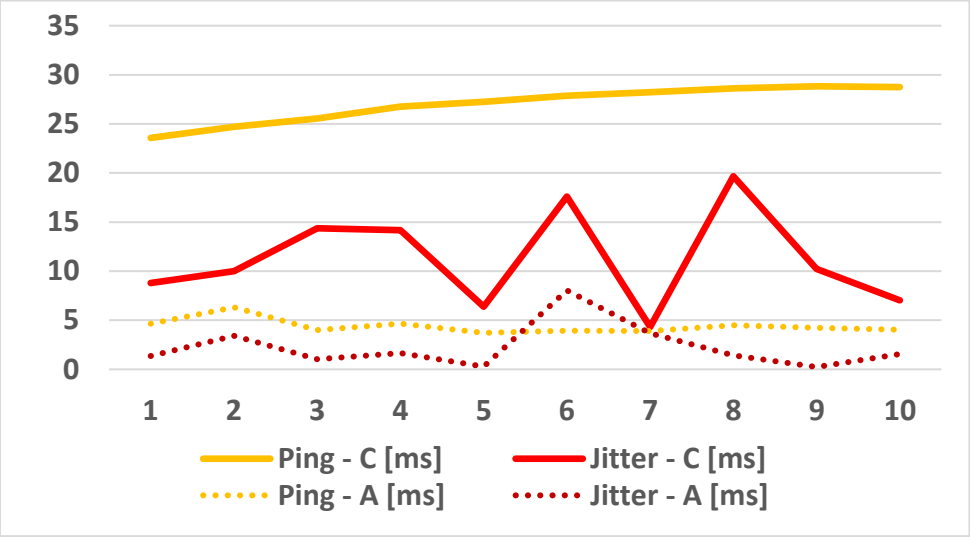
Tabulka 7 – PPTP – PC:C / Internet – CESNET – Ping, Jitter, Download, Upload

	Ping [ms]	Jitter [ms]	Download [Mbps]	Upload [Mbps]
1.	23,57	8,80	13,43	10,25
2.	24,69	10,00	13,55	10,70
3.	25,56	14,38	13,64	9,98
4.	26,76	14,17	13,85	11,58
5.	27,26	6,38	13,05	9,53
6.	27,87	17,60	13,83	10,95
7.	28,23	4,32	13,74	10,64
8.	28,60	19,66	13,28	10,99
9.	28,83	10,23	13,47	10,37
10.	28,76	7,02	13,25	11,36
Průměr:	**27,01**	**11,26**	**13,51**	**10,64**

Zdroj: Vlastní

Naměřené hodnoty ping, download a upload lze shledat poměrně za konstantní, avšak naměřené hodnoty jitter kolísají ve značně větším rozmezí. Dle očekávání, lze u všech těchto hodnot pozorovat značné zhoršení výsledků oproti stejnému testu prováděném na počítačové stanici „A“, která je fyzicky umístěna za routerem ER6020v1. Celkový průměr z deseti měření hodnoty ping je 27,01 ms. Na počítačové stanici „A“ byl oproti tomu naměřený průměrný ping jen 4,39 ms. Výsledky jitter se pohybovaly v poměrně velkém rozmezí 4,32 ms až 19,66 ms a průměr je tedy 11,26 ms, ovšem na počítačové stanici „A“ byla průměrná hodnota jitter pouze 2,27 ms. Největší propad je zřejmý u hodnot download a upload. Na počítačové stanici „A“ byl změřen průměrný download 94,82 Mbps a upload 84,71 Mbps. Po připojení VPN se na počítačové stanici „C“ pohyboval download kolem průměrné hodnoty pouhých 13,51 Mbps a upload kolem průměrné hodnoty 10,64 Mbps. Propad hodnot ping a jitter oproti naměřeným výsledkům na stanici „A“ je zobrazen v Grafu 1.

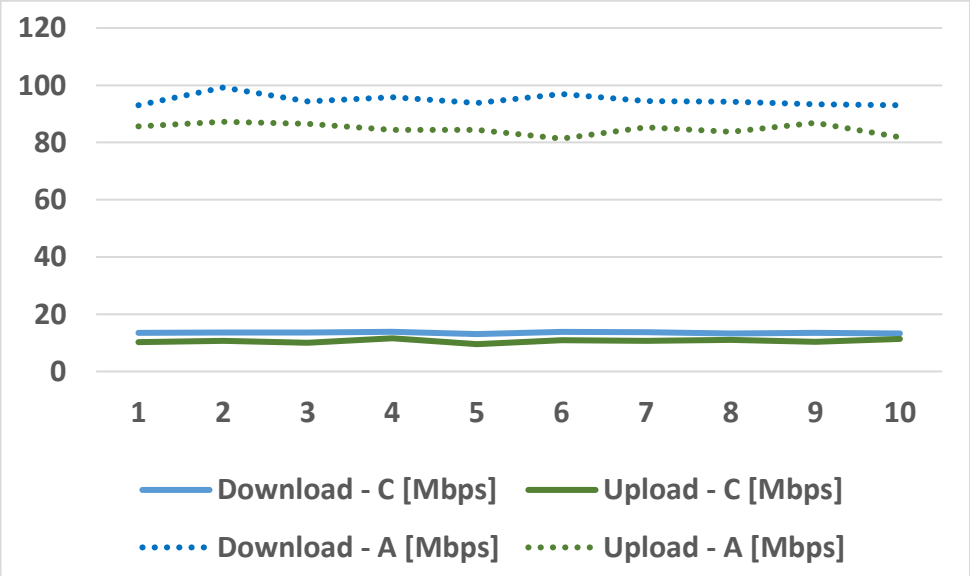
Graf 1 – PPTP – PC:A / PC:C – Ping, Jitter



Zdroj: Vlastní

Plná čára představuje naměřené hodnoty na počítačové stanici „C“ a tečkovaná čára představuje naměřené hodnoty na počítačové stanici „A“. Na následujícím Grafu 2 jsou obdobně zobrazeny plnou čarou naměřené hodnoty download a upload na počítačové stanici „C“ a tečkovanou čarou naměřené hodnoty na počítačové stanici „A“.

Graf 2 – PPTP – PC:A / PC:C – Download, Upload



Zdroj: Vlastní

5.6.5. Záznam přenášených paketů

Z doložených záznamů komunikace v přílohách je na první pohled zřejmé, že komunikace PPTP (viz Příloha I) je oproti IPsecu nebo L2TP/IPsec značně odlišná. Celá komunikace probíhá na základě výše zmiňovaných protokolů TCP, PPTP, GRE a protokoly, které spadají pod protokol PPP – LCP, CHAP, IPCP a CCP (Comp).

- **PPP LCP:**

  Při nastavování komunikace PPP obě strany odesílají pakety typu Link Control Protocol (LCP). Tento protokol určuje standardy následného přenosu dat. Zařízení v síti nemohou používat protokol PPP k přenosu dat, dokud pakety LCP neurčí přijatelnost spojení.

- **PPP CHAP:**

  Ověřovací protokol Challenge Handshake Authentication Protocol (CHAP) slouží k ověření totožnosti uživatele. V tomto případě ověření pomocí jména a hesla. Jméno uživatele je zasíláno ve formátu prostého textu, a je tak dokonce čitelné v jednom PPP CHAP paketu (Number 964).

- **PPP IPCP:**

  Internet Protocol Control Protocol (IPCP) je protokolem spadajícím pod Network Control Protocol (NCP), který vytváří a konfiguruje internetový protokol přes spojení typu point-to-point protocol (PPP). IPCP je zodpovědný za konfiguraci IP adres a také za povolení a zakázání modulů protokolu IP na obou koncích spojení. IPCP používá stejný mechanismus výměny paketů jako LCP.

- **PPP CCP (Comp):**

  Pomocí protokolu Compression Control Protocol (CCP) se obě strany dohodnou na algoritmu použitém pro kompresi dat.

5.7. IPsec konfigurace a testování

Konfigurace VPN IPsec je o něco komplikovanější než u PPTP nebo L2TP. IPsec na rozdíl od PPTP a L2TP neumožňuje přihlášení přes Windows rozhraní. To znamená, že na počítačové stanici „C“ musí být nainstalovaný VPN klient – jako je například právě aplikace TheGreenBow VPN, který připojení umožní. IPsec zároveň využívá protokol IKE. Právě proto musí být ve Službách („Start => Služby“) počítačové stanice „C“ spuštěna služba s názvem „IKE and AuthIP IPsec Keying Modles“, jinak spojení nelze navázat.

Konfigurace IPsecu na routeru ER6020v1 je kaskádovitá a rozdělená do čtyř částí, kde je každá část závislá na předchozí. Nejprve musí být nastaven IKE Proposal, který je využit při následující konfiguraci IKE Policy. Posléze je možné nakonfigurovat IPsec Proposal, který je společně s IKE Policy potřebný při závěrečné konfiguraci IPsec Proposal.

5.7.1. Nastavení IPsec serveru

Zprvu je tedy nutné nastavit IKE Proposal a IKE Policy. Konfigurace IKE Proposal je možná v záložce „VPN => IKE => IKE Proposal“.

IKE Proposal:

- Proposal Name: *IKEproposal*
- Authenticaton: *SHA-1*
- Encryption: *3DES*
- DH Group: *DH2*

Proposal Name je nastaven na „IKEproposal“. Pro Authenticaton je vybrán protokol SHA-1 a pro Encryption protokol 3DES. Diffie-Hellman (DH) Group určuje sílu klíče použitého v procesu výměny klíčů. Vyšší čísla skupin jsou bezpečnější, ale vyžadují více času na výpočet. Zde je vybrán DH2 – Diffie-Hellman 2: 1024bit. Po přidání tohoto nastavení je možné přejít do záložky „VPN => IKE => IKE Policy“ a IKE Policy nastavit.

IKE Policy:

- Policy Name: *IKEpolicy*
- Exchange Mode: *Main*
- Local ID Type: *FQDN*
- Local ID: *1234*
- Remote ID Type: *FQDN*
- IKE Proposal 1: *IKEproposal*
- Pre-Share Key: *123456*
- SA Lifetime: *28800*
- DPD: *Disable*
- DPD Interval: –

Policy Name je nastaven na „IKEpolicy“. Jako Exchange Mode je vybrán pomalejší, ale bezpečnější Main Mode. Local ID Typ i Remote ID Typ je zde pro vyjednávání IKE

vybrán FQDN (Fully Qualified Domain Name) neboli plně specifikované doménové jméno. Local ID je pak nastaven na „1234“ a Remote ID na „4321“. IKE Proposal 1 je vybrán z předchozího kroku přednastavený „IKEproposal“. Při využití IPsec šifrování je nutné zvolit i Pre-Share Key. Pro zvýšení bezpečnosti je lepší, aby klíč byl dlouhý nelogický řetězec znaků. Pro vytvoření tohoto klíče je vhodné využít nějaký webový portál, který tento řetězec vygeneruje. V rámci této práce je ale pro usnadnění využit triviální klíč „123456“. SA Lifetime určuje při vyjednávání IKE životnost ISAKMP SA (po vypršení životnosti SA je odstraněn i související ISAKMP SA). V rámci této práce je ponecháno výchozí nastavení – 28800 sec. Dead Peer Detect (DPD) umožňuje koncovému bodu IKE zkontrolovat, jestli je partner stále aktivní. Zde je DPD vypnut, a tudíž je i znemožněno vyplnění posledního parametru DPD Interval.

Po uložení IKE Policy lze považovat nastavení IKE za kompletní a je možné přejít k nastavení IPsecu. Obdobně jako u IKE se proces konfigurace IPsec dělí na dvě části – IPsec Proposal a IPsec Policy. První je nutné nakonfigurovat IPsec Proposal („VPN => IPsec => IPsec Proposal“).

IPsec Proposal:

- Proposal Name: *IPsecProposal*
- Security Protocol: *ESP*
- ESP Authenticaton: *SHA-1*
- ESP Encryption: *3DES*

Proposal Name je tradičně nastaven na „IPsecProposal“. Jako Security Protocol je vybrán zabezpečený ESP s využitím protokolů SHA-1 a 3DES (ESP Authenticaton a Encryption). Po přidání tohoto IPsec Proposal je možné pokračovat v nastavení IPsec Policy („VPN => IPsec => IPsec Policy“).

General:

- IPsec: *Enable*

IPsec Policy:

- Policy Name: *IPsecPolicy*
- Mode: *Client-to-LAN*
- Local Subnet: *192.168.0.0/25*
- Remote Subnet: –

- WAN: *WAN1*
- Remote Host: *0.0.0.0*
- Policy Mode: *IKE*
- IKE Policy: *IKEpolicy*
- IPsec Proposal 1: *IPsecProposal*
- PFS: *NONE*
- SA Lifetime: *28800*
- Status: *Active*

V sekci General je IPsec spuštěn přepnutím do stavu Enable. Ve spodní sekci IPsec Policy je nastaven Policy Name jako „IPsecPolicy“ a Mode je vybrán jako Client-to-LAN. Local Subnet musí být nastaven ze stejného rozsahu jako je vnitřní brána VPN serveru. Zde je vyplněna adresa sítě 192.168.0.0/25. CIDR s číslem 25 rozděluje síť na dvě podsítě stejně, jako maska sítě 255.255.255.128 (jen jiný zápis stejné funkce). Remote Subnet po zvolení módu Client-to-LAN opět nelze vyplnit, protože je využíván pouze pro mód LAN-to-LAN. Jako WAN port je zvolen opět WAN1 s konektivitou do veřejné sítě (přes router ER6020v2). Remote Host zde funguje jako White List. Pro zvýšení bezpečnosti je zde možné vyplnit jednu konkrétní veřejnou IP adresu, z které se může klient přihlásit. Zde je ale Remote Host vyplněn jako 0.0.0.0. To reprezentuje libovolnou IP adresu a toto dodatečné zabezpečení je deaktivováno. Se správnými přihlašovacími údaji se pak může přihlásit kterýkoliv klient bez ohledu na jeho veřejnou IP adresu. V Policy Mode je vybrán IKE, aby v IKE Policy bylo možné vybrat předdefinovaný „IKEpolicy“ a stejně tak je pro IPsec Proposal 1 vybrán předdefinovaný „IPsecProposal“. Perfect Forward Secutiry (PFS) umožňuje zvýšení zabezpečení sítě. Při povolení PFS je vygenerován zvlášť klíč pro Fázi 2, který je nezávislý na klíči použitém ve Fázi 1. Zde je ovšem vybrána možnost NONE. Tím je PFS deaktivováno a klíč pro Fázi 2 je vygenerován na základě klíče použitého ve Fázi 1. SA Lifetime je opět ponechán ve výchozím nastavení 28800 sec. V závěru je IPsec spuštěn přepnutím Statusu na Active a následném uložení.

5.7.2. Nastavení IPsec klienta

Po spuštění aplikace TheGreenBow se otevře okno Configuration Panel, kde je v levém sloupci několik záložek. Kliknutím pravým tlačítkem myši na „IKE V1“ se otevře menu kde je třeba vybrat poslední možnost „New Phase 1“ (Ctrl+N). V okně „Authentication“ jsou vyplněny následující parametry:

Remote Gateway:

- Interface: *Any*
- Remote Gateway: *89.233.145.246*

Authnetication:

- Preshared Key: *123456*
- Confirm *123456*

Cryptography:

- Encryption: *3DES*
- Authentication: *SHA-1*
- Key Group: *DH2 (1024)*

Interface je IP adresa síťového rozhraní počítače. Pokud se tato IP adresa může změnit, je lepší vybrat možnost „Any“. Remote Gateway je vyplněn jako veřejná adresa routeru ER6020v2 (89.233.145.246). V sekci Authnetication a Cryptography jsou vyplněny všechny parametry totožně jako tomu bylo při nastavování serveru. Pre-share Key je tedy vyplněn jako „123456“ a znovu potvrzen, jako šifrovací algoritmus je vybrán 3DES, jako autentizační protokol SHA-1 a Key Group je zvolen DH2 (1024). V záložce „Protocol“ je ještě třeba vyplnit Local a Remote ID stejně jako jsou vyplněny na VPN serveru a NAT–T nastavit na „Automatic“.

- Local ID: *DNS 4321*
- Remote ID: *DNS 1234*

Po dokončení nastavení Fáze 1 lze přejít ke konfiguraci Fáze 2. V levém sloupci je vybrána záložka „Gateway“ a po kliknutí pravým tlačítkem myši je v menu vybrána poslední možnost „New Phase 2“ (Ctrl+N), která otevře okno „IPsec“.

Addresses:

- VPN Client address: *192.168.1.20*
- Address type: *Subnet address*
- Remote LAN address: *192.168.0.0*
- Subnet mask: *255.255.255.128*

ESP:

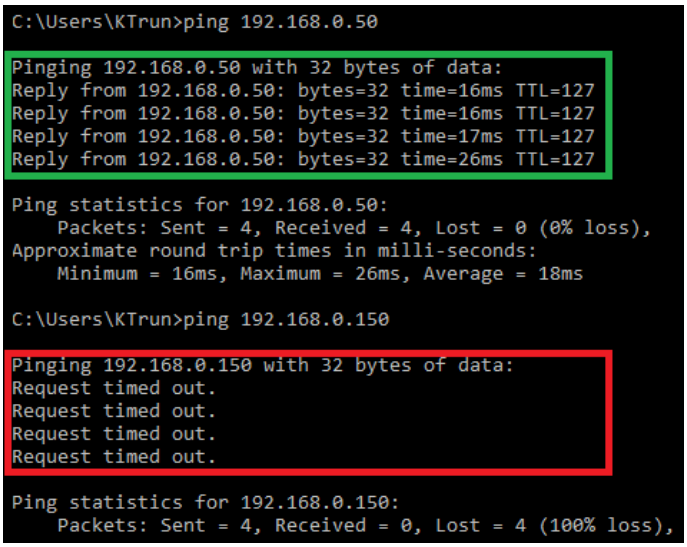
- Encryption: *3DES*
- Authentication: *SHA-1*

- Mode: *Tunnel*

VPN Client address je přidělená IP adresa tohoto zařízení. V případě využití aplikace Green Bow musí být IP adresa z jiného rozsahu než vzdálená lokální síť. Zde je IP adresa nastavena na 192.168.1.20. Z otevíracího seznamu Address type je vybrána možnost „Subnet address“. Remote LAN address je vyplněna jako 192.168.0.0 a maska sítě jako 255.255.255.128, což je v podstatě jen jiný zápis adresy sítě 192.168.0.0/25, která je takto nastavena na VPN serveru. Díky tomuto nastavení má počítačová stanice „C“ přístup pouze k zařízením v první podsíti. V sekci ESP je pak vyplněno opět šifrování pomocí 3DES, autentizace pomocí SHA-1 a Mode je zvolen jako „Tunnel“.

Po dokončení konfigurace Fáze 2 je možné v levém sloupci kliknout pravým tlačítkem myši na „Tunnel“ a připojit VPN kliknutím na „Open tunnel“ (Ctrl+O). Správnost zapojení je možné ověřit na routeru v záložce „VPN => IPsec => IPsec SA“. Zde se v seznamu připojených zařízení nachází i počítačová stanice „C“. IPsec neumožňuje v nastavení Windows změnit profil sítě z Veřejné na Privátní (jako tomu bylo u PPTP). Z toho důvodu je na počítačové stanici „A“ vypnut firewall, který jinak volání příkazu ping blokuje. Po ověření funkcí ping, kdy je úspěšně z počítačové stanice „C“ volána počítačová stanice „A“ a neúspěšně počítačová stanice „B“ lze přejít k dalšímu testu. Viz Obrázek 14 s doloženým záznamem z příkazového řádku o úspěšné konfiguraci.

Obrázek 14 – IPsec – CMD – Test Ping



Zdroj: Vlastní

5.7.3. Test mezi počítačovou stanicí „A“ a „C“

Na počítačové stanici „A“ i „C“ je v programu TamoSoft nastaven server port na 27100. Počítačové stanice „C“ opět slouží jako klient a počítačová stanice „A“ jako server. Z počítačové stanice „C“ je tedy volána IP adresa stanice „A“ (192.168.0.50). Test je vždy po minutovém intervalu spuštěn znovu. K změření latence je použit příkazový řádek a z počítačové stanice „C“ je volána počítačová stanice „A“. Naměřené výsledky jsou zaznamenány v Tabulce 8.

Tabulka 8 – IPsec – PC:A / PC:C – TamoSoft, CMD – Main Mode

	TCP [Mbps]		UDP [Mbps]		Lost packets [%]		Ping [MS]
	UP	**DOWN**	**UP**	**DOWN**	**UP**	**DOWN**	
1	19,10	36,52	10,71	26,95	87,4%	0,0%	16
2	18,46	37,13	11,08	26,95	87,0%	0,0%	10
3	18,94	37,02	10,86	26,87	85,0%	0,0%	11
4	18,96	36,72	10,91	26,84	85,1%	0,0%	10
5	18,93	36,13	12,14	26,96	79,0%	0,0%	8
6	19,09	37,67	10,98	27,04	84,8%	0,0%	12
7	18,78	37,01	11,63	26,82	82,0%	0,0%	12
8	19,21	38,65	10,72	25,89	85,8%	0,0%	14
9	19,12	35,63	10,97	26,46	84,3%	0,0%	10
10	17,71	29,86	10,80	27,15	82,9%	0,1%	8
Průměr:	**18,83**	**36,23**	**11,08**	**26,79**	**84,33%**	**0,01%**	**11,10**

Zdroj: Vlastní

Rychlost downloadu se ve všech měření projevila podstatně vyšší než rychlost uploadu. TCP upload hodnoty nijak výrazně nevybočují z řady a vždy se pohybují kolem průměrných 18,83 Mbps. U TCP downloadu, až na poslední měření, kdy rychlost spadla na 29,86 Mbps, se všechny hodnoty rovněž pohybují kolem průměrných 36,23 Mbps. Rychlost UDP je u uploadu i downloadu stabilní s průměrnou hodnotou uploadu 11,08 Mbps a downloadu 26,79 Mbps. Průměrná ztrátovost UDP paketů je 84,33 % při uploadu a téměř nulová při downloadu. Průměrná latence z deseti měření mezi počítačovou stanicí „A“ a „C“ je 11,10 ms.

5.7.4. Test z počítačové stanice „C“ do Internetu

Router ER6020v1 bohužel v případě IPsecu nepodporuje režim Mode Config, který je zde nezbytný pro nastavení módu „no split tunneling“ neboli nelze přesměrovat komunikaci do Internetu přes vzdálenou bránu VPN serveru (router ER6020v1). Z tohoto

důvodu musí být tento test vynechán, protože bez tohoto nastavení je veškerá komunikace do Internetu směrována přes UPC router před počítačovou stanicí „C“.

Pro ověření, že se chyba není na straně VPN klienta nebo počítačové stanice „C“, byl proveden test, kdy se počítačová stanice pomocí aplikace Green Bow VPN připojila k VPN serveru spuštěnému na routeru od společnosti Zyxel – ATP200, který toto nastavení umožňuje. Při konfiguraci na VPN serveru musel být spuštěn Mode Config. Následně na VPN klientu byla nastavena Remote LAN address i Subnet mask na 0.0.0.0 a v záložce „IKEv1 Parameters“ bylo zaškrtnuto políčko „Disable Split Tunneling“. Po této konfiguraci počítačová stanice dle očekávání přistupovala k Internetu přes vzdálenou bránu VPN serveru, který byl realizován na routeru ATP200.

5.7.5. Záznam přenášených paketů

Dokumentace záznamu komunikace je v případě IPsecu doložena dvakrát. Jednou pro současně nastavený Main Mode a podruhé je Exchange Mode přenastaven na rychlejší, avšak méně bezpečný Aggressive Mode. Pro přepnutí módu je třeba na VPN serveru deaktivovat IPsec („VPN => IPsec => List of Configuration“), v IKE Policy přepnout Exchange Mode na Aggressive, opět aktivovat IPsec a v aplikaci GreenBow VPN zaškrtnout ve Fázi 1 v záložce Protocol pole „Aggressive Mode“.

V Příloze II je doložen záznam z přenosu paketů s nastaveným Hlavním módem (Main Mode) a v Příloze III je záznam s nastaveným Agresivním módem (Aggressive Mode). Oba záznamy jsou, až na několik prvních paketů, které určují, zdali se jedná o Hlavní nebo Agresivní mód zcela totožné.

Po připojení VPN dojde pomocí protokolu ISAKMP k vyjednání první i druhé fáze protokolu IKE. Vyjednání první fáze je v případě Hlavního módu rozdělena do 6 paketů (šesticestný handshake) a v případě Agresivního módu do 3 paketů (třícestný handshake). Vyjednání druhé fáze (Quick Mode) se poté skládá vždy ze tří paketů. Veškerá následující komunikace je šifrována pod protokolem ESP, a tudíž je nečitelná. Při odpojení VPN dojde k výměně tří informačních paketů protokolu ISAKMP, které připojení ukončí.

V záznamu komunikace se vždy nachází i paket typu UDPENCAP (NAT-keepalive). NAT a NAPT upravují hlavičku IP paketu a způsobí, že pakety chráněné protokolem AH nemohou provést ověření kontrolním součtem. NAPT, který upravuje TCP a UDP porty, nemůže upravit porty v šifrovaném záhlaví TCP paketu chráněného protokolem ESP. Tento problém však řeší UDP Encapsulation (UDP zapouzdření). V praxi se UDP zapouzdření

používá pouze na paketech ESP. NAT nebo NAPT pak může upravit nezašifrované IP záhlaví ESP paketu zapouzdřeného v UDP, aniž by došlo k narušení autentizace, nebo šifrování. [40]

5.8. L2TP konfigurace a testování

Posledním měřeným VPN protokolem je L2TP. Postup konfigurace L2TP je velice obdobný jako u PPTP. Při využití protokolu L2TP je přihlášení z počítačové stanice „C“ opět možné z Windows rozhraní. Na počítačové stanici „C“ muselo dojít k několika specifickým úpravám. Jak už bylo zmíněno, L2TP používá pro šifrování IPsec. To znamená, že ve Službách musela zůstat povolena Služba „IKE and AuthIP IPsec Keying Modles“, jako tomu bylo u předchozí konfigurace IPsec. Navíc, vzhledem k tomu, že se VPN server nachází za NAT zařízením bylo třeba přidat registr DWORD. V Editoru registru („Windows => Start => Regedit“) je ve větvi „Počítač\ HKEY_LOCAL_MACHINE\ SYSTEM\ CurrentControlSet\ Services\ PolicyAgent“ přidána nová hodnota DWORD s názvem „AssumeUDPEncapsulationContextOnSendRule“. Hodnota této položky je nastavena na „2“ (0 = není povoleno se připojit k serverům, které jsou za NAT-T, 1 = je povoleno se připojit k serverům, které jsou za NAT-T, nicméně klienti NEMOHOU být za NAT-T, 2 = klient i server mohou být za NAT-T). Posléze je nutné počítač restartovat. Bez tohoto nastavení se nelze k VPN serveru umístěným za NAT zařízením připojit. [41]

5.8.1. Nastavení L2TP serveru

Nastavení L2TP serveru je na stejné záložce jako PPTP – „VPN => L2TP/PPTP“. Konfigurace L2TP je rozdělena do sekce General a sekce L2TP/PPTP Tunnel. Vzhledem k tomu, že L2TP využívá šifrování IPsec musí být IPsec v záložce „VPN => IPsec => IPsec Proposal => General“ spuštěn přepnutím do stavu Enable!

General:

- Hello Interval: *60 Sec*
- Primary DNS: *192.168.1.1*
- Secondary DNS: *8.8.8.8*
- Net BIOS Pass.: *Enable*

Nastavení zde zůstalo stejné jako u PPTP. Hello Interval je ponechán ve výchozím nastavení 60 sec, Primary DNS je nastavena na vnitřní bránu routeru ER6020v2, Secondary DNS na IP adresu Googlu a NetBIOS Passthrough je nastaven na Enable.

L2TP/PPTP Tunnel:

- Protocol: *L2TP*
- Mode: *Server*
- Account Name: *L2TPuzivatel*
- Password: *Heslo123*
- Tunnel: *Client-to-LAN*
- Max Connections: *1*
- Encryption: *Enable*
- Pre-Share Key: *123456*
- Client IP: *0.0.0.0*
- IP Adress Pool: *VPN Pool*
- Remote Subnett: –
- Status: *Active*

Jako Protocol je vybrán testovaný L2TP. Accout Name a Password jsou nastaveny na „L2TPuzivatel“ a „Heslo123“. Mode musí být nastaven jako Server. Tunnel je vybrán Client-to-LAN a u Max Connections je nastavena „1“. Encryption je povoleno, aby komunikace přes veřejnou síť byla zabezpečena. Vzhledem k tomu, že L2TP využívá k šifrování IPsec, tak je zde třeba vyplnit i Pre-Share Key. Pro jednoduchost je stejně jako u předchozího měření zvolen primitivní předsdílený klíč „123456“. S vybráním možnosti Client-to-LAN se opět zneaktivní Remote Subnett. Client IP zde funguje jako White List. Pro zvýšení bezpečnosti zde lze vyplnit jednu konkrétní veřejnou IP adresu klienta, ze které je možné se přihlásit. L2TP tedy poté povolí připojení pouze takovému uživateli, který se pod touto IP adresou nachází a všechny ostatní zamítne. Při této konfiguraci je Client IP vyplněn jako 0.0.0.0, což představuje libovolnou adresu a k VPN se lze připojit z jakékoliv veřejné IP adresy. Aby se vzdálená počítačová stanice po připojení nacházela v první podsíti je jako IP Adress Pool vybrán „VPN pool“.

Pomocí tlačítka Add lze tento L2TP tunel zaktivnit. Tunel lze následně spravovat v List of Configuration.

5.8.2. Nastavení L2TP klienta

Přihlášení na počítačové stanici „C“ s využitím protokolu L2TP je díky podpoře Microsoftu možné přímo z Windows rozhraní. Je opět nutné přidat připojení VPN („Start => Nastavení sítě VPN => VPN => Přidat připojení VPN“). Přihlašovací údaje a informace jsou vyplněny následovně:

- Poskytovatel připojení VPN: *Windows*
- Název připojení: *L2TP*
- Název nebo adresa serveru: *89.233.145.246*
- Typ sítě VPN: *L2TP/IPsec*
- Předsdílený klíč: *123456*
- Typ přihlašovacích údajů: *Uživatelské jméno a heslo*
- Uživatelské jméno: *L2TPuzivatel*
- Heslo: *Heslo123*

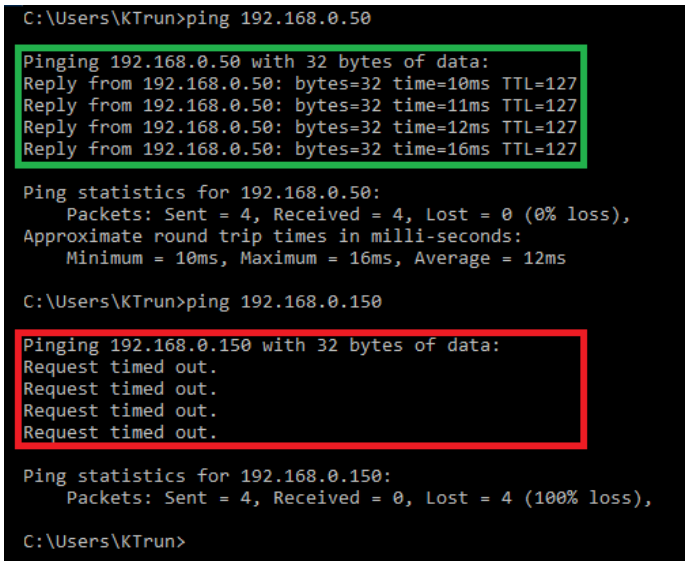
Jako Poskytovatel připojení VPN musí být zvolen předdefinovaný Windows. Název připojení je nastaven na „L2TP“ a veřejná IP adresa routeru ER6020v2 (89.233.145.246) je nastavena jako adresa serveru. Jako typ sítě je vybrán protokol L2TP s podporou šifrování IPsec (L2TP/IPsec). Přihlášení je možné po zadání správného uživatelského jména a hesla účtu.

Před připojením je třeba stejně jako u PPTP upřesnit nastavení adaptéru. Do nastavení se opět přistupuje přes odkaz „Změnit možnosti adaptéru“. V seznamu Síťových připojení se nachází právě vytvořené připojení L2TP. Ve vlastnostech připojení je třeba na záložce „Zabezpečení“ vybrat Typ VPN jako „Layer 2 Tunneling Protocol with IPsec (L2TP/IPsec)“. Následně je nutné upravit nastavení IPv4 adaptéru, aby měl uživatel přístup k Internetu přes vzdálenou bránu routeru ER6020v1. Na záložce Síť je ve vlastnostech „Protokolu IP verze 4 (TCP/IPv4)“ tlačítko „Upřesnit“. V nově otevřeném okně je třeba zaškrtnout možnost „Používat výchozí bránu vzdálené sítě“ a nastavení uložit.

Po úspěšném připojení je u této VPN přepnut profil sítě z Veřejné na Privátní, aby počítačová stanice „C“ byla v rámci této sítě viditelná pro ostatní zařízení.

Po ověření testu ping, kdy je úspěšně volána počítačová stanice „A“ (192.168.0.50) a neúspěšně počítačová stanice „B“ (192.168.0.150) lze považovat natavení za kompletní a pokračovat k dalším testům. Záznam z příkazového řádku o úspěšné konfiguraci je na Obrázku 15.

Obrázek 15 – L2TP – CMD – Test Ping



Zdroj: Vlastní

5.8.3. Test mezi počítačovou stanicí „A“ a „C“

Na počítačové stanici „A“ i „C“ je opět v programu TamoSoft nastaven server port na 27100. Klient na počítačové stanici „C“ se po zadání IP adresy 192.168.0.50 připojí k serveru na počítačové stanici „A“. Test je po minutovém intervalu vždy resetován. Pro změření latence je z počítačové stanice „C“ pomocí příkazového řádku volána počítačová stanice „A“. Každý test je proveden celkem 10x a naměřené výsledky jsou zobrazeny v Tabulce 9.

Tabulka 9 – L2TP – PC:A / PC:C – TamoSoft, CMD

	TCP [Mbps]		UDP [Mbps]		Lost packets [%]		Ping [MS]
	UP	**DOWN**	**UP**	**DOWN**	**UP**	**DOWN**	
1	18,12	25,24	20,36	30,72	80,5%	0,0%	12
2	18,88	24,73	20,25	30,57	81,0%	0,0%	18
3	18,16	19,74	20,34	30,66	79,9%	0,0%	10
4	18,44	24,45	20,09	30,66	79,5%	0,0%	10
5	18,82	24,57	20,29	30,76	80,2%	0,0%	14
6	18,43	19,81	20,22	30,59	77,9%	0,0%	8
7	18,28	23,98	20,29	30,39	78,6%	0,0%	9
8	18,51	23,58	20,34	30,36	80,0%	0,0%	13
9	18,78	22,78	20,31	30,42	79,6%	0,0%	10
10	18,74	24,39	20,43	30,43	76,9%	0,0%	12
Průměr:	**18,52**	**23,33**	**20,29**	**30,56**	**79,41%**	**0,00%**	**11,60**

Zdroj: Vlastní

U L2TP žádné naměřené výsledky nijak výrazně nevybočují z řady. TCP upload pohybuje v minimálním rozmezí 18,12 Mbps až 18,88 Mbps a průměr z těchto deseti měření je 18,52 Mbps. U downloadu je rozmezí o něco větší, a to sice od 19,74 Mbps do 25,24 Mbps s průměrem 23,33 Mbps. U UDP lze pozorovat značně konstantní hodnoty jak u uploadu, tak downloadu. Upload se tu pohybuje vždy kolem průměrných 20,29 Mbps a stejně tak download kolem 30,56 Mbps. Latence je při tomto měření velice obstojná. Průměrná odezva z deseti volání počítačové stanice „A“ z počítačové stanice „C“ je 11,6ms.

5.8.4. Test z počítačové stanice „C“ do Internetu

VPN adaptér na počítačové stanici „C“ je opět nastaven tak, aby k Internetu přistupoval přes vzdálenou bránu VPN serveru. Pomocí příkazového řádku a příkazu „*tracert 8.8.8.8*“ je volán vzdálený server umístěný ve veřejné síti. Pokud datové pakety cestují nejprve k veřejné bráně VPN serveru, lze konfiguraci shledat za úspěšnou a přejít k poslednímu testu. Na webovém portálu „speedtest.cesnet.cz/“ jsou celkem 10x změřeny hodnoty ping, jitter, download a upload. Změřené výsledky jsou v Tabulce 10.

Tabulka 10 – L2TP – PC:C / Internet – CESNET – Ping, Jitter, Download, Upload

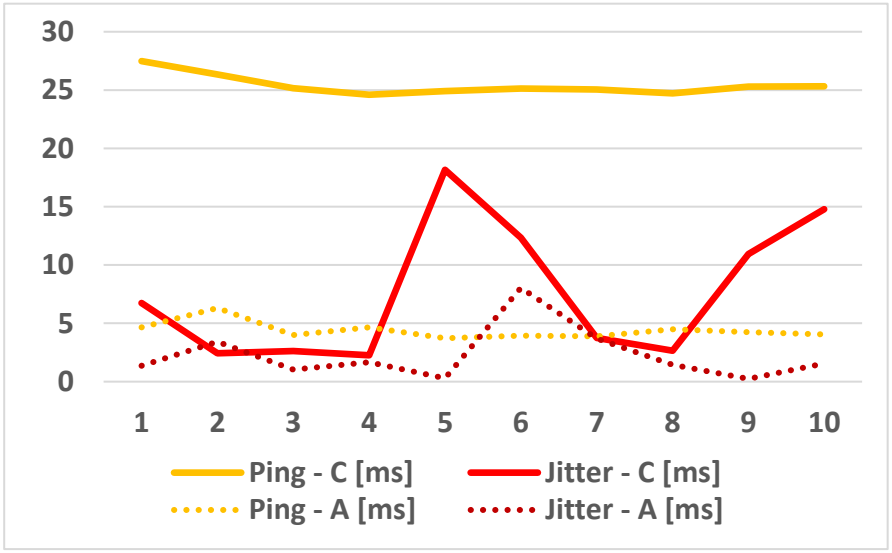
	Ping [ms]	Jitter [ms]	Download [Mbps]	Upload [Mbps]
1.	27,48	6,74	22,05	18,50
2.	26,33	2,44	24,87	18,61
3.	25,16	2,62	24,32	18,70
4.	24,60	2,26	24,69	18,74
5.	24,92	18,16	23,11	18,72
6.	25,13	12,33	24,70	18,76
7.	25,06	3,76	23,74	18,77
8.	24,72	2,66	24,63	18,74
9.	25,29	10,93	24,78	18,84
10.	25,32	14,79	23,51	18,84
Průměr:	**25,40**	**7,67**	**24,04**	**18,72**

Zdroj: Vlastní

U všech hodnot lze opět pozorovat značné zhoršení. Naměřené hodnoty ping na počítačové stanici „C“ se konstantě pohybují kolem 25,4 ms. Na počítačové stanici „A“ je ovšem průměrný ping pouhých 4,39 ms. Výsledky měřeného jitter se opět pohybují ve větším rozmezí od 2,26 ms až po 18,16 ms. Průměr z těchto deseti měření je 7,67 ms, což ve výsledku oproti naměřeným 2,27 ms na počítačové stanici „A“ není tak velký rozdíl. Největší propad je opět zjevný na hodnotách download a upload. Průměr naměřených hodnot download na počítačové stanici „C“ je 24,04 Mbps a rychlost uploadu se vždy pohybuje kolem stabilních 18,72 Mbps. Na počítačové stanici jsou ovšem tyto průměrné hodnoty

94,82 Mbps a 84,71 Mbps. Na Grafu 3 je zobrazen rozdíl hodnot ping a jitter oproti stejnému testu prováděného na počítačové stanici “A“.

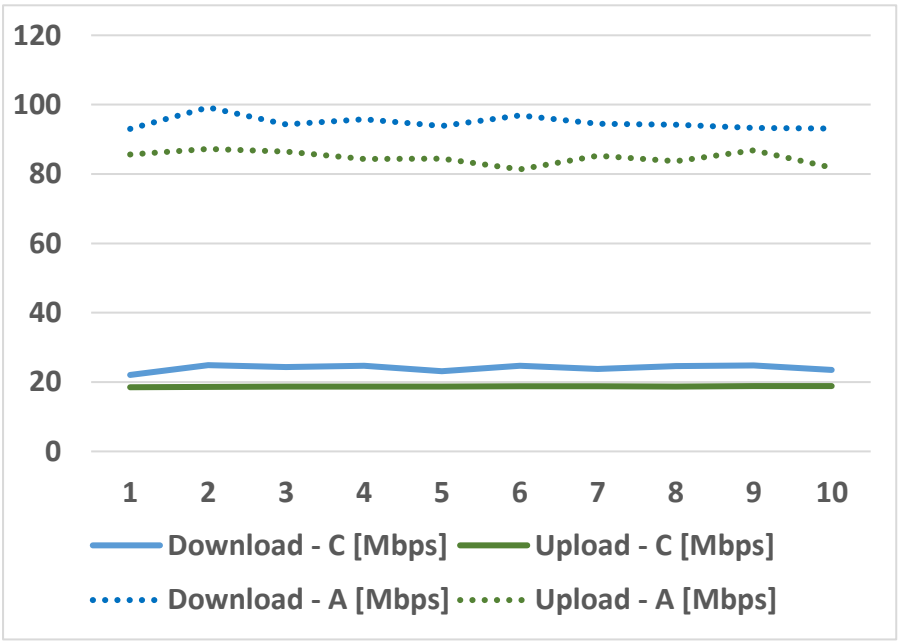
Graf 3 – L2TP – PC:A / PC:C – Ping, Jitter



Zdroj: Vlastní

Tradičně jsou plnou čarou zobrazeny naměřené výsledky z počítačové stanice „C“ a tečkovanou čarou naměřené výsledky z počítačové stanice „A“. Na následujícím Grafu 4, jsou ve stejném smyslu zobrazeny hodnoty download a upload naměřené na počítačové stanici „A“ a „C“.

Graf 4 – L2TP – PC:A / PC:C – Download, Upload



Zdroj: Vlastní

5.8.5. Záznam přenášených paketů

Záznam z přenosu L2TP je doložen v Příloze IV. Vzhledem k využitému IPsec šifrování je záznam komunikace v podstatě stejný jako u samotného IPsecu. L2TP/IPsec pracuje v Hlavním módu (Main Mode), takže vyjednání první fáze pomocí protokolu ISAKMP je tvořeno celkem 6 pakety (šesticestný handshake) a vyjednání druhé fáze (Quick Mode) se opět skládá ze tří paketů. Následná komunikace je šifrována a ukryta pod protokolem ESP. Odpojení VPN je v závěru vyjednáno třemi pakety protokolu ISAKMP.

5.9. Zhodnocení naměřených výsledků

V této kapitole jsou navzájem porovnány naměřené výsledky všech tří testovaných VPN. V první podkapitole jsou zde rozebrány naměřené výsledky přenosové rychlosti TCP a UDP paketů, ztrátovosti UDP paketů a odezvy mezi počítačovou stanicí „A“ a „C“. V druhé podkapitole jsou porovnány naměřené výsledky druhého dodatečného testu, který je zaměřen na měření odezvy, jitter, downloadu a uploadu z počítačové stanice „C“ do Internetu.

Dodatečné zabezpečení pomocí implementace subbnettigu neboli rozdělení počítačových stanic do podsítí tak, aby počítačová stanice „C“ po navázání VPN tunelu měla vždy přístup k počítačové stanici „A“ a naopak neměla přístup k počítačové stanici „B“, byla u všech tří konfigurovaných VPN úspěšně implementována. Po navázání PPTP, IPsec nebo L2TP tunelu má tedy počítačová stanice „C“ přístup pouze k zařízením v první podsíti. Tato konfigurace je u PPTP a L2TP naprosto identická. Adaptéru počítačové stanice „C“ je přidělena IP adresa z přednastaveného IP poolu, jehož rozsah je limitován na rozsah první podsítě. Tato realizace v případě IPsecu je oproti tomu značně odlišná. VPN klient TheGreenBow VPN neumožňuje přidělit adaptéru počítačové stanice „C“ IP adresu ze stejného rozsahu jako je vzdálená podsíť. Ovšem umožňuje nastavit rozsah přístupu do vzdálené podsítě. Pomocí vhodně nastavené IP adresy a masky sítě lze efektivně omezit, kam má ve vzdálené síti počítačová stanice „C“ přístup a kam ne.

5.9.1. Porovnání výsledků mezi PC:A a PC:C

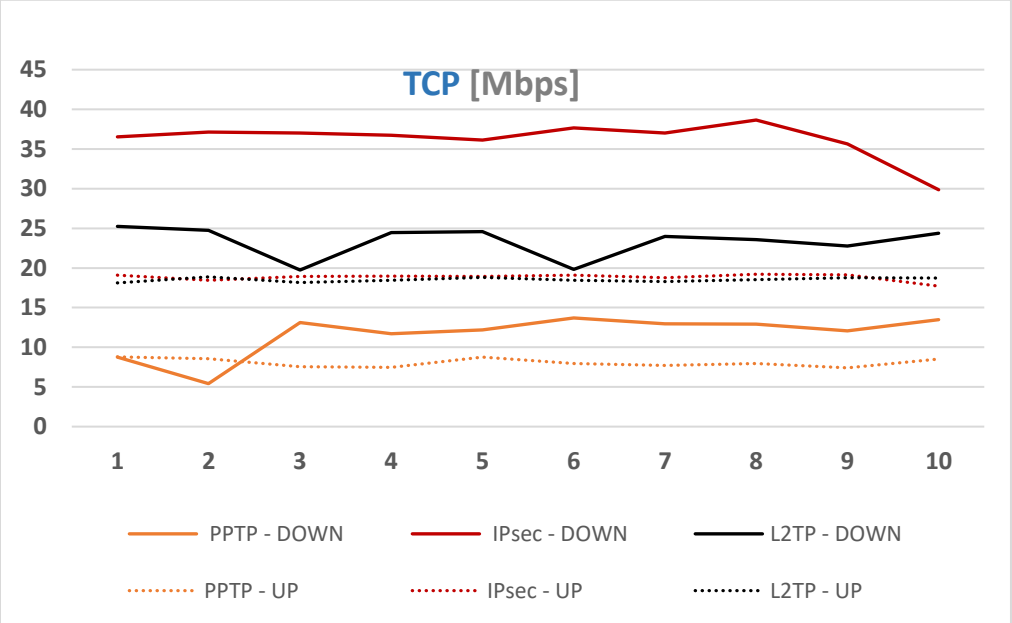
Jednotlivé naměřené hodnoty po navázání PPTP, IPsec a L2TP tunelu jsou navzájem porovnány a výsledky jsou vždy zobrazeny v příslušných grafech. Důležité je zdůraznit, že aplikace TamoSoft počítá pouze s přenesenými daty paketu (režijní data v hlavičce,

nebo záhlaví paketu nebere v potaz). V každém grafu je oranžovou barvou prezentován protokol PPTP, červenou barvou IPsec a černou barvou L2TP.

- **TCP Upload/Download:**

  Jako nejrychlejší se při tomto testu ukázal protokol IPsec. Průměrná rychlost uploadu dosahovala 18,83 Mbps a download se vyšplhal dokonce až na 36,23 Mbps. Druhý nejlépe dopadl protokol L2TP. Průměrná rychlost uploadu z deseti prováděných měření je takřka stejná jako u IPsecu, a to sice 18,52 Mbps, avšak downlaod nedopadl tak dobře. Průměrná rychlost vyšla pouze 23,33 Mbps. Nejhůře v tomto testu bez diskuse dopadl PPTP. Průměrná rychlost uploadu vyšla pouhých 8,07 Mbps a downloadu 11,64 Mbps. V následujícím Grafu 5 jsou zobrazeny naměřené výsledky z deseti prováděných měření. Tečkovanou čarou jsou zobrazeny hodnoty upload a plnou čarou hodnoty download.

Graf 5 – VPN – PC:A / PC:C – TCP Upload / Download



Zdroj: Vlastní

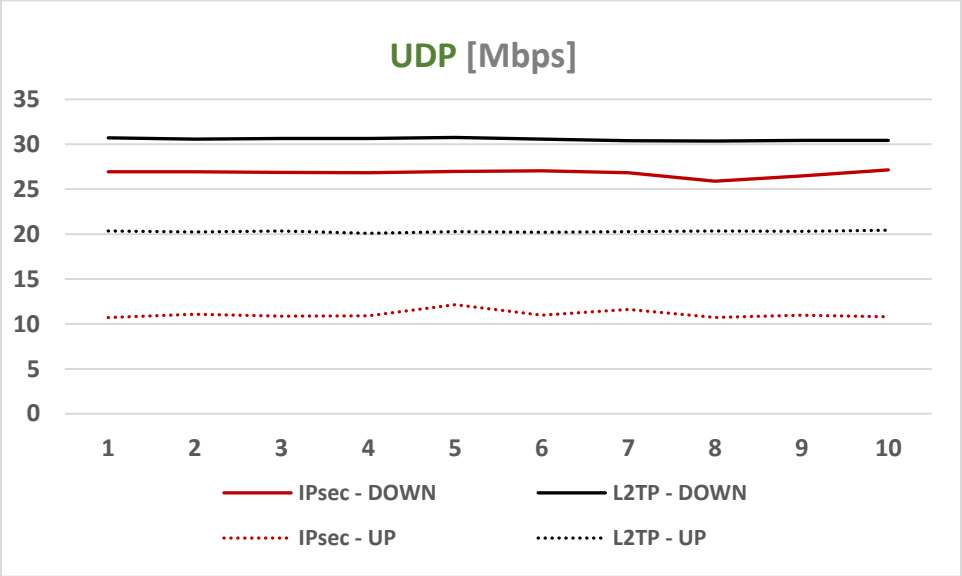
Protokol TCP zaručuje doručení každého odeslaného paketu. To znamená, že měření ztrátovosti paketů zde samozřejmě nemá smysl.

- **UDP Upload/Download, ztrátovost paketů:**

Z testu rychlosti uploadu a downloadu UDP paketů musel být protokol PPTP vynechán, protože neumožňuje přenášet takové množství datových paketů tohoto typu skrz PPTP tunel.

V porovnání rychlosti IPsecu a L2TP uploadu a downloadu UDP paketů dopadl lépe protokol L2TP. Průměrná rychlost uploadu vyšla 20,29 Mbps a downloadu 30,56 Mbps. U IPsecu vyšla průměrná rychlost uploadu pouhých 11,08 Mbps. Download dopadl podstatně lépe a průměrná rychlost dosahuje 26,79 Mbps. Naměřené výsledky z deseti prováděných testů jsou graficky zobrazeny v Grafu 6. Tečkovanou čarou jsou opět zobrazeny hodnoty uploadu a plnou čarou hodnoty downloadu.

Graf 6 – VPN – PC:A / PC:C – UDP Upload / Download

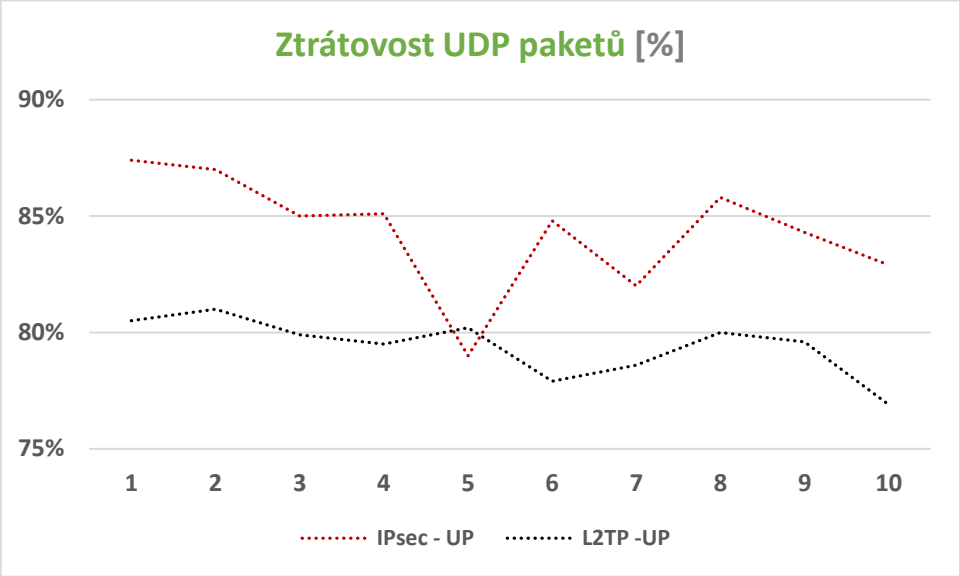


Zdroj: Vlastní

Protokol UDP oproti TCP nezaručuje doručení každého paketu. Ztráta UDP paketů se počítá jako procento dat, která byla ztracena během přenosu. Pokud tedy bude například UDP Down 60,00 Mbps se ztrátou paketů 40,0 %, tak během posledního testovacího cyklu server odeslal 1 megabit dat za 10 milisekund, ovšem klient za těchto 10 milisekund obdržel pouze 0,6 megabitů, zatímco 0,4 megabitů bylo na cestě ztraceno.

Průměrná ztrátovost při uploadu UDP paketů z deseti prováděných měření je u L2TP 79,41 % a u IPsecu o něco vyšších 84,33 %. Takto vysoká ztrátovost paketů je způsobena providerem na straně VPN serveru. T-mobile v tomto případě k odesílání dat využívá radiovou síť, která ztrátovost podstatně zvyšuje. Při downloadu je naopak ztrátovost v podstatě nulová. Ztrátovost UDP paketů napříč upload testem je zobrazena v Grafu 7.

Graf 7 – VPN – PC:A / PC:C – UDP Upload – Ztrátovost paketů

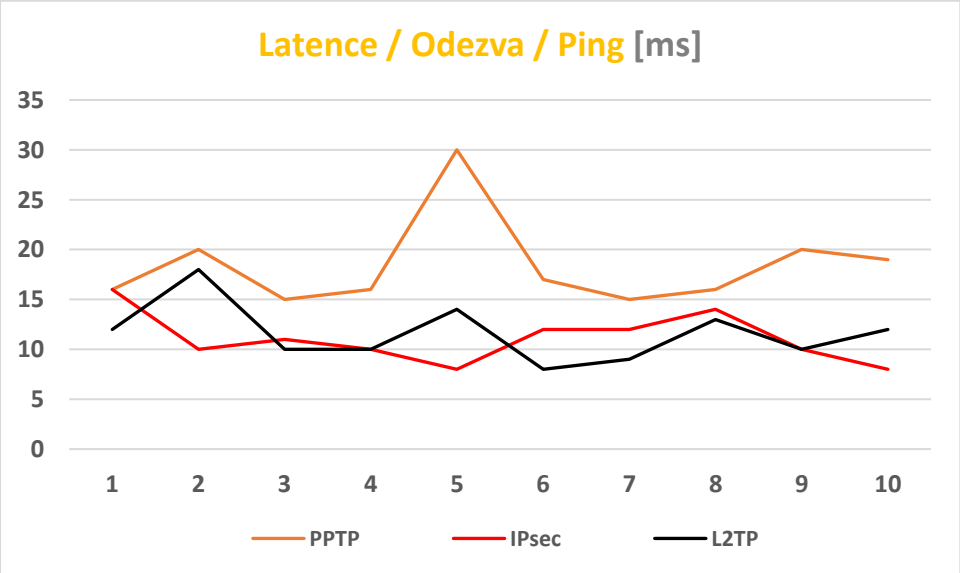


Zdroj: Vlastní

- **Latence:**

Latence neboli ping je hodnota velice proměnlivá a náchylná na současné zatížení sítě. Výsledky je proto třeba brát s určitou rezervou. Je důležité připomenout, že oproti rychlosti downloadu a uploadu je brána nejlepší hodnota latence jako ta nejnižší. V případě tohoto testu vyšel nejlépe IPsec s průměrnou latencí z deseti měření 11,1 ms. Jako druhý nejlepší se s minimálním rozdílem projevil L2TP. Průměrná latence v tomto případě byla 11,6 ms. Nejhůře se zde ukázal opět protokol PPTP s průměrnou latencí 18,4 ms. Výsledky měření jsou zobrazeny v Grafu 8.

Graf 8 – VPN – PC:A / PC:C – Latence / Odezva / Ping



Zdroj: Vlastní

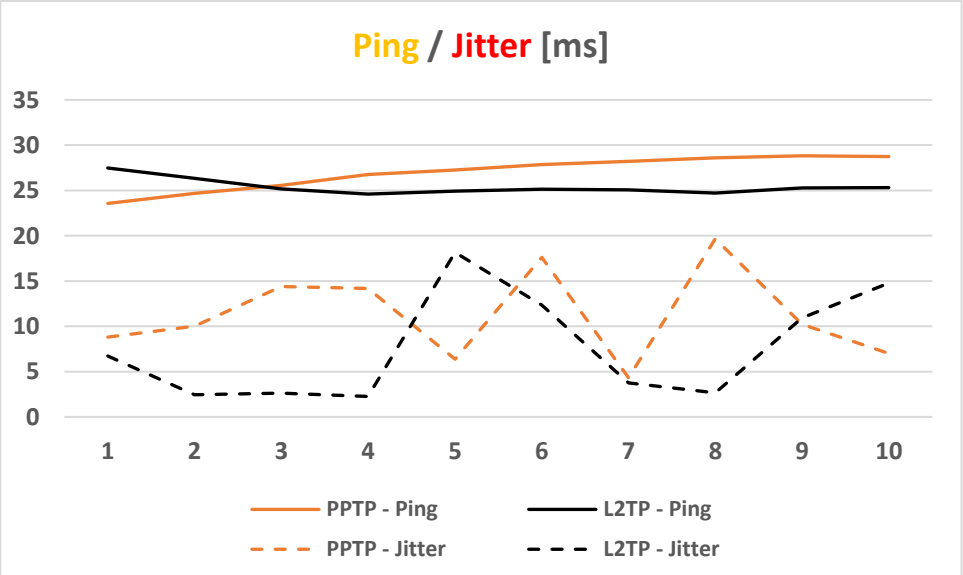
5.9.2. Porovnání výsledků z PC:C do Internetu

Z tohoto testu musí být kompletně vynechán protokol IPsec. Routeru ER6020v1 v takovéto konfiguraci neumožňuje režim „no split tunneling“. Měření zde tedy nemá smysl, protože veškerá komunikace z počítačové stanice „C“ do Internetu je směrována přes lokální UPC router a nikoliv přes vzdálený VPN server. Porovnány jsou zde tedy pouze protokoly PPTP a L2TP. V každém grafu je stejně jako v předchozí podkapitole oranžovou barvou prezentován protokol PPTP a černou barvou L2TP.

- **Ping a jitter:**

V případě tohoto testu jsou výsledky latence (ping) u PPTP i L2TP takřka totožné. Avšak průměrná latence 25,4 ms protokolu L2TP je o něco lepší než 27,01 ms protokolu PPTP. Hodnoty jitter v obou případech kolísaly ve značném rozsahu. Ovšem při pohledu na průměrné hodnoty se opět lépe projevuje protokol L2TP s 7,67 ms. V případě PPTP je průměrná hodnota 11,26 ms. Všechny naměřené hodnoty ping (plná čára) a jitter (čárkovaná čára) jsou zobrazeny v Grafu 9.

Graf 9 – VPN – PC:C – Ping, Jitter



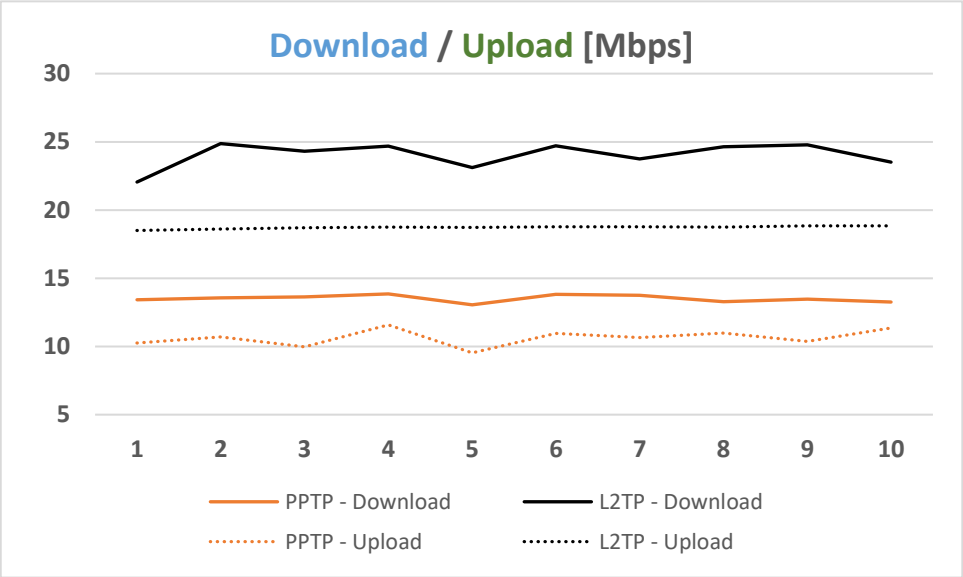
Zdroj: Vlastní

- **Download a upload:**

V testu rychlosti uploadu a downloadu vyšel zcela beze sporu mnohem lépe L2TP. Jeho průměrný download z deseti prováděných měření byl 24,04 Mbps a upload 18,72 Mbps. Průměrný download v případě protokolu PPTP vyšel 13,51 Mbps a upload

10,64 Mbps. Naměřené výsledky jsou zobrazeny v následujícím Grafu 10. Plnou čarou jsou opět zobrazeny hodnoty download a tečkovanou čarou hodnoty upload.

Graf 10 – VPN – PC:C – Download, Upload



Zdroj: Vlastní

5.9.3. Přehled výsledků

Níže v Tabulce 11 je navíc doplněn kompletní přehled všech prováděných testů a jejich výsledků. VPN protokoly jsou zde navzájem porovnávány primárně podle vzešlých naměřených průměrných hodnot.

- **PC:A / PC:C** – Test mezi počítačovou stanicí „A“ a „C“ (TamoSoft, CMD)
- **PC:C / Internet** – Test z počítačové stanice „C“ do Internetu (Cesnet)

Tabulka 11 – VPN – Přehled výsledků

Prováděný test		1.	Průměrný výsledek	2.	Průměrný výsledek	3.	Průměrný výsledek
PC:A / PC:C	TCP - Upload	IPsec	18,83 Mbps	L2TP	18,53 Mbps	PPTP	8,07 Mbps
	TCP - Download	IPsec	36,23 Mbps	L2TP	23,33 Mbps	PPTP	11,64 Mbps
	UDP - Upload	L2TP	20,29 Mbps	IPsec	11,08 Mbps	/	/
	UDP - Download	L2TP	30,56 Mbps	IPsec	26,79 Mbps	/	/
	UDP - Lost packets	L2TP	79,41%	IPsec	84,33%	/	/
	Ping	IPsec	11,1 ms	L2TP	11,6 ms	PPTP	18,4 ms
PC:C / Internet	Upload	L2TP	24,04 Mbps	PPTP	13,51 Mbps	/	/
	Download	L2TP	18,72 Mbps	PPTP	10,64 Mbps	/	/
	Ping	L2TP	24,4 ms	PPTP	27,01 ms	/	/
	Jitter	L2TP	7,67 ms	PPTP	11,26 ms	/	/

Zdroj: Vlastní

6. Závěr

Výsledky provedených testů byly značně ovlivněny nejen využitým softwarem, ale i poměrně limitujícím hardwarem. Z naměřených výsledků by se těžko jednoznačně vybíral nejlepší protokol, ale s jistotou lze vybrat nejhorší. Protokol PPTP skončil v každém testu na posledním místě. Je nutné připomenout, že protokol PPTP dnes už nelze ani z bezpečnostního hlediska považovat za zcela bezpečný. Lze tedy konstatovat, že z pohledu bezpečnosti i rychlosti, by se před protokolem PPTP měly upřednostňovat protokoly jako IPsec nebo L2TP. Jedinou výhodou protokolu PPTP se ukázala jednoduchost konfigurace. Nevyžaduje žádné další dodatečné úpravy na vzdálené počítačové stanici a díky podpoře Windows není třeba ani žádný klientský VPN software. Tuto výhodu bohužel nesdílí protokol IPsec. Jeho konfiguraci lze považovat oproti PPTP nebo L2TP za podstatně náročnější. Samotný VPN protokol IPsec Windows nepodporuje. K připojení je tedy nutné využít klientský VPN software třetích stran. Tyto softwary značně ovlivňují možnosti konfigurace a k jejich používání se ve většině případů musí předplácet časově omezená licence (v případě The GreenBow VPN se jedná o částku 58€ na jeden rok). IPsec v současnosti podporuje všechny nejlepší (veřejné) kryptografické i autentizační protokoly, a jeho zabezpečení je do dnešních dnů považováno za spolehlivé. V prováděných testech přenosové rychlosti a latence dopadl podobně jako L2TP. IPsec se lépe projevil v rychlosti přeposílání TCP paketů a celkové latence. O něco horší výsledky lze pozorovat v případě přenosové rychlosti a ztrátovosti UDP paketů. Tam naopak dopadl lépe L2TP. L2TP tak do jisté míry představuje kombinaci nejlepších vlastností předchozích dvou VPN protokolů. Konfigurace je velice obdobná jako u PPTP a díky podpoře Windows se lze na vzdálené počítačové stanici připojit přes Windows rozhraní bez nutnosti instalace klientského VPN softwaru. O flexibilitě konfigurace a samotného L2TP svědčí i fakt, že ho bylo možné přizpůsobit každému prováděnému testu a z žádného tak nemusel být vynechán. K šifrování a autentizaci komunikace využívá bezpečný protokol IPsec, díky čemu o jeho kvalitě zabezpečení není pochyb.

Součástí zadání byla i implementace subbnettigu neboli rozdělení počítačových stanic do podsítí tak, aby vzdálená počítačová stanice po navázání VPN tunelu měla přístup pouze k vybraným zařízením v lokální síti VPN serveru. Toto dodatečné zabezpečení je u všech tří konfigurovaných VPN úspěšně implementováno.

REFERENCE

1. Peterka, Jiří. Referenční model ISO/OSI. *eArchiv.cz.* [Online] 1996. [Citace: 12. Září 2020.] https://www.earchiv.cz/anovinky/ai1552.php3.
2. Sochor, Tomáš. *Počítačové sítě II.* Ostrava : Ostravská univerzita v Ostravě, 2009.
3. Peterka, Jiří. Síťový model TCP/IP. *eArchiv.cz.* [Online] 1992. [Citace: 3. únor 2019.] http://www.earchiv.cz/a92/a231c110.php3.
4. —. Protokol UDP. *eArchiv.* [Online] 1993. [Citace: 2. Prosinec 2020.] https://www.earchiv.cz/a93/a303c110.php3.
5. Protokol UDP 1.část. *builder.cz.* [Online] 3. Únor 2003. [Citace: 2. Prosinec 2020.] https://www.builder.cz/rubriky/c/c--/protokol-udp-1-cast-156226cz.
6. Comer, Douglas E. *Computer Networks and Internets.* místo neznámé : Pearson; 6th Edition (January 12, 2014), 2014. 978-0133587937.
7. Bouška, Petr. Co je network a subnet (síť a podsíť). *SAMURAJ-cz.* [Online] 5. září 2007. [Citace: 25. Listopad 2018.] https://www.samuraj-cz.com/clanek/tcpip-adresy-masky-subnety-a-vypocty/.
8. Peterka, Jiří. Brouter, gateway. *eArchvi.cz.* [Online] 1993. [Citace: 26. Listopad 2018.] http://www.earchiv.cz/a93/a343c120.php3.
9. Votruba, Zdeněk. Online záznam přednášky. *Adresace sítě – AV záznam z 2.přednášky.* Praha : autor neznámý, 2018.
10. Vozňák, Daniel. Jak funguje a k čemu slouží DHCP. *eABM.* [Online] 26. leden 2017. [Citace: 14. únor 2019.] http://blog.eabm.cz/jak-funguje-a-k-cemu-slouzi-dhcp/.
11. Allan, Grazini Rick a Johnson. *Routing Protocols and Concepts.* místo neznámé : Cisco Press, 2007. 1587132729.
12. Petr, Bouška. Cisco routing - směrovací protokoly . *Samuraj.cz.* [Online] 20. Březen 2009. [Citace: 30. Říjen 2020.] https://www.samuraj-cz.com/clanek/cisco-routing-1-obecne-vlastnosti-smerovacich-protokolu/.
13. Maggio, Alessandro. ICTShore. *Routing concepts: How to read the Routing Table .* [Online] 16. Březen 2019. [Citace: 24. Říjen 2020.] https://www.ictshore.com/free-ccna-course/routing-table-fundamentals/.
14. Kára, Michal. Tuneluji, tuneluješ, tunelujeme: jak a k čemu . *Root.cz.* [Online] 10. Červenec 2003. [Citace: 29. Leden 2021.] https://www.root.cz/clanky/tuneluji-tunelujes-tunelujeme-jak-a-k-cemu/.
15. *Journal of Research of the National Institute of Standards and Technology. .* Westlund, Harold B. 2002.
16. Durčák, Pavel. NaPočítači.cz. *Symetrické a asymetrické šifrování.* [Online] 18. Září 2018. [Citace: 14. Listopad 2020.] https://www.napocitaci.cz/33/symetricke-a-asymetricke-sifrovani-uniqueidgOkE4NvrWuNY54vrLeM677jX7sp3Lu-ZpLpGVMy1prA/https://cs.wikipedia.org/wiki/Asymetrick%C3%A1_kryptografie.
17. O., Alanazi Hamdan. Triple DES Encryption. *IBM Knowlledge Center.* [Online] IBM, 17. Květen 2010. [Citace: 15. Březen 2021.] https://www.ibm.com/support/knowledgecenter/SSLTBW_2.1.0/com.ibm.zos.v2r1.csfb400/gloss.htm#T.
18. Morris J. Dworkin, Elaine B. Barker, James R. Nechvatal, James Foti, Lawrence E. Bassham, E. Roback, James F. Dray Jr. Advanced Encryption Standard (AES). *nist.cz.* [Online] 26. Listopad 2001. [Citace: 23. Březen 2021.] https://www.nist.gov/publications/advanced-encryption-standard-aes.
19. Lake, Josh. What is AES encryption and how does it work? *comparitech.com.* [Online] 17. Únor 2020. [Citace: 23. Březen 2021.] https://www.comparitech.com/blog/information-security/what-is-aes-encryption/.

20. Sandeshi, Sachithi & Priyanjana, W & Sajindra, Hirushan & Bandara, B. RSA in Communication. *ResearcheGate.* [Online] 1. Září 2020. [Citace: 13. Březen 2021.] https://www.researchgate.net/publication/346770905_RSA_in_Communication.
21. [MS-PTPT]: Point-to-Point Tunneling Protocol (PPTP) Profile. *Microsoft.com.* [Online] Microsoft, 2020. Října 30. [Citace: 20. Leden 2021.] https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-ptpt/32e8cf6d-2e0d-4843-8dc0-a4934e16e1f5.
22. Point-to-Point Protocol (PPP). *Cisco.com.* [Online] Cisco Systems. [Citace: 22. Leden 2021.] https://www.cisco.com/c/en/us/tech/wan/point-to-point-protocol-ppp/index.html.
23. Chapter: Implementing Generic Routing Encapsulation . *Cisco.com.* [Online] Cisco Systems. [Citace: 23. Leden 2021.] https://www.cisco.com/c/en/us/td/docs/routers/asr9000/software/asr9k-r6-2/lxvpn/configuration/guide/b-l3vpn-cg-asr9000-62x/m-l3vpn-implementing-generic-routing-encapsulation.html.
24. [MS-CHAP]: Extensible Authentication Protocol Method for Microsoft Challenge Handshake Authentication Protocol (CHAP). *Microsoft.com.* [Online] Microsoft, 14. Únor 2019. [Citace: 24. Leden 2021.] https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-chap/8fea1dd1-66d6-4874-88a5-34bcdbb58907.
25. G. Pall, G. Zorn. Microsoft Point-To-Point Encryption (MPPE) Protocol. *IETF.* [Online] Březen 2001. [Citace: 12. Březen 2021.] https://tools.ietf.org/html/rfc3078.
26. Virtual Private Dialup Network (VPDN). *cisco.com.* [Online] Cisco Systems. [Citace: 29. Leden 2021.] https://www.cisco.com/c/en/us/tech/dial-access/virtual-private-dialup-network-vpdn/index.html.
27. IPsec Technical Reference. *Microsoft.* [Online] Microsoft, 10. Říjen 2009. [Citace: 24. Listopad 2020.] https://docs.microsoft.com/en-us/previous-versions/windows/it-pro/windows-server-2003/cc776369(v=ws.10)?redirectedfrom=MSDN.
28. Protokol Authentication Header (AH). *IBM Knowledge Center.* [Online] IBM. [Citace: 26. Listopad 2020.] https://www.ibm.com/support/knowledgecenter/ssw_ibm_i_71/rzaja/rzajaahheader.htm.
29. Protokol ESP (Encapsulating Security Payload). *IBM knowledge center.* [Online] IBM. [Citace: 26. Listopad 2020.] https://www.ibm.com/support/knowledgecenter/cs/ssw_ibm_i_72/rzaja/rzajaesp.htm#synchmaincontent.
30. Rivest, R. The MD5 Message-Digest Algorithm. *ietf.org.* [Online] 1. Duben 1992. [Citace: 20. Březen 2021.] https://tools.ietf.org/html/rfc1321.
31. SSL, Tým podpordy. ssl.com. *Co je kryptografická funkce hash?* [Online] SSL, 10. Listopad 2015. [Citace: 19. Březen 2021.] https://www.ssl.com/cs/Nej%C4%8Dast%C4%9Bj%C5%A1%C3%AD-dotazy/co-je-kryptografick%C3%A1-hashovac%C3%AD-funkce/.
32. Hash Functions. *nist.gov.* [Online] NIST - CSRC, 17. Červen 2020. [Citace: 20. Březen 2021.] https://csrc.nist.gov/projects/hash-functions.
33. Jupiter networks. *IPsec Security Associations Overview.* [Online] Jupiter networks, 24. Září 2020. [Citace: 29. Listopad 2020.] https://www.juniper.net/documentation/en_US/junos/topics/topic-map/ipsec-security-associations-overview.html.
34. Smyre, Bonnie. IKE VPNs. *raxis.com.* [Online] Raxis, 2018. Květen 23. [Citace: 16. Leden 2021.] https://raxis.com/blog/2018/05/23/ike-vpns-supporting-aggressive-mode.
35. [MS-L2TPIE]: Layer 2 Tunneling Protocol (L2TP) IPsec Extensions. *microsoft.com.* [Online] 7. Duben 2021. [Citace: 8. Duben 2021.] https://docs.microsoft.com/en-us/openspecs/windows_protocols/ms-l2tpie/62de2d02-ad01-49fe-996e-f9dae20f9e15.

36. Han, Deleand. Default encryption settings for the Microsoft L2TP/IPSec VPN Client. *microsoft.com.* [Online] 12. Červenec 2020. [Citace: 22. Únor 2021.] https://docs.microsoft.com/en-us/troubleshoot/windows-client/windows-security/default-encryption-settings-for-l2tp-ipsec-vpn-client.
37. Votruba, Zdeněk. Online záznam přednášky. *VLAN – TGT26Z přednáška VLAN.* Praha : autor neznámý, 2018.
38. Peterka, Jiří. CIDR, alias; Classless InterDomain Routing. *eArchiv.cz.* [Online] 1996. [Citace: 12. prosinec 2018.] http://www.earchiv.cz/anovinky/ai1681.php3.
39. Bouška, Petr. Adresy a jejich výpočty v počítačových sítích založených na TCP/IP. *SAMURAJ-cz.* [Online] 21. červenec 2010. [Citace: 26. listopad 2018.] https://www.samuraj-cz.com/clanek/adresovani-v-ip-sitich/.
40. Traversing NATs and NAPTs with UDP-Encapsulated ESP Packets. *microsoft.com.* [Online] 20. Duben 2017. [Citace: 10. Duben 2021.] https://docs.microsoft.com/en-us/windows-hardware/drivers/network/traversing-nats-and-napts-with-udp-encapsulated-esp-packets.
41. Han, Deland. Configure a L2TP/IPsec server behind a NAT-T device. *Microsoft.com.* [Online] 22. Září 2020. [Citace: 30. Březen 2021.] https://docs.microsoft.com/cs-CZ/troubleshoot/windows-server/networking/configure-l2tp-ipsec-server-behind-nat-t-device.
42. Ben, Lutkevich. DMZ (networking) . *searchsecurity.techtarget.com.* [Online] 15. Prosinec 2019. [Citace: 28. Březen 2021.] https://searchsecurity.techtarget.com/definition/DMZ.
43. Daniel, Missler. Building Yourself a DMZ. *danielmiessler.com.* [Online] 17. Prosinec 2019. [Citace: 28. Březen 28.] https://danielmiessler.com/study/dmz/.

PŘÍLOHY

Příloha I – PPTP – Wireshark

Number	Time	Source	Destination	Protocol	Length	Info
873	2.453517	192.168.2.24	89.233.145.246	TCP	66	13520 → 1723 [SYN]…
894	2.481175	89.233.145.246	192.168.2.24	TCP	66	1723 → 13520 [SYN…
895	2.481291	192.168.2.24	89.233.145.246	TCP	54	13520 → 1723 [ACK]…
896	2.481407	192.168.2.24	89.233.145.246	PPTP	210	Start-Control-Conn…
900	2.498728	89.233.145.246	192.168.2.24	PPTP	210	Start-Control-Conn…
901	2.498842	192.168.2.24	89.233.145.246	PPTP	222	Outgoing-Call-Request
911	2.513327	89.233.145.246	192.168.2.24	PPTP	86	Outgoing-Call-Reply
912	2.516839	192.168.2.24	89.233.145.246	PPTP	78	Set-Link-Info
913	2.518894	192.168.2.24	89.233.145.246	PPP LCP	71	Configuration Request
917	2.533769	89.233.145.246	192.168.2.24	TCP	60	1723 → 13520 [ACK]...
945	2.614839	89.233.145.246	192.168.2.24	PPP LCP	69	Configuration Request
946	2.614839	89.233.145.246	192.168.2.24	PPP LCP	65	Configuration Reject
947	2.615031	192.168.2.24	89.233.145.246	PPP LCP	73	Configuration Ack
948	2.615060	192.168.2.24	89.233.145.246	PPP LCP	64	Configuration Request
952	2.633652	89.233.145.246	192.168.2.24	PPP LCP	68	Configuration Ack
953	2.633904	192.168.2.24	89.233.145.246	PPTP	78	Set-Link-Info
955	2.634007	192.168.2.24	89.233.145.246	PPP LCP	72	Identification
959	2.634031	192.168.2.24	89.233.145.246	PPP LCP	81	Identification
960	2.634049	192.168.2.24	89.233.145.246	PPP LCP	74	Identification
963	2.635187	89.233.145.246	192.168.2.24	PPP CHAP	74	Challenge (NAME=…
964	2.635810	192.168.2.24	89.233.145.246	PPP CHAP	120	Response (NAME='PPTPuzivatel', VALUE=…
967	2.650142	89.233.145.246	192.168.2.24	TCP	60	1723 → 13520 [ACK]...
968	2.663957	89.233.145.246	192.168.2.24	PPP CHAP	100	Success (MESSAGE=…
969	2.663957	89.233.145.246	192.168.2.24	PPP IPCP	84	Configuration Request
974	2.667929	192.168.2.24	89.233.145.246	PPP IPCP	84	Configuration Request
975	2.667959	192.168.2.24	89.233.145.246	PPP IPCP	78	Configuration Reject
976	2.668022	192.168.2.24	89.233.145.246	PPP CCP	60	Configuration Nak
983	2.679653	89.233.145.246	192.168.2.24	PPP LCP	74	Protocol Reject
984	2.680365	89.233.145.246	192.168.2.24	PPP CCP	64	Configuration Nak
985	2.680551	192.168.2.24	89.233.145.246	PPP CCP	64	Configuration Request
988	2.685590	89.233.145.246	192.168.2.24	PPP IPCP	64	Configuration Request
989	2.685802	192.168.2.24	89.233.145.246	PPP IPCP	64	Configuration Ack
991	2.686412	89.233.145.246	192.168.2.24	PPP CCP	64	Configuration Request
992	2.686536	192.168.2.24	89.233.145.246	PPP CCP	64	Configuration Ack
995	2.702368	89.233.145.246	192.168.2.24	PPP CCP	64	Configuration Ack
1007	2.726846	192.168.2.24	89.233.145.246	GRE	46	Encapsulated PPP
1355	3.633725	89.233.145.246	192.168.2.24	PPP IPCP	70	Configuration Reject
1356	3.633992	192.168.2.24	89.233.145.246	PPP IPCP	76	Configuration Request
1366	3.652993	89.233.145.246	192.168.2.24	PPP IPCP	76	Configuration Nak

1367	3.653192	192.168.2.24	89.233.145.246	PPP IPCP	76	Configuration Request
1371	3.668244	89.233.145.246	192.168.2.24	PPP IPCP	76	Configuration Ack
1383	3.690194	192.168.2.24	89.233.145.246	PPP Comp	98	Compressed data
1500	3.952794	192.168.2.24	89.233.145.246	GRE	46	Encapsulated PPP
1505	3.968044	192.168.2.24	89.233.145.246	PPP Comp	110	Compressed data
1525	4.010198	89.233.145.246	192.168.2.24	PPTP	70	Echo-Request
1526	4.010335	192.168.2.24	89.233.145.246	PPTP	74	Echo-Reply
1567	4.096440	192.168.2.24	89.233.145.246	PPP Comp	1161	Compressed data
1594	4.111413	192.168.2.24	89.233.145.246	PPP Comp	183	Compressed data
1595	4.111556	89.233.145.246	192.168.2.24	TCP	60	1723 → 13520 [ACK]...
2795	6.037013	192.168.2.24	89.233.145.246	PPP Comp	150	Compressed data
2906	6.170804	192.168.2.24	89.233.145.246	PPTP	78	Set-Link-Info
2907	6.171123	192.168.2.24	89.233.145.246	PPP LCP	66	Termination Request
2926	6.190457	89.233.145.246	192.168.2.24	PPP Comp	90	Compressed data
2927	6.191407	89.233.145.246	192.168.2.24	PPP LCP	60	Termination Ack
2931	6.206101	89.233.145.246	192.168.2.24	TCP	60	1723 → 13520 [ACK]...
2989	6.249417	192.168.2.24	89.233.145.246	PPTP	70	Call-Clear-Request
2997	6.262540	89.233.145.246	192.168.2.24	PPTP	202	Call-Disconnect-Notify
3035	6.316938	192.168.2.24	89.233.145.246	TCP	54	13520 → 1723 [ACK]…
3038	6.329960	89.233.145.246	192.168.2.24	PPTP	70	Stop-Control-Conn…
3039	6.330051	192.168.2.24	89.233.145.246	PPTP	70	Stop-Control-Conn…
3054	6.344422	89.233.145.246	192.168.2.24	TCP	60	1723 → 13520 [FIN...
3057	6.344499	192.168.2.24	89.233.145.246	TCP	54	13520 → 1723 [ACK]…
3571	7.262178	192.168.2.24	89.233.145.246	TCP	54	13520 → 1723 [FIN…
3583	7.281865	89.233.145.246	192.168.2.24	TCP	60	1723 → 13520 [ACK]...

Příloha II – IPsec (Main Mode) – Wireshark

Number	Time	Source	Destination	Protocol	Length	Info
879	3.220055	192.168.2.24	89.233.145.246	ISAKMP	242	Identity Protection (Main Mode)
883	3.240974	89.233.145.246	192.168.2.24	ISAKMP	242	Identity Protection (Main Mode)
884	3.244431	192.168.2.24	89.233.145.246	ISAKMP	278	Identity Protection (Main Mode)
936	3.423484	89.233.145.246	192.168.2.24	ISAKMP	310	Identity Protection (Main Mode)
937	3.427299	192.168.2.24	89.233.145.246	ISAKMP	138	Identity Protection (Main Mode)
948	3.463360	89.233.145.246	192.168.2.24	ISAKMP	106	Identity Protection (Main Mode)
949	3.464413	192.168.2.24	89.233.145.246	ISAKMP	210	Quick Mode
960	3.479644	89.233.145.246	192.168.2.24	ISAKMP	242	Quick Mode
961	3.480265	192.168.2.24	89.233.145.246	ISAKMP	98	Quick Mode
973	3.541732	192.168.2.24	89.233.145.246	ESP	406	ESP (SPI=0x12a9f84d)
996	3.603583	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x12a9f84d)
997	3.603859	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x12a9f84d)

998	3.603995	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x12a9f84d)
1325	4.353316	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x12a9f84d)
1328	4.368247	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x12a9f84d)
1329	4.368405	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x12a9f84d)
1576	5.103550	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x12a9f84d)
1583	5.118445	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x12a9f84d)
1584	5.118713	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x12a9f84d)
1885	5.864491	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x12a9f84d)
1894	5.879501	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x12a9f84d)
2230	6.835334	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
2231	6.835478	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
2413	7.395693	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x12a9f84d)
2479	7.566961	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
2480	7.567322	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
2491	7.596629	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
2492	7.596802	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
2676	8.161477	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x12a9f84d)
2823	8.540024	192.168.2.24	89.233.145.246	ESP	406	ESP (SPI=0x12a9f84d)
2932	8.914872	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x12a9f84d)
2986	9.105574	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
2991	9.106503	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
3012	9.121764	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
3016	9.122884	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
3198	9.669806	192.168.2.24	89.233.145.246	ESP	286	ESP (SPI=0x12a9f84d)
3278	9.856542	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
3279	9.856738	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
3292	9.886067	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
3293	9.886343	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
3325	9.988780	89.233.145.246	192.168.2.24	UDPENCAP	60	NAT-keepalive
3519	10.617508	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
3523	10.617636	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
3534	10.647171	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
3535	10.647441	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x12a9f84d)
3700	11.177751	192.168.2.24	89.233.145.246	ESP	286	ESP (SPI=0x12a9f84d)
3793	11.423527	192.168.2.24	89.233.145.246	ISAKMP	114	Informational
3794	11.423590	192.168.2.24	89.233.145.246	ISAKMP	122	Informational
3798	11.439754	89.233.145.246	192.168.2.24	ISAKMP	114	Informational

Příloha III – IPsec (Aggressive Mode) – Wireshark

Number	Time	Source	Destination	Protocol	Length	Info
1298	4.248172	192.168.2.24	89.233.145.246	ISAKMP	422	Aggressive
1358	4.429866	89.233.145.246	192.168.2.24	ISAKMP	514	Aggressive
1359	4.434771	192.168.2.24	89.233.145.246	ISAKMP	138	Aggressive
1360	4.435297	192.168.2.24	89.233.145.246	ISAKMP	210	Quick Mode

1370	4.455998	89.233.145.246	192.168.2.24	ISAKMP	242	Quick Mode
1372	4.456804	192.168.2.24	89.233.145.246	ISAKMP	98	Quick Mode
1417	4.577112	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
1418	4.577357	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
1419	4.577483	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
1712	5.340328	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
1713	5.340515	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
1714	5.340641	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
2062	6.095349	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
2063	6.095522	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
2064	6.095720	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
2330	6.858681	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
2331	6.858866	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
2332	6.859025	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
2366	6.982397	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2377	6.997531	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2400	7.028622	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2401	7.028881	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2610	7.619136	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
2676	7.742750	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2682	7.757889	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2693	7.787822	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2694	7.788031	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2929	8.378779	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
2964	8.504770	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2973	8.519406	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2985	8.549419	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
1584	5.118713	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
1885	5.864491	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
1894	5.879501	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
2230	6.835334	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2231	6.835478	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2413	7.395693	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
2479	7.566961	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2480	7.567322	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2491	7.596629	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2492	7.596802	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
2676	8.161477	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
2986	8.549610	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x4329f0c1)
3026	8.666456	89.233.145.246	192.168.2.24	UDPENCAP	60	NAT-keepalive
3175	9.132091	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x4329f0c1)
3311	9.475656	192.168.2.24	89.233.145.246	ESP	406	ESP (SPI=0x4329f0c1)
3389	9.758801	192.168.2.24	89.233.145.246	ISAKMP	114	Informational
3390	9.758871	192.168.2.24	89.233.145.246	ISAKMP	122	Informational
3392	9.773633	89.233.145.246	192.168.2.24	ISAKMP	114	Informational

Příloha IV – L2TP – Wireshark

Number	Time	Source	Destination	Protocol	Length	Info
552	1.576508	192.168.2.24	89.233.145.246	ISAKMP	450	Identity Protection (Main Mode)
567	1.596528	89.233.145.246	192.168.2.24	ISAKMP	222	Identity Protection (Main Mode)
569	1.601707	192.168.2.24	89.233.145.246	ISAKMP	430	Identity Protection (Main Mode)
879	2.610844	192.168.2.24	89.233.145.246	ISAKMP	430	Identity Protection (Main Mode)
908	2.721209	89.233.145.246	192.168.2.24	ISAKMP	446	Identity Protection (Main Mode)
909	2.721747	89.233.145.246	192.168.2.24	ISAKMP	446	Identity Protection (Main Mode)
917	2.726089	192.168.2.24	89.233.145.246	ISAKMP	114	Identity Protection (Main Mode)
918	2.740923	89.233.145.246	192.168.2.24	ISAKMP	114	Identity Protection (Main Mode)
920	2.741887	192.168.2.24	89.233.145.246	ISAKMP	482	Quick Mode
927	2.761044	89.233.145.246	192.168.2.24	ISAKMP	250	Quick Mode
928	2.761591	192.168.2.24	89.233.145.246	ISAKMP	106	Quick Mode
929	2.761945	192.168.2.24	89.233.145.246	ESP	190	ESP (SPI=0x0b624f6e)
1238	3.774286	192.168.2.24	89.233.145.246	ESP	190	ESP (SPI=0x0b624f6e)
1240	3.786762	89.233.145.246	192.168.2.24	ESP	198	ESP (SPI=0xb30d5976)
1241	3.787014	192.168.2.24	89.233.145.246	ESP	102	ESP (SPI=0x0b624f6e)
1242	3.787069	192.168.2.24	89.233.145.246	ESP	94	ESP (SPI=0x0b624f6e)
1243	3.787095	192.168.2.24	89.233.145.246	ESP	134	ESP (SPI=0x0b624f6e)
1254	3.803889	89.233.145.246	192.168.2.24	ESP	94	ESP (SPI=0xb30d5976)
1256	3.805789	89.233.145.246	192.168.2.24	ESP	110	ESP (SPI=0xb30d5976)
1257	3.805789	89.233.145.246	192.168.2.24	ESP	94	ESP (SPI=0xb30d5976)
1258	3.805927	192.168.2.24	89.233.145.246	ESP	134	ESP (SPI=0x0b624f6e)
1259	3.812203	192.168.2.24	89.233.145.246	ESP	118	ESP (SPI=0x0b624f6e)
1293	3.918527	89.233.145.246	192.168.2.24	ESP	110	ESP (SPI=0xb30d5976)
1294	3.918527	89.233.145.246	192.168.2.24	ESP	94	ESP (SPI=0xb30d5976)
1295	3.918527	89.233.145.246	192.168.2.24	ESP	102	ESP (SPI=0xb30d5976)
1309	3.931170	89.233.145.246	192.168.2.24	ESP	110	ESP (SPI=0xb30d5976)
2928	7.162370	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x0b624f6e)
2931	7.163091	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x0b624f6e)
2934	7.163710	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x0b624f6e)
2938	7.164357	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x0b624f6e)
2945	7.165419	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x0b624f6e)
2949	7.166045	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x0b624f6e)
2951	7.166118	192.168.2.24	89.233.145.246	ESP	174	ESP (SPI=0x0b624f6e)
2953	7.166870	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x0b624f6e)
2957	7.167550	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x0b624f6e)

2960	7.168307	192.168.2.24	89.233.145.246	ESP	150	ESP (SPI=0x0b624f6e)
3020	7.331747	192.168.2.24	89.233.145.246	ESP	190	ESP (SPI=0x0b624f6e)
3021	7.331795	192.168.2.24	89.233.145.246	ESP	190	ESP (SPI=0x0b624f6e)
3022	7.331966	192.168.2.24	89.233.145.246	ESP	190	ESP (SPI=0x0b624f6e)
3108	7.609960	192.168.2.24	89.233.145.246	ESP	110	ESP (SPI=0x0b624f6e)
3113	7.623695	89.233.145.246	192.168.2.24	ESP	126	ESP (SPI=0xb30d5976)
3115	7.624433	89.233.145.246	192.168.2.24	ESP	94	ESP (SPI=0xb30d5976)
3150	7.678454	192.168.2.24	89.233.145.246	ESP	118	ESP (SPI=0x0b624f6e)
3151	7.681537	192.168.2.24	89.233.145.246	ESP	118	ESP (SPI=0x0b624f6e)
3159	7.706984	89.233.145.246	192.168.2.24	ESP	94	ESP (SPI=0xb30d5976)
3160	7.706984	89.233.145.246	192.168.2.24	ESP	94	ESP (SPI=0xb30d5976)
3166	7.724013	192.168.2.24	89.233.145.246	ISAKMP	122	Informational
3167	7.724102	192.168.2.24	89.233.145.246	ISAKMP	130	Informational
3188	7.746545	89.233.145.246	192.168.2.24	ISAKMP	114	Informational