



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

VYSOKORYCHLOSTNÍ VIRTUÁLNÍ PRIVÁTNÍ SÍŤ

HIGH-SPEED VIRTUAL PRIVATE NETWORKS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

Anna Porubova

VEDOUCÍ PRÁCE

SUPERVISOR

Ing. Lukáš Malina, Ph.D.

BRNO 2017

Bakalářská práce

bakalářský studijní obor **Teleinformatika**
Ústav telekomunikací

Studentka: Anna Porubova

ID: 156223

Ročník: 3

Akademický rok: 2016/17

NÁZEV TÉMATU:

Vysokorychlostní virtuální privátní sítě

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s problematikou virtuálních privátních sítí (VPN). Analyzujte a zhodnoťte jednotlivá řešení VPN. Navrhněte testovací scénáře a vybraná řešení nakonfigurujte a ověřte. Dále porovnejte jejich bezpečnost, funkcionalitu a změřte jejich efektivitu. Navrhněte a experimentálně ověřte zabezpečenou vysokorychlostní VPN. Navrhněte a vypracujte vhodnou laboratorní úlohu pro demonstraci konfigurace zabezpečené VPN pomocí dostupných nástrojů.

DOPORUČENÁ LITERATURA:

[1] Lewis, Mark. Comparing, Designing, and Deploying VPNs, 2006.

[2] Boyles, Tim. CCNA Security Study Guide: Exam 640 - 553. John Wiley and Sons, 2010.

Termín zadání: 1.2.2017

Termín odevzdání: 8.6.2017

Vedoucí práce: Ing. Lukáš Malina, Ph.D.

Konzultant:

doc. Ing. Jiří Mišurec, CSc.
předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Abstrakt

Cílem tohoto semestrálního projektu je seznámit se s problematikou virtuálních privátních sítí (VPN). Popsat typy VPN a jejich výhody. Analyzovat a zhodnotit jednotlivá řešení VPN. V rámci projektů budou navrženy testovací scénáře a jejich konfigurační řešení.

Klíčová slova

VPN tunel, PPTP, OpenVPN, Strongswan, IPsec, iperf, šifrování

Abstract

Purpose of this project is to become familiar with the issue of virtual private networks (VPN). Describe VPN types and their benefits. Analyze and evaluate various VPN solutions. There will be suggested test scenarios and their configuration solutions.

Keywords

VPN tunel, PPTP, OpenVPN, Strongswan, IPsec, iperf, encryption

PORUBOVA, Anna Vysokorychlostní virtuální privátní síť: semestrální projekt. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, Rok.2016/2017. 32 s. Vedoucí práce byl prof. Ing. Lukáš Malina, Ph.D.

Prohlášení

Prohlašuji, že svůj semestrální projekt na téma „Vysokorychlostní virtuální privátní síť“ jsem vypracovala samostatně pod vedením vedoucího semestrálního projektu a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autorka uvedeného semestrálního projektu dále prohlašuji, že v souvislosti s vytvořením tohoto semestrálního projektu jsem neporušila autorská práva třetích osob, zejména jsem nezasáhla nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědoma následků porušení ustanovení S11 a následujících autorského zákona č. 121/2000Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

Brno

.....

(podpis autora)

Poděkování

Ráda bych poděkovala vedoucímu semestrálního projektu panu Ing. Lukášovi Malinovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno

.....

(podpis autora)

Obsah

| | |
|---|----|
| Úvod | 11 |
| 1 POPIS A TYPY VPN | 12 |
| 1.1 Bezpečnost ve VPN..... | 12 |
| 1.1 Typy VPN | 12 |
| 1.2 Popis VPN..... | 13 |
| 1.3 Architektura VPN a výhody VPN..... | 14 |
| 1.4 Výhody VPN..... | 15 |
| 2 VPN PROTOKOLY | 15 |
| 2.1 PPTP..... | 16 |
| 2.2 IPSec | 17 |
| 2.2.1 L2TP | 17 |
| 2.2.2 IPsec/Strongswan..... | 18 |
| 2.3 OpenVPN | 19 |
| 2.4 Porovnání typů VPN | 20 |
| 2.5 Přehled jiných vysokorychlostních VPN..... | 21 |
| 2.5.1 FreeS/WAN..... | 21 |
| 2.5.2 Openswan..... | 22 |
| 2.5.3 Libreswan..... | 22 |
| 3 PRAKTICKÁ ČÁST: INSTALACE VPN | 23 |
| 3.1 Instalace Strongswan se sdíleným klíčem | 23 |
| 3.1.1. Konfigurace serveru Strongswan | 23 |
| 3.1.2 Konfigurace klienta Strongswan | 24 |
| 3.1.3 Aktivace tunelu Strongswan | 25 |
| 3.2 Instalace PPTP | 25 |
| 3.2.1 Konfigurace serveru PPTP..... | 25 |
| 3.2.2 Konfigurace klienta PPTP..... | 27 |
| 3.2.3 Aktivace tunelu PPTP | 27 |
| 3.3 OpenVPN | 28 |
| 3.3.1 Konfigurace serveru OpenVPN | 28 |
| 3.3.3 Aktivace tunelu OpenVPN..... | 30 |
| 4 NASTROJ PRO MĚŘENÍ VÝKONOSTI VPN | 30 |
| 4.1 Výsledky měření základních parametrů..... | 32 |

| | |
|--|----|
| 5 AUTENTIZACE A KHIHOVNY STRONGSWAN..... | 35 |
| 5.1 Praktická část: Implementace šifrovacích IKE metod a ESP zapouzdření | 36 |
| 6 LABORATORNÍ ÚLOHA..... | 40 |
| Závěr | 44 |
| Literatura | 45 |

Seznam obrázků

| | |
|---|----|
| Obr. 1.1: Architektura tunelování | 13 |
| Obr. 1.2: VPN end to end architektura | 14 |
| Obr. 1.3: VPN remote access architektura..... | 14 |
| Obr. 1.4: VPN inter VLAN architektura..... | 15 |
| Obr. 1.5: Zapouzdření PPTP paketů | 16 |
| Obr. 4.1: Utilizace CPU a RAM | 32 |
| Obr. 4.2: Měření propustnosti na server | 33 |
| Obr. 4.3. Měření propustnosti na klientu | 34 |
| Obr. 5.1: Rychlost (download) při metodách šifrování IKE (AES a Camellia) | 37 |
| Obr. 5.2: Rychlost (download) se zapouzdřením ESP (AES a Camellia) | 39 |
| Obr. 6.1: Schéma zapojení..... | 40 |

Seznam tabulek

| | |
|--|----|
| Tab 2.1: Porovnání několika typů VPN..... | 20 |
| Tab. 4.1: Utilizace CPU a RAM | 32 |
| Tab. 4.2: Měření propustnosti na serveru. | 33 |
| Tab. 4.3: Měření propustnosti na klientu. | 34 |
| Tab 5.1: Implementace metod šifrování IKE (AES a Camellia) | 37 |
| Tab 5.2: Implementace zapouzdření ESP (AES a Camellia)..... | 38 |

Úvod

V současné době máme spoustu informací a je třeba zajistit dostupnost těchto informací určeným skupinám uživatelů. V této práci byla popsána technologie VPN (Virtual Private Networks), která je jedním ze způsobů, jak poskytnout bezpečný přístup k určitým datům bez uniknutí jakýchkoliv informací.

Existuje celá řada VPN řešení a každý uživatel si může vybrat pro něho nejvhodnější řešení. Vytvořit VPN tunel lze několika způsoby a ve své práci nabízím základní přehled VPN technologií a jejich vlastností. Uvádím možnosti realizace VPN pomocí vybraného softwaru, který je dostupný pro Linuxové operační systémy. Podle těchto řešení porovnávám některé parametry VPN, dle kterých čtenář může rozhodnout, které řešení využije.

V první kapitole se zabývám informací o VPN obecně a představuji důvody používání VPN a k čemu slouží.

Ve druhé kapitole jsou popsány jednotlivé typy VPN, používané protokoly, princip vytvoření tunelu a zapouzdření paketů.

Další kapitoly jsou zaměřeny na praktickou realizaci vybraných typů VPN pomocí operačního systému CentOS7 a také na technologií měření parametrů sítí. Výsledky jsou představeny v tabulce.

1 POPIS A TYPY VPN

Pod pojmem VPN nebo také virtuální privátní síť, se ukrývá metoda, používaná pro zvýšení bezpečnosti a ochrany soukromí ve veřejných a soukromých sítích, jako je například Wi-Fi Hotspot a Internet. VPN jsou nejčastěji využívány společnostmi pro zlepšení ochrany citlivých dat, avšak také využívání soukromých VPN se stává stále více populární. Privátní síť se vytvoří stanovením virtuálního spojení point-to-point pomocí specializovaných připojení, tzv. virtuálních tunelovacích protokolů nebo šifrováním provozu. VPN nemůže provést online připojení naprosto anonymně, ale může zásadně zlepšit soukromí a bezpečnost.

1.1 Bezpečnost ve VPN

Bezpečnost je hlavním důvodem, proč korporace využívají VPN již mnoho let. Objevuje se stále více jednoduchých metod na zachycení dat cestujících po síti. WiFispoofing, Honeypot attacks a Firesheep jsou tři jednoduché způsoby, jak zachytit potřebnou informaci pohybující se v internetové síti. VPN využívá pokročilé šifrovací protokoly a bezpečné tunelovací techniky šifrování všech online přenosů. Většina zkušených uživatelů PC by dnes pravděpodobně ani nezkoušela připojení k Internetu bez brány firewall a antivirového softwaru. Stále se zvyšující bezpečnostní hrozby a rostoucí závislost dnešních moderních zařízení na internetu dělají z VPN nezbytnou součást bezpečnostních opatření. Vzhledem k tomu, že veškerý provoz je chráněn, tato metoda je preferovaná víc než například použití proxy serveru. Aby nedošlo k zveřejnění soukromých údajů, VPN zpravidla povoluje přístup k datům pouze ověřeným uživatelům.

1.1 Typy VPN

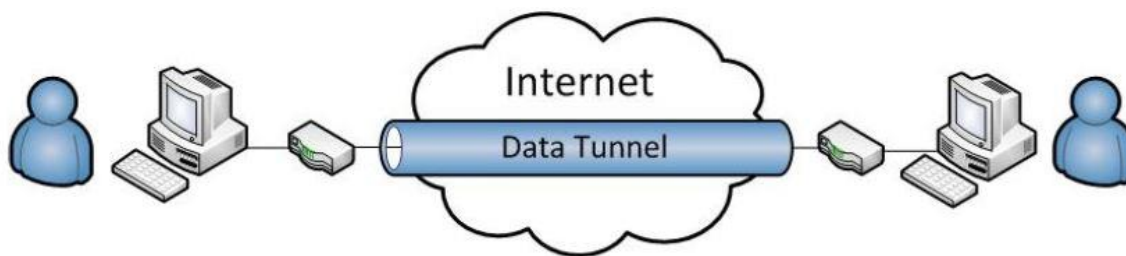
VPN lze vytvořit pomocí hardwaru nebo softwaru. S VPN, které byly vytvořeny pomocí hardwaru, lze dosáhnout vyššího výkonu a zároveň je snadno nastavit. Hardwarové VPN jsou ale naopak omezenější než softwarové VPN, protože nabízí méně konfigurovatelných parametrů. Existuje celá řada produktů velkých výrobců síťových zařízení, které umožňují vytváření hardwarových VPN, například výrobky od společností Cisco, Linksys, Symantec, Nokia atd.

Softwarové VPN mají nižší výkon a složitější konfiguraci oproti těm hardwarovým, ale díky své flexibilní konfiguraci více vyhovují zvyšujícím se požadavkům bezpečnosti. Existují dva způsoby, jak vytvořit softwarovou VPN. První z nich je za použití programů, které obsahují samotný operační systém, jakým je například Windows XP a vyšší, Windows 2003 Server a vyšší, a systémy na platformě GNU/Linux. Tyto sítě však mají tendenci být poněkud omezené. Druhou možností, která je doporučena, jsou specifické aplikace k vytvoření open source sítí VPN jako OpenVPN, OpenSSHVTun, Hamachi, FreeS/WAN atd. Tento typ aplikací nabízí širokou škálu možností individuálních nastavení a lépe tak tedy vyhoví specifickým požadavkům na vytvoření dobře zabezpečené privátní sítě.

1.2 Popis VPN

VPN je síť, která umožňuje komunikaci mezi vzdálenými uživateli nebo komunikaci mezi různými sítěmi daleko od sebe. Tato metoda může propojit dvě nebo víc sítí a vytvořit jedinou privátní síť umožňující komunikaci bod-bod (point to point). Tato komunikace je bezpečně stanovena mezi dvěma extrémami a je zcela transparentní pro všechny uživatele sítě, které existují mezi oběma konci VPN. Z tohoto důvodu, technika se nazývá *tunelování*, viz Obr. 1.1. Datové pakety, které jsou směřované přes veřejnou síť, jako je Internet nebo jakékoli jiné obchodní síť, je soukromý tunel, který simuluje bod-bod připojení.

Technika tunelování zahrnuje zapouzdření síťového protokolu přes jiný protokol, a tím vytváří tunel v počítačové síti. To znamená, že otevře spojení mezi dvěma body (vysílač-přijímač) pomocí zabezpečeného protokolu, jako by mohly být SSL (Secure Socket Layer) a SSH (Secure Shell).

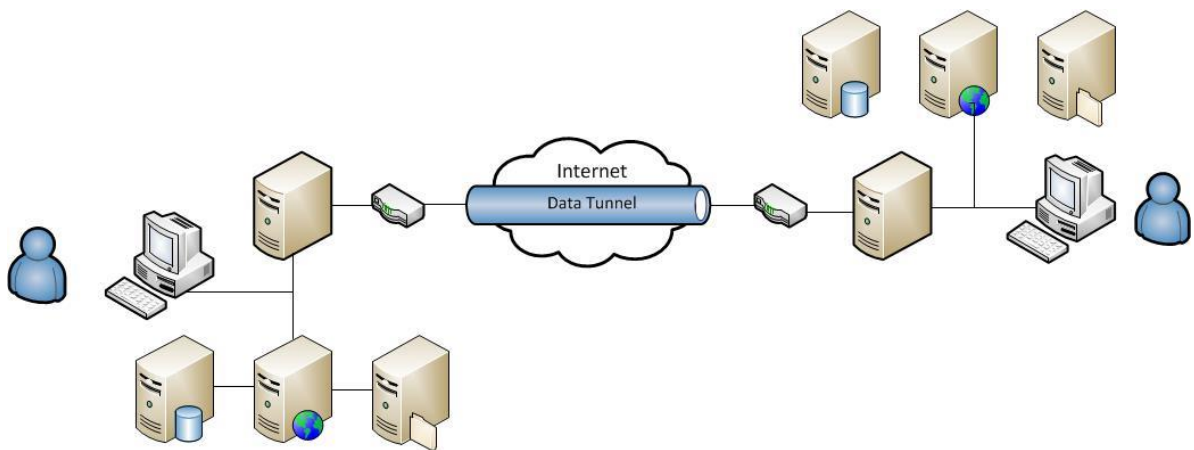


Obr. 1.1: Architektura tunelování

1.3 Architektura VPN a výhody VPN

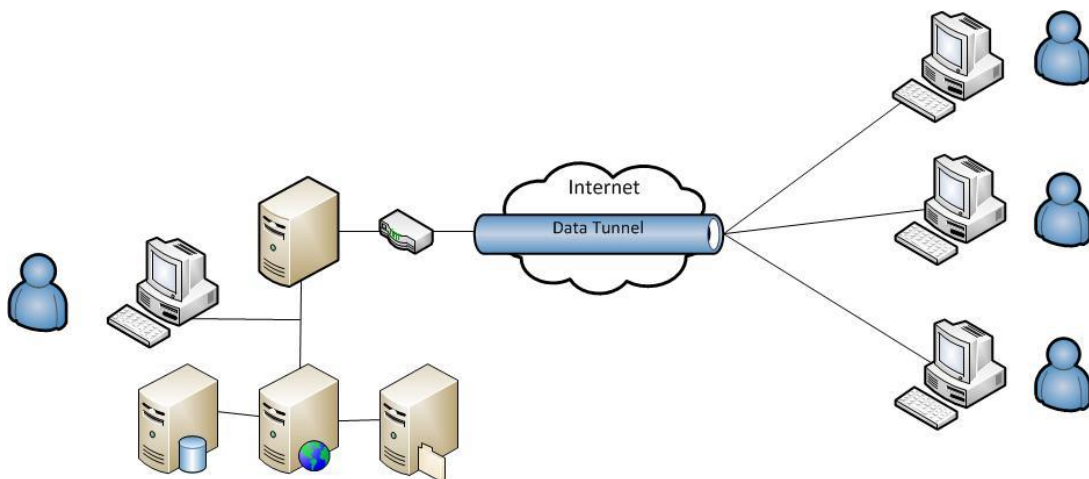
K vytváření VPN se používají tři typy architektur:

- **VPN end-to-end.** V tomto modelu VPN z firemní kanceláře nebo ústředí připojují k centrálnímu bodu, kde je umístěn VPN server, který je zodpovědný za vytváření tunelů a řízení veškerého provozu, viz Obr. 1.2.



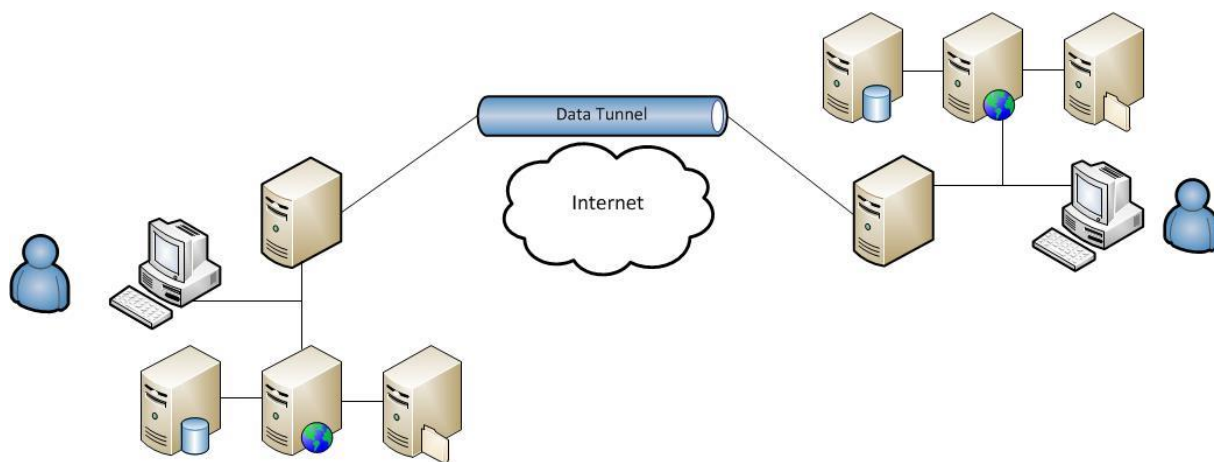
Obr. 1.2: VPN end to end architektura

- **VPN vzdálený přístup (remote access).** Uživatel se připojí k firemní síti ze vzdáleného místa, například z domu, letiště, hotelu apod. VPN remote access poskytuje spojení uživatelů vytvořením zabezpečeného virtuálního tunelu mezi PC nebo jiným zařízením uživatele a serverem VPN, viz Obr. 1.3.



Obr. 1.3: VPN remote access architektura

- **VPN inter VLAN.** Tato architektura umožňuje vytvořit tunel v rámci sítí LAN. V tomto případě VPN server neposkytuje přímé směrování mezi intranetem a veřejnou sítí. To je velmi užitečný způsob, jak izolovat nějaké oblasti nebo síťové prostředky. Pouze ověření uživatelé v intranetu mohou mít přístup k serveru, viz Obr. 1.4.



Obr. 1.4: VPN inter VLAN architektura

1.4 Výhody VPN

Implementace VPN má řadu výhod:

- Integrita dat, odeslané zprávy nelze změnit.
- Důvěrnost, pouze ověření uživatelé mohou přistupovat k informacím z VPN.
- Snadné použití pro nezkušené uživatele, transparentní systém.
- Uspodňuje komunikaci mezi dvěma uživateli umístěných daleko od sebe.

2 VPN PROTOKOLY

Existuje celá řada různých protokolů k vytvoření bezpečných tunelů mezi dvěma sítěmi. Uživatel by měl pečlivě přezkoumat různé možnosti všech VPN před tím, než bude implementovat svoje řešení. Protokoly se používají k zapouzdření a odpouzdrění paketů. Mezi nejpoužívanější VPN protokoly jsou L2TP, IPSec, PPTP a OpenVPN . PPTP je protokol VPN pro zapouzdření paketů prostřednictvím veřejné sítě.

2.1 PPTP

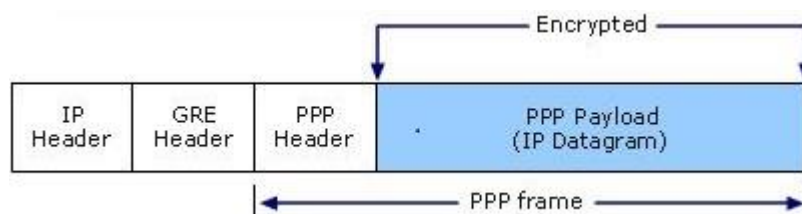
PPTP je protokol, který pracuje na druhé vrstvě OSI modelu, která nese název „datová vrstva“. PPTP je rozšířením Point-to-Point Protokolu (PPP). Protokol vytváří PPP rámce, které obsahují zapouzdřené IP (Internet Protocol) pakety. Důvodem pro to je nutnost zajistit, aby IP pakety mohly být směrovány po internetové síti. K tomu, aby se dalo této problematice správně porozumět, bychom si nejprve měli vysvětlit, jak protokol PPP funguje.

PPP je síťový protokol používaný pro správu spojení klienta a serveru přes vytáčené připojení nebo pomocí sériového připojení point-to-point. Protokol PPP zapouzdří IP paket do rámců PPP. Tyto balíčky pak mohou být použity k vytvoření připojení typu point-to-point mezi vysílajícím a přijímajícím počítačem.

Když klient chce poslat nějaký IP paket přes privátní síť, musí být nejdříve IP paket zapouzdřen do PPP rámce pro vytvoření připojení typu point-to-point k serveru PPTP. Technika zapouzdření PPTP je založena na jiném internetovém standardu, který se jmenuje GRE (Generic Routing Encapsulation protocol), viz Obr. 1.5.

Autentifikační metody, které mohou být použity, jsou Password Authentication Protocol (PAP), Challenge Handshake Authentication Protocol (CHAP) nebo Extensible Authentication Protocol - Transport Layer Security, což je zkratka (EAP-TLS).

Šifrování v PPTP ovládá protokol MPPE (Microsoft Point-to-Point Encryption). Používány šifrovací algoritmus je RC4, který povoluje 40, 56 a 128 bitové klíče.



Obr. 1.5: Zapouzdření PPTP paketů

2.2 IPSec

IP Security je standardní protokol, který pracuje s IP protokolem. Je vytvořený s rozdílnými schopnostmi a protokoly. Hlavní část RFC jasně a do detailu popisuje každý protokol a IPSec využívá a dává IPSec specifikace. Na příklad, IPSec, specifikovaný v RFC 4301, vytváří hranici mezi chráněnou a nechráněnou částí sítě. Pakety proudící skrze tuto hranici jsou zpracovávány odlišně. Pakety v závislosti na tom, jak je IPSec nakonfigurovaný mohou proudit bez překážek, být vyhozeny nebo zpracovány různými službami. Velkou výhodou IPSec je, že pracuje v síťové vrstvě a pracuje s IPv4 i IPv6, což znamená, že všechny existující aplikace mohou využít IPSec bez modifikace.

IPSec se liší od tradičních aplikací jako je SSH, která operuje v aplikační vrstvě. IPSec byl vyvinut, aby zvýšil bezpečnost IP komunikace. IPSec může být použit v point to point připojení mezi dvěma počítači, aby učinil komunikaci bezpečnou. IPSec také umožňuje bezpečnou komunikaci mezi připojeným klientem a VPN serverem. IPSec není protokol, ale spíše soubor protokolů, které pracují s rozličnými protokoly a vykonávají vlastní cíl důvěryhodnosti s integritou a autentizací.

Bod, který je nutné vyjasnit je, že IPSec tunely pomáhají jen zabezpečit unicast provoz, ale nemohou být použity k zabezpečení více nebo široko rozlehlých paket.

IPSec může být rozdělen na dvě části:

- **Security Protocols** jsou protokoly, které definují, jaké informace mohou být přidány k IP paketu, aby dosáhla přesné důvěryhodnosti, integrity a autentizace.
- **Internet Key Exchange (IKE)** je používán k autentizaci dvou zařízení. Ta zařízení si vyměňují tajný klíč k šifrování a dešifrování dat a dohodnou se, jaký protokol bude použit.

2.2.1 L2TP

Layer 2 Tunneling Protocol (L2TP) má svůj původ v PPTP. Vzhledem k tomu, že L2TP neposkytuje bezpečnostní funkce, jako je šifrování nebo silná autentizace, se obvykle L2TP kombinuje s technologií IPsec. Aby se zabránilo příliš vysoké režii, využívá se běžně transportního módu ESP (Encapsulating Security Payload). To znamená, že nejprve

se naváže spojení přes IPsec, opět za použití IKE a pak se tento vytvořený kanál použije pro vytvoření L2TP tunelu. Poté se připojení přes IPsec použije k přepravě L2TP zapouzdřených uživatelských dat. V porovnání s obvyčejnou technologií IPsec s dodatečným zapouzdřením pomocí L2TP (přidává se IP/UDP paket a hlavička L2TP) dělá tuto metodu o něco méně efektivní (tím spíše, pokud je použito ve spojení s ESP v tunelovém režimu, což se u některých implementacích děje). NAT (NAT-T) je také problematictější s L2TP / IPsec díky společnému využívání systému ESP v transportním režimu. Jednou z výhod L2TP oproti obvyčejnému IPsec je ta, že můžeme přepravovat i jiné protokoly než IP protokol. Security-wise jsou si podobné, ale záleží na způsobu ověřování, režim ověřování (hlavní nebo agresivní režim), sílu klíče, použité algoritmy atd.

2.2.2 IPsec/Strongswan

IPsec/Strongswan (dál jen Strongswan) je kompletní řešení založené na IPsec, který poskytuje šifrování a ověřování pro servery a klienty. Může být použitý k zajištění komunikace ve vzdálených sítích, takže připojení na dálku je stejné jako ve spojovací místně. Strongswan je IPsec démon s plnou podporou pro IKEv1 a IKEv2. Je nativně podporovaná většinou moderních klientů, včetně Linuxu, Windows 7, Apple iOS, Mac OS X a BlackBerry OS. NAT funguje automaticky, žádná další konfigurace není potřebná.

Aby bylo zajištěno, že osoba, s kterou je spojení IKE navázáno, je ve skutečnosti ta, kterou tvrdí, že je a že má tedy být ověřena, Strongswan nabízí několik způsobů, jak toho dosáhnout:

- Authentication Public Key
- Pre-Shared-Key (PSK)
- Extensible Authentication Protocol (EAP)
- extended Authentication (XAuth)

Konfigurační soubory používané Strongswan, jsou následující:

- ipsec.conf: umožňují konfiguraci připojení IPsec
- ipsec.secrets: seznam hesel (sdílené klíče, soukromé klíče)
- ipsec.d: ukládá certifikáty a soukromé klíče
- strongswan.conf: umožňuje konfiguraci globálních nastavení

2.3 OpenVPN

OpenVPN je volně dostupný software vytvořený v roce 2001. Je navržený tak, aby vytvářel bezpečné sítě ve vrstvě 2. nebo 3. OSI modelu (Tunel nebo Most). Takže zákazník nemůže vytvořit spojení do VPN serveru skrze webový prohlížeč dokud OpenVPN nepracuje v 7. „aplikační“ vrstvě OSI modelu.

OpenVPN má dvě metody šifrování dat. První metodou je použití před-sdílených statických klíčů, která říká, že všichni uživatelé používají ten stejný klíč k šifrování a dešifrování. Druhá metoda využívá certifikát s SSL/TLS a RSA klíče. Tato druhá metoda, přestože je více komplexní, je tou, která byla vybrána pro implementaci naší VPN z důvodu bezpečnosti. V metodě sdílených klíčů, když někdo dostane klíč, pak se může jednoduše zmocnit dat a nainstalovat si klíč na svůj počítač. Od tohoto okamžiku se může chovat jako uživatel sítě, a tedy rozšifrovat všechny informace. Bezpečný SSL/TLS mód v OpenVPN je cílený na uživatele, kteří budou generovat vlastní certifikáty, a proto jejich vlastní Certifikační Autoritu (CA), která vytváří důvěryhodnou entitu, která je zodpovědná za vydávání a rušení certifikátů. V našem případě, klient a server zprostředkovávají tajný klíč (symetrický), obecně známý jako hlavní tajemství, které používá Diffie-Hellman kryptografický protokol. Ten je klíčovým navázáním protokolu mezi stranami, které doposud neměli žádný předchozí kontakt. V tomto případě má každý klient privátní klíč a uživatelské jméno k uskutečnění autentizace (*.key*). Nyní OpenVPN autentizuje *.key* soubor kontrolou, zda je podepsaný CA.

OpenVPN může být použitý k přenosu protokolu obou UDP a TCP pro spojení mezi body. Když se používá UDP, klient, který vytvoří žádost o spojení, bude čekat na odpověď ze serveru přibližně 5 sekund (standardně), předtím než se pokusí znovu spojit. TCP také používá časovače, které se mohou lišit, vzrůstají v případě ukončení času, aby předešly přetížení sítě. To může způsobit problémy, když dvě vrstvy používají tuto metodu, lze říci, že je pravděpodobné, že ve VPN, TCP paket je zablokovan jinou TCP, kterou může poslat. V tomto případě je problémem, že TCP nižší vrstvy nedostává pakety, které čekají, její časovače budou zvyšovat své vzrůstající řady opakovaných přenosů. Na druhé straně, TCP vyšší vrstvy bude čekat na ACK, s žádnou odezvou, což také zvýší jejich časovače a opakované přenosy. To může eventuálně vést k přetížení sítě. Ale hlavní vadou použití TCP je, že protokol pracuje s maximálním rozsahem paketů. Proto, když se uzavře znovu

vrchními vrstvami, pak může paket přesáhnout rozměry a musí být rozdělen. Tato fragmentace může způsobit, že směrovač nemůže směrovat tyto velké pakety a pak příjemce nemůže poskládat všechny informace dohromady, protože může dojít ke ztrátě paket nebo se mohou nějaké informace opakovat. Kvůli všem uvedeným případům, OpenVPN (standardně) používá UDP, protože to nabízí větší bezpečí proti útokům a dovoluje OpenVPN pracovat více efektivně.

2.4 Porovnání typů VPN

V tabulce můžeme nahlédnout a porovnat vlastnosti jednotlivých typů VPN mezi sebou, viz Tab 2.. [1]

Tab 2.1: Porovnání několika typů VPN.

| | Délka šifrování | Podporované OS | Bezpečnost | Používané porty | Stabilita/ Kompatibilita |
|-------------------|------------------|--|---|---|--|
| PPTP | 128 bits | Windows, Mac OS X, Linux, iOS, Android, DD-WRT | MPPE šifrování s kryptografickým algoritmem PC4 | TCP port 1723 and GRE (Protocol 47). PPTP může být snadno blokován omezením protokolu GRE | Nespolehlivé. Drobné problémy s kompatibilitou u GRE protokolu a některých routerů |
| IPSec | 256 bits | Windows, Mac OS X, Linux, iOS, Android | IKE protokoly, ESP | UDP 500 pro počáteční výměnu klíčů, 50 pro zašifrovaná data IPSEC, UDP 1701 pro konfiguraci, UDP 4500 pro NAT | Docela stabilní a rychlé. Obtížně nakonfigurovat spolehlivou práci mezi zařízeními za NAT routery |
| OpenVPN | 160/256 bits | Windows, Mac OS X, Linux | OpenSSL library | Běží na libovolném portu buď UDP nebo TCP | Stabilní a rychlé přes bezdrátové, mobilní a jiné nespolehlivé sítě, kde ztráty paketů a zahlcení jsou běžné |
| Strongswan | 128/192/256 bits | Linux 2.6, 3.x and 4.x kernels, Android, FreeBSD, OS X and Windows | IKE protokoly, ESP | Porty 4500/UDP, 500/UDP, 51/UDP a 50/UDP | Konfigurace je podobná IPsec. Je také rychle a stabilní |

2.5. Přehled jiných vysokorychlostních VPN

2.5.1 FreeS/WAN

FreeS/WAN (Free SecureWide-Area Networking) je Linuxová implementace IPsec protokolů. Služba IPsec poskytuje služby šifrování a autentizace na úrovni IP (Internet Protocol).

Při práci na této úrovni IPsec může zabezpečit jakýkoliv provoz přenášený přes IP, na rozdíl od jiného šifrování, které obecně chrání pouze určité protokoly - PGP pro poštu, SSH pro vzdálené přihlášení, SSL pro práci na webu a tak dále. Tento přístup má značné výhody i určité omezení.

IPsec lze použít na jakémkoli počítači, který provádí IP networking. IPsec může být instalován všude tam, kde je potřeba k ochraně provozu. Službu IPsec lze také spustit na směrovačích, na firewallech, na různých aplikačních serverech a na stolních nebo přenosných počítačích pro koncové uživatele.

Implementace má tři hlavní části:

- **KLIPS** (Kernel IPsec) implementuje ESP a zpracovává pakety v jádře
- **Pluto** (IPsec démon) implementuje IKE a vyjednává o spojení s jinými systémy
- Různé skripty poskytují rozhraní správce pro stroje

VanillaFreeS/WAN implementuje tyto části specifikací IPsec. Dole jsou některá pravidla implementace:

- Chcete-li používat certifikáty X.509 s FreeS / WAN, budete potřebovat patch X.509 nebo Openswan, která obsahuje tento patch
- Chcete-li použít NAT (Network Address Translation) traversu s FreeS/WAN, budete potřebovat traversový patch NAT společnosti Arkoon Network Security nebo Openswan, která jej obsahuje.

FreeS/WAN vždy navrhuje trojitě DES šifrování a Perfect Forward Secrecy (PFS). Navíc navrhuje skupiny 5 a 2 Diffie Hellman (v tomto pořadí) a MD5 a SHA-1 hashes.

2.5.2 Openswan

Openswan je také implementací IPsec pro Linux, který byl zahájen jako projekt FreeS/WAN, nadále používá *GNU General Public License*¹. Má podporu pro většinu rozšíření (RFC + IETF drafts) související s protokolem IPsec, včetně IKEv2, digitálních certifikátů X.509, NAT Traversal a mnoha dalších.

Konfigurace Openswan je podobná Strongswan. Je třeba opravit konfigurační soubory */etc/ipsec.conf* a */etc/ipsec.secrets*. Spouštění tunelu se dá příkazem „systemctl start openswan.service“.

2.5.3 Libreswan

Libreswan je bezplatná implementace softwaru nejpoužívanějšího a standardizovaného protokolu VPN založeného na IPsec a IKE. Tyto standardy jsou vyráběny a udržovány týmem Internet IETF (EngineeringTaskForce).

V oblasti bezpečnosti počítačů je Libreswan vidlicí implementace VPN OpenswanIPsec, kterou vytvořili téměř všichni vývojáři společnosti Openswan. Libreswan podporuje IKEv1 a IKEv2. Je spuštěn na Linuxu 2.4 až 4.x, FreeBSD a Apple OSX. Na Linuxu může používat vestavěný zásobník IPsec ("XFRM / NETKEY") nebo vlastní zásobník IPsec ("KLIPS"). Používá kryptografickou knihovnu NSS.

VPN StrongSwan a Libreswan pocházejí z projektu FreeS/WAN. Openswan a Libreswan jsou stále mnohem bližší k původu, kde je Strongswan v dnešní době v podstatě kompletní reimplementace. Současná architektura Strongswan byla původně navržena pro IKEv2 téměř před 10 lety, ale od verze 5.x se používá i pro IKEv1. Dodává se s rozšiřitelným, dobře měřítkovým multi-threading designem a zaměřuje se na IKEv2

¹GNU General Public License česky „všeobecná veřejná licence GNU“

3 PRAKTICKÁ ČÁST: INSTALACE VPN.

Implementace VPN a měření jednotlivých parametrů VPN bylo provedeno na virtuálním zdroji VirtualBox 5.1.16, se zadanou operační pamětí RAM 1.5GB. Fyzický zdroj: počítač HP EliteBook 840, Intel(R) Core(TM) i5-5300U CPU @ 2.30GHz, RAM 8.00GB, OS 64-bit.

V praktické části naimplementujeme tři typy VPN – Strongswan, PPTP a OpenVPN. Pro instalaci VPN byl použit linuxový operační systém CentOS7, nainstalovaný na dvou virtuálních strojích. Před instalací jakékoliv VPN se nejdříve musíme přesvědčit, zda je na serveru a klientovi nainstalován EPEL repozitář. EPEL repozitář poskytuje užitečné softwarové balíky, které nejsou zahrnuty v oficiální CentOS. Nainstalovat EPEL repozitář se dá pomocí dalšího příkazu:

```
# yum install epel-release
```

3.1 Instalace Strongswan se sdíleným klíčem

V tunelu se bude sdílet tajný klíč mezi dvěma stroji. Před sdílením klíče pro symetrický šifrovací algoritmus, tyto tajné klíče se využívají Diffie-Hellmanovým algoritmem pro vzájemnou autentizaci. Pro instalaci Strongswanu použijeme příkaz:

```
# yum install strongswan -y
```

3.1.1. Konfigurace serveru Strongswan

Zvláštnost Strongswanu je v tom, že soubor *ipsec.conf* se musí konfigurovat jak na straně serveru, tak i na straně klienta. *Ipsec.conf* je hlavní konfigurační soubor Strongswan. V tomto souboru, definujeme takové parametry politiky tunelu, jako šifrovací algoritmus, hash algoritmus atd. Tento soubor je uložen ve složce */etc/strongswan*.

```
# vi /etc/strongswan/ipsec.conf
    config setup
        charondebug="all"
        uniqueids=yes
        strictcrpolicity=no
conn %default
conn tunnel
    left=207.154.207.79
    leftsubnet=10.1.0.0/16
    right=207.154.207.77
```

```
rightsubnet=11.1.0.0/16
ike=aes256-sha2_256-modp1024!
esp=aes256-sha2_256!
keyingtries=0
ikelifetime=1h
lifetime=8h
dpddelay=30
dpdtimeout=120
dpdaction=clear
authby=secret
keyexchange=ikev2
auto=start
type=tunnel
```

IPsec tajemství (sdílené klíče, hesla soukromého klíče, PIN pro odemknutí HSM) jsou uloženy v souboru *ipsec.secrets*. Jak je uvedeno níže, sdílené tajemství mezi oběma stranami VPN je "sharedsecret".

```
# vi /etc/strongswan/ipsec.secrets
207.154.207.79 07.154.207.77: PSK 'sharedsecret'
```

3.1.2 Konfigurace klienta Strongswan

Konfigurace klienta je podobná, jenom je třeba správně nastavit IP adresy *left* i *right* na obou stranách:

```
# vi /etc/strongswan/ipsec.conf
    config setup
        charondebug="all"
        uniqueids=yes
        strictcrpolicypolicy=no
conn %default
conn tunnel
    left=207.154.207.77
    leftsubnet=11.1.0.0/16
    right=207.154.207.79
    rightsubnet=10.1.0.0/16
    ike=aes256-sha2_256-modp1024!
    esp=aes256-sha2_256!
    keyingtries=0
    ikelifetime=1h
    dpdtimeout=120
    dpdaction=clear
    authby=secret
    keyexchange=ikev2
    auto=start
    type=tunnel
```

Soubor *ipsec.secrets* obsahuje sdílený tajný klíč na vzdálené straně:

```
# vi /etc/strongswan/ipsec.secrets
207.154.207.79      07.154.207.77: PSK 'sharedsecret'
```

3.1.3 Aktivace tunelu Strongswan

Po změnách tunel musíme aktivovat na obou stranách pomocí příkazu:

```
# systemctl start strongswan
```

Ověříme status Strongswan tunelu:

```
# ip xfrm state
src 207.154.207.77 dst 207.154.207.79
    proto esp spi 0xce151e64 reqid 1 mode tunnel
    replay-window 32 flag af-unspec
    auth-trunc hmac(sha256)
0x696f257370540a91a236c6a85c6fe8893af25ff18999b4604db499f46084c8c8 128
    enc cbc(aes)
0xc0859d5c50768fbdf22be3abfa039fca26b54b836f8f892a5eaf4c8f3e2ae30a
src 207.154.207.79 dst 207.154.207.77
    proto esp spi 0xcd4029fc reqid 1 mode tunnel
    replay-window 32 flag af-unspec
    auth-trunc hmac(sha256)
0x759a267b151dbc7e655125661db44f8f2e5a75ed13929d98386da8a94870e831 128
    enc cbc(aes)
0x1bd9d21da19ff9191aca112280f128a181a012689270708b22df8c0dc9cad1bf
```

3.2 Instalace PPTP

Point-to-Point Tunneling Protocol (PPTP) umožňuje velmi rychle implementovat vlastní VPN, a je kompatibilní s většinou mobilních zařízení. I když PPTP je méně bezpečné než OpenVPN, je také rychlejší a využívá méně CPU.

3.2.1 Konfigurace serveru PPTP

Vybereme jeden server, který bude odpovědný za manipulaci s IP adresami a ověření všech serverů v síti VPN. Nainstalujeme PPTP pomocí příkazu:

```
# yum -y install ppp pptpd
```

U PPTP budeme upravovat konfigurační soubor */etc/pptpd.conf*. Kde *localip* je IP adresa serveru a *remoteip* jsou IP adresy, které budou přiřazeny klientům, kteří se připojují k němu.

```
# vi /etc/pptpd.conf
option /etc/ppp/options.pptpd
logwtmp
localip 10.0.10.1
remoteip 10.0.10.2-254
```

Další nastavení specifikujeme v souboru */etc/ppp/options.pptpd*:

```
# vi /etc/ppp/options.pptpd
name pptpd
refuse-pap
refuse-chap
refuse-mschap
refuse-eap
proxyarp
lock
nobsdcomp
novj
novjccomp
nologfd
noauth
ms-dns 8.8.8.8
ms-dns 8.8.4.4
```

Dále bychom měli nastavit autentizaci pro PPTP přidáním uživatele a hesla. Uložíme to do souboru */etc/ppp/chap-secrets*:

```
# vi /etc/ppp/chap-secrets
username pptpd password123 *
```

Vložíme řádek *net.ipv4.ip_forward=1* do souboru */etc/sysctl.conf*, tím povolíme IPv4 směrování, poté změny uložíme:

```
# vi /etc/sysctl.conf
net.ipv4.ip_forward=1

# sysctl -p
```

Vytvoříme NAT pravidla pro iptables:

```
# chmod +x /etc/rc.d/rc.local
```



```
# echo "iptables -t nat -A POSTROUTING -s 10.0.10.0/24 -o eth0 -j MASQUERADE"
>> /etc/rc.d/rc.local
# iptables -t nat -A POSTROUTING -s 10.0.10.0/24 -o eth0 -j MASQUERADE
```

3.2.2 Konfigurace klienta PPTP

Po instalaci PPTP vytvoříme nový soubor `/etc/ppp/peers/server` a přidáme následující řádky, jméno a heslo narazíme vlastními hodnotami:

```
# yum install pptp -y
# echo "username pptd password123 *" >> /etc/ppp/chap-secrets
# vi /etc/ppp/peers/server
pty "pptp 139.59.155.23 --nolaunchpppd"
name username
password password123
remotename PPTP
file /etc/ppp/options.pptp
ipparam server
```

3.2.3 Aktivace tunelu PPTP

Spouštění PPTP na serveru:

```
# systemctl start pptpd
# systemctl enable pptpd.service
```

Výzva serveru na klientu:

```
# pppd call server
```

Ve výstupu `ifconfig` klienta vidíme, že tunel se vytvořil:

```
# ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 138.68.106.147 netmask 255.255.240.0 broadcast 138.68.111.255
    inet6 fe80::ec2b:86ff:fec8:7611 prefixlen 64 scopeid 0x20<link>
    ether ee:2b:86:c8:76:11 txqueuelen 1000 (Ethernet)
    RX packets 8894 bytes 10162193 (9.6 MiB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 5530 bytes 584911 (571.2 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
```

```
loop txqueuelen 0 (Local Loopback)
RX packets 8 bytes 576 (576.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 8 bytes 576 (576.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

ppp0: flags=4305<UP,POINTOPOINT,RUNNING,NOARP,MULTICAST> mtu 1500
inet 10.0.10.2 netmask 255.255.255.255 destination 10.0.10.1
ppp txqueuelen 3 (Point-to-Point Protocol)
RX packets 7 bytes 60 (60.0 B)
RX errors 0 dropped 0 overruns 0 frame 0
TX packets 7 bytes 66 (66.0 B)
TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

3.3 OpenVPN

OpenVPN je aplikace k vybudování sdílené privátní sítě, kterou lze snadno nainstalovat a nakonfigurovat na serveru. Jedná se o řešení pro ty, kteří potřebují bezpečné síťové připojení přes veřejný internet. Příkaz pro instalaci OpenVPN:

```
yum -y install openvpn easy-rsa iptables-services
```

3.3.1 Konfigurace serveru OpenVPN

Nejdřív musíme vytvořit klíče a certifikáty:

- Certifikační autority (CA)
- Klíčů a certifikát serveru
- Klíč Diffie-Hellman
- Klíč a certifikát klienta

Když klíče a certifikáty jsou vytvořené, upravíme soubor *server.conf*:

```
# vi /etc/openvpn/server.conf
port 1337
proto udp
dev tun
ca /etc/openvpn/keys/ca.crt
cert /etc/openvpn/keys/server.crt
key /etc/openvpn/keys/server.key
dh /etc/openvpn/keys/dh1024.pem
server 192.168.200.0 255.255.255.0
push "redirect-gateway def1"
```

```
push "dhcp-option DNS 8.8.8.8"
push "dhcp-option DNS 8.8.4.4"
duplicate-cn
keepalive 20 60
comp-lzo
persist-key
persist-tun
daemon
log-append /var/log/myvpn/openvpn.log
verb 3
```

Vypínání firewallu:

```
# systemctl mask firewalld
# systemctl stop firewalld
```

Vypínání SELinux tím, že změníme SELINUX na *disabled*:

```
# vim /etc/sysconfig/selinux
SELINUX=disabled
```

Konfigurace směrování a Iptables:

```
# systemctl enable iptables
# systemctl start iptables
# iptables -F

# iptables -t nat -A POSTROUTING -s 192.168.200.024 -o eth0 -j MASQUERADE
# iptables-save > /etc/sysconfig/iptablesvpn
```

3.3.2 Nastavení klienta OpenVPN

Pro připojení k OpenVPN serveru, klient vyžaduje klíč a certifikát, který jsme již vytvořili, stáhneme tři soubory ze serveru pomocí SFTP nebo SCP: *ca.crt*, *client.crt*, *client.key*. Poté na klientu vytvoříme nový soubor s názvem *client.ovpn* a do toho vložíme konfiguraci:

```
client
dev tun
proto udp
#Server IP and Port
remote 192.168.1.104 1337
```

```
resolv-retry infinite
nobind
persist-key
persist-tun
mute-replay-warnings
ca ca.crt
cert client.crt
key client.key
ns-cert-type server
comp-lzo
```

3.3.3 Aktivace tunelu OpenVPN

Nastartujeme OpenVPN na serveru:

```
# systemctl -f enable openvpn@server.service
# systemctl start openvpn@server.service
```

Potom povolíme OpenVPN na klientu:

```
# openvpn --config client.ovpn
```

4 NASTROJ PRO MĚŘENÍ VÝKONOSTI VPN

Pro měření výkonu jednotlivých VPN byl použit jednoduchý nástroj *iperf*. Měření probíhá tak, že *iperf* je spuštěn na jedné straně na serveru a na druhé na klientu. Nástroj umožňuje generovat provoz pro analýzu propustnosti sítě. Výhodou je, že existuje *iperf* pro systémy Windows i Linux:

Instalace *iperf* na CentOS:

```
# yum install iperf
```

Spouštění *iperf* na serveru:

```
# iperf -s
```

Nastavení *iperf* na straně klienta dáme ve formátu *iperf -c <adresa iperf serveru> -d*, takovým způsobem rychlost změříme duplexně (tedy download/upload) současně.

```
# iperf -c 192.168.0.103 -d
```

Příklad výstupu iperf příkazu:

```
[server]# iperf -s
-----
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[ 4] local 139.59.155.23 port 5001 connected with 138.68.106.147 port 58160
-----
Client connecting to 138.68.106.147, TCP port 5001
TCP window size: 502 KByte (default)
-----
[ 6] local 139.59.155.23 port 42944 connected with 138.68.106.147 port 5001
[ ID] Interval      Transfer    Bandwidth
[ 6]  0.0-10.0 sec  1.32 GBytes  1.13 Gbits/sec
[ 4]  0.0-10.0 sec   730 MBytes  610 Mbits/sec
[ 5] local 10.0.10.1 port 5001 connected with 10.0.10.2 port 53274
-----
Client connecting to 10.0.10.2, TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[ 6] local 10.0.10.1 port 57216 connected with 10.0.10.2 port 5001

[client]# iperf -c 10.0.10.1 -d
-----
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
-----
-----
Client connecting to 10.0.10.1, TCP port 5001
TCP window size: 85.0 KByte (default)
-----
[ 3] local 10.0.10.2 port 53274 connected with 10.0.10.1 port 5001
[ 5] local 10.0.10.2 port 5001 connected with 10.0.10.1 port 57216
[ ID] Interval      Transfer    Bandwidth
[ 3]  0.0-10.0 sec   13.6 MBytes  11.4 Mbits/sec
[ 5]  0.0-10.0 sec   23.1 MBytes  19.4 Mbits/sec
```

Iperf podporuje ladění různých parametrů týkajících se časování, vyrovnávacích pamětí a protokolů (TCP, UDP, SCTP s IPv4 a IPv6). Pro každý test uvádí šířku pásma, ztráty a další parametry. Kromě toho se dá změnit velikost balíčku a nastavit port pro source a destinaci. Podporuje různé operační systémy: Windows, Linux, Android, MacOS X, FreeBSD, OpenBSD, NetBSD, VxWorks, Solaris a další.

4.1 Výsledky měření základních parametrů

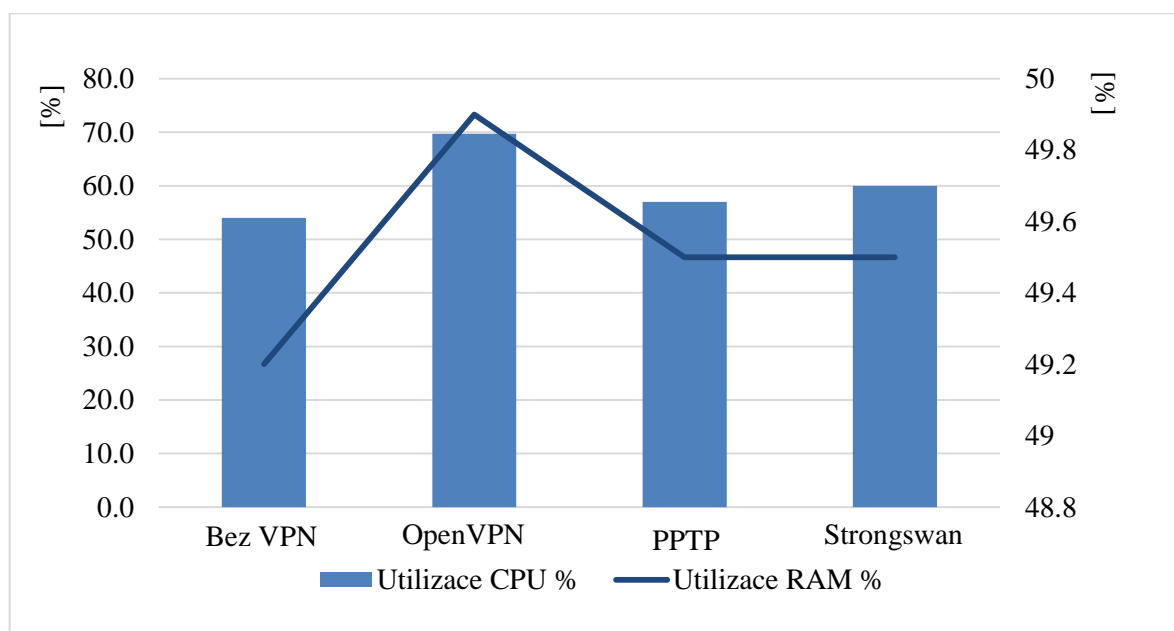
Ve výsledcích porovnáme utilizace CPU a RAM, propustnost VPN, viz Tab. 4.1, utilizace CPU a RAM, měření propustnosti na serveru a na klientu, viz Tab. 4.2 a Tab. 4.3.

Tab. 4.1: Utilizace CPU a RAM

| | Utilizace CPU | Utilizace RAM |
|------------------|---------------|---------------|
| | % | % |
| Bez VPN | 54,0 | 49,2 |
| Strongswan/IPsec | 60,0 | 49,4 |
| PPTP | 57,0 | 49,5 |
| OpenVPN | 69,7 | 49,9 |

Z grafu na Obr. 4.1 Obr. 4.2 je vidět, kolik procent CPU a RAM využívá systém bez VPN. Také vidíme, že utilizace RAM je skoro stejná pro všechny VPN, proto porovnáme utilizaci CPU. Během testování bylo potvrzeno, že nejmenší využití RAM je při PPTP, jak se očekávalo. Na druhém místě v grafu je Strongswan. Utilizace CPU nepřeskočila za 60%.

Nejhorší výsledek je pro OpenVPN, využití CPU se rovná kolem 70%, což při spouštění dalších aplikací může silně přetížít systém.



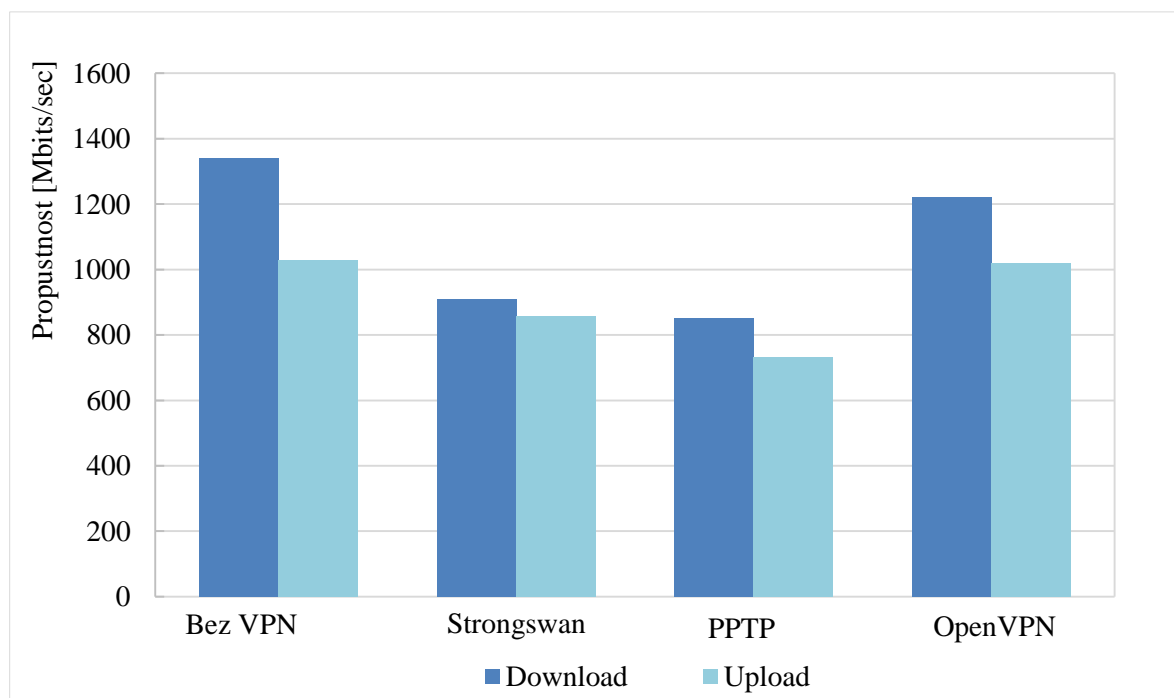
Obr. 4.1: Utilizace CPU a RAM

Tab. 4.2: Měření propustnosti na serveru.

| | U-Upload, D-Download | Přenos paketů | Propustnost |
|------------|-------------------------|---------------|-------------|
| | [-] | [Mbits] | [Mbits/sec] |
| Bez VPN | D | 12480 | 1340 |
| | D | 9680 | 1030 |
| Strongswan | D | 8000 | 861 |
| | U | 8480 | 907 |
| PPTP | D | 7000 | 732 |
| | U | 8104 | 845 |
| OpenVPN | D | 9520 | 1020 |
| | U | 11360 | 1210 |

Přenos v tabulce Tab. 4.2 byla převedená z MBytes na Mbits. Je třeba si všimnout, že v grafu nebyl uvážen přenos paketů, protože to nemá vliv na propustnost tunelu.

Z testu je vidět, že nejrychlejší VPN je OpenVPN. Avšak kvůli vysoké utilizaci CPU (jejíž velikost víme z předchozího testu), dáme přednost Strongswanu. Strongswan je pomalejší kvůli dvojnásobnému zapouzdření dat. Nejpomalejší ze třech VPN je PPTP.



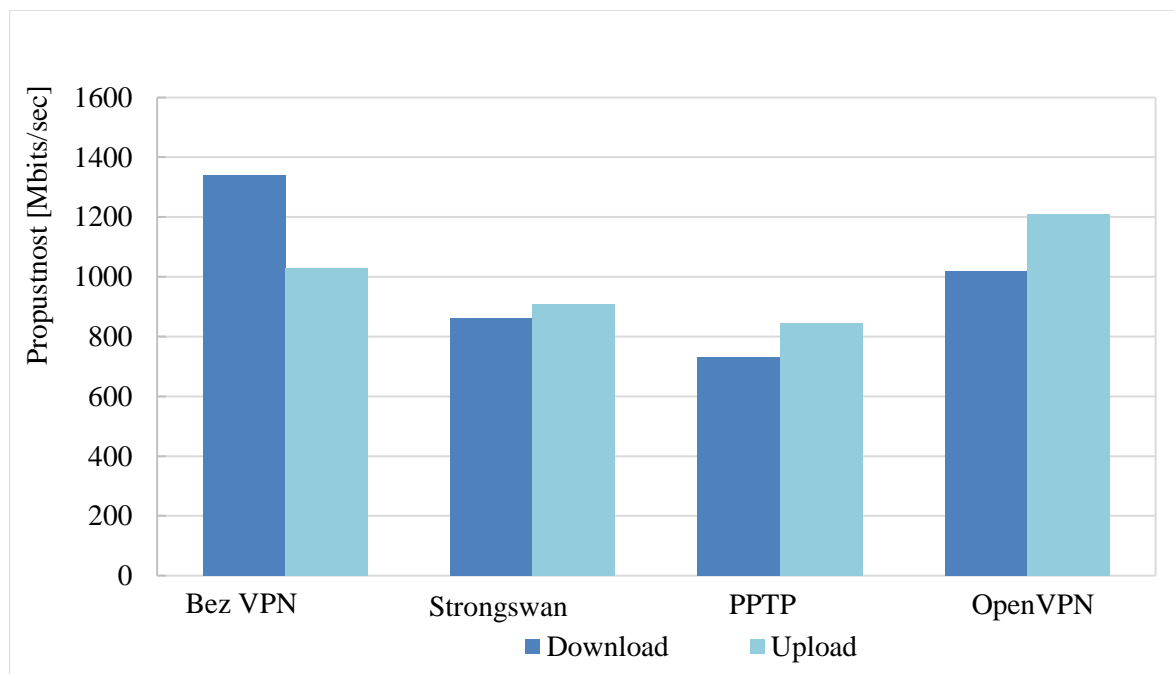
Obr. 4.2: Měření propustnosti na server

Tab. 4.3: Měření propustnosti na klientu.

| | U-Upload, D-Download | Přenos paketů | Propustnost |
|------------|-------------------------|---------------|-------------|
| | [-] | [Mbits] | [Mbits/sec] |
| Bez VPN | D | 12480 | 1340 |
| | U | 9680 | 1030 |
| Strongswan | D | 8480 | 909 |
| | U | 8000 | 858 |
| PPTP | D | 8104 | 850 |
| | U | 7000 | 731 |
| OpenVPN | D | 11360 | 1220 |
| | U | 9520 | 1020 |

Na grafu Obr. 4.3 jsou podobné výsledky s grafem Obr. 4.2, protože měření na serveru a na klientu probíhalo současně obousměrně pomocí nástroje *iperf*.

Pro zhodnocení, nejrychlejší VPN v tomto testu je OpenVPN, nejpomalejší je PPTP. Střední výsledek je u Strongswan, kterým se budeme dále zabývat.



Obr. 4.3. Měření propustnosti na klientu

5 AUTENTIZACE A KNIHOVNY STRONGSWAN

Podíváme se blíže na vlastnosti a parametry Strongswan. Pro tento typ VPN se dá nastavit spousta parametrů dle potřeby uživatele. Začneme z toho, jaké klíče existují. Soubor */etc/ipsec.secrets* obsahuje neomezený počet následujících typů klíčů:

- **RSA** definuje RSA privátní klíč
- **ECDSA** definuje ECDSA privátní klíč
- **BLISS** definuje BLISS privátní klíč (od 5.2.2)
- **P12** definuje PKCS#12 kontejner (od 5.1.0)
- **PSK** definuje sdílený klíč
- **EAP** definuje EAP ověřovací údaje
- **NLTM** definuje NTLM ověřovací údaje
- **XAUTH** definuje XAUTH ověřovací údaje
- **PIN** definuje smartcard PIN

V našem případě byl použit PSK klíč. Autentizace pomocí sdíleného klíče vyžaduje, aby oba systémy mohli stanovit identické tajemství (tajemství není přeneseno IKE protokolem). Autentizace pomocí veřejného klíče, jako je RSA, vyžaduje, aby každý host měl vlastní privátní klíč. Host z rozumných důvodů může používat různé privátní klíče pro různé rozhraní a různé klienty.

Jednou z nejdůležitějších věcí ve VPN je šifrování. Strongswan má docela velké krypto knihovny, které si zaslouží naši pozornost, to jsou:

- **GMP** (GNU MultiplePrecisionArithmeticLibrary)
- **Gcrypt** (nebo Libgcrypt)
- **OpenSSL**

GMP je bezplatná knihovna pro aritmetiku s libovolnou přesností, pracující s celými a racionálními čísly a čísly s pohyblivou řádovou čárkou. Neexistuje žádný praktický limit na přesnost, kromě dostupné paměti zdroje, na kterém běží GMP. Hlavními cílovými aplikacemi GMP jsou kryptografické aplikace a výzkum, aplikace pro zabezpečení Internetu, algebraické systémy, výpočetní výzkumní algebra atd.

GMP je navržena tak, aby byla co nejrychlejší pro malé a velké operandy. Platformy GMP jsou Unix-type systémy, jako GNU/Linux, Solaris, HP-UX, Mac OS X/Darwin, BSD, AIX, etc. Také může pracovat na Windows 32/64 bit.

Gcrypt (Libgcrypt) je kryptografická knihovna vyvinutá jako samostatný modul GnuPG. Může být použita samostatně od GnuPG, ale záleží na jeho error-reporting knihovně. Libgcrypt se skládá z několika dílčích knihoven, které implementují vlastní algoritmy. Funkce zahrnuje symetrické šifrování, šifrování s veřejným klíčem a dohoda o klíče, zpracování certifikátů, kryptografické hashové funkce a kryptografický generátor pseudonáhodných čísel.

OpenSSL je softwarová knihovna, která se používá pro zabezpečení komunikace přes počítačové sítě. OpenSSL má široký rozsah kryptografických algoritmů používaných v různých internetových standardech. Služby poskytované touto knihovnou se používají pro implementace SSL, TLS, S/MIME, SSH, OpenPGP a jiné kryptografické standardy. Jsou dostupné verze pro Unix systémy (Solaris, Linux, macOS, QNX), OpenVMS a pro Microsoft Windows.

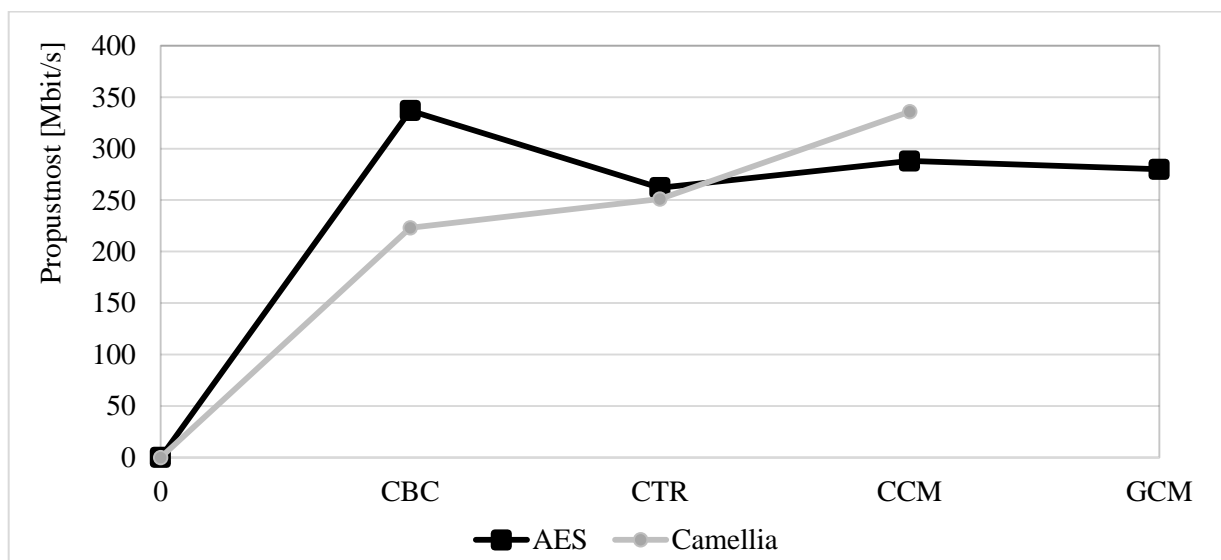
5.1 Praktická část: Implementace šifrovacích IKE metod a ESP zapouzdření

Díky různým pluginům a krypto knihovnám, Strongswan má široký výběr šifrovacích algoritmů IKE protokolů (IKEv1 a IKEv2) a zapouzdření ESP (Encapsulating Security Payload). The Internet Key Exchange (IKE) je standardní protokol IPsec (Internet Protocol Security) používaný k zajištění zabezpečení komunikace VPN sítí a vzdáleného hostitele nebo přístupu k síti. ESP poskytuje autentizaci, integritu a důvěrnost, které chrání proti manipulaci s údaji a hlavně zajišťují ochranu obsahu zpráv.

Zkusíme změnit metody šifrování a zapouzdření ve VPN a otestovat rychlost navázání spojení a rychlost komunikace. Všechny potřebné pluginy už jsou automaticky nainstalované v aktuální verzi Strongswanu (5.4.0). Cílem je zjistit, jaký je vliv různých metod šifrování a zapouzdření na rychlost VPN. Parametry IKE a ESP se mění v konfiguračním souboru */etc/strongswan/ipsec.conf* na Serveru a Klientu. V tomto případě všechny mody CBC, CTR, CCM byly vybrané s délkou 256 bit. Délka přenášeného paketů v iperf je nastavena na 100 bit, výsledky jsou zobrazeny ze Serveru.

Tab 5.1: Implementace metod šifrování IKE (AES a Camellia)

| Mode (256 bit) | Schéma šifrování (IKE) | U-Upload, D-Download | Přenos paketů [Mbytes] | Propustnost [Mbits/sec] |
|---------------------------------|--------------------------------------|-------------------------|------------------------------|----------------------------|
| AES | | | | |
| AES-CBC | ike=aes256-sha512-modp4096! | D | 402 | 337 |
| | | U | 382 | 319 |
| AES-COUNTER | ike=aes256ctr-sha512-modp4096! | D | 312 | 262 |
| | | U | 301 | 250 |
| AES-CCM with 64 bit ICV | ike=aes256ccm8-sha512-modp4096! | D | 346 | 288 |
| | | U | 321 | 266 |
| AES-GCM with 64 bit ICV | ike=aes256gcm8-sha512-modp4096! | D | 335 | 280 |
| | | U | 323 | 270 |
| Camellia | | | | |
| Camellia-CBC | ike=camellia256-sha512-modp4096! | D | 268 | 224 |
| | | U | 269 | 223 |
| 256 bit Camellia- COUNTER | ike=camellia256ctr-sha512-modp4096! | D | 336 | 281 |
| | | U | 300 | 251 |
| Camellia-CCM with 64 bit ICV | ike=camellia256ccm8-sha512-modp4096! | D | 376 | 315 |
| | | U | 404 | 336 |
| Camellia-GCM with 64 bit ICV | Není podporované | | | |



Obr. 5.1: Rychlost (download) při metodách šifrování IKE (AES a Camellia)

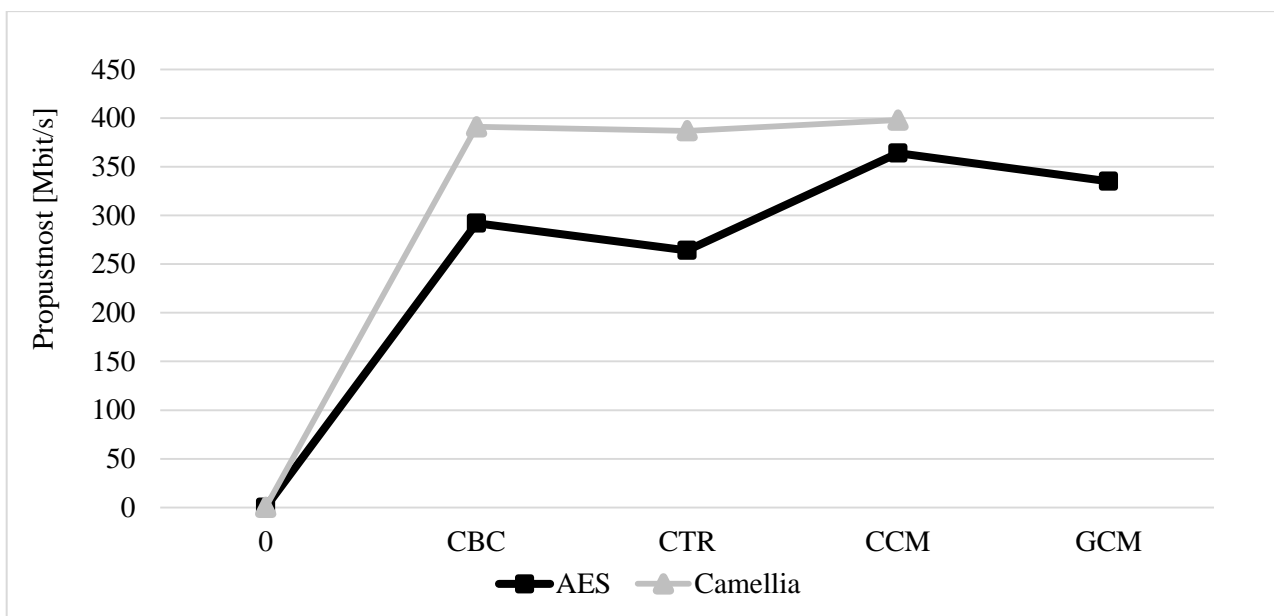
Z grafu Obr. 5.1 vidíme, že nejvyšší dosáhnuta rychlost v tunelů je při šifrovací metodě IKE AES v modu CBC. Metoda Camellia je stejném modu CBC má nejnižší rychlost. Obě šifry, AES a Camellia, mají skoro stejné výsledky v modu CTR, avšak v modu CCM Camellia má větší propustnost o několik bitů.

Z měření můžeme posoudit, že nejvýhodnějším výběrem pro IKE šifrování by byla metoda AES v modu CBC, nebo Camellia v modu CCM. Kvůli tomu, že šifra Camellia není dostupná v modu GCM pro Strongswan, IKE by dobrým řešením i v pro GCM.

V další tabulce Tab 5.2 jsou výsledky měření propustnosti v tunelu s ESP zapouzdřením ve stejných modech jako i pro IKE (CBC, CTR a CCM):

Tab 5.2: Implementace zapouzdření ESP (AES a Camellia)

| Mode (256 bit) | ESP Zapouzdření | U-Upload, D-Download | Přenos paketů [Mbytes] | Propustnost [Mbits/sec] |
|---------------------------------|-----------------------------|-------------------------|------------------------------|----------------------------|
| AES | | | | |
| AES-CBC | esp=aes256-sha512! | D | 384 | 292 |
| | | U | 341 | 284 |
| AES-COUNTER | esp=aes256ctr-sha512! | D | 316 | 264 |
| | | U | 263 | 220 |
| AES-CCM with 64 bit ICV | esp=aes256ccm8-sha512! | D | 435 | 364 |
| | | U | 445 | 373 |
| AES-GCM with 64 bit ICV | esp=aes256gcm8-sha512! | D | 400 | 335 |
| | | U | 469 | 393 |
| Camellia | | | | |
| Camellia-CBC | esp=camellia256-sha512! | D | 467 | 391 |
| | | U | 462 | 385 |
| Camellia- COUNTER | esp=camellia256ctr-sha512! | D | 461 | 387 |
| | | U | 420 | 251 |
| Camellia-CCM with 64 bit ICV | esp=camellia256ccm8-sha512! | D | 476 | 398 |
| | | U | 439 | 367 |
| Camellia-GCM with 64 bit ICV | Není podporované | | | |



Obr. 5.2: Rychlost (download) se zapouzdřením ESP (AES a Camellia)

Z výsledku měření rychlosti se zapouzdřením ESP jsme zjistili, že metoda Camellia je nejrychlejší v modech CBC, CTR a CCM. Takže můžeme říct, že Camellia je nejlepším řešením při implementování ESP ve všech třech modech v IPsec tunelu. Výjimkou je mód GCM, ve kterém Camellia není dostupná pro Strongswan.

6 LABORATORNÍ ÚLOHA.

Cíl

Cílem laboratorní úlohy je umožnit studentům blíže se seznámit s funkcí a konfigurací Strongswan, jeho metodami zabezpečení a šifrování, a také změřit jeho efektivitu pomocí nástroje iperf.

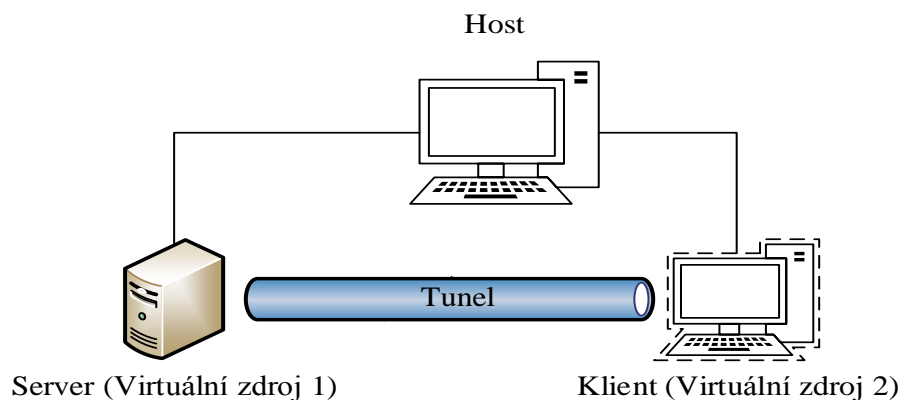
Úvod

Strongswan je kompletní řešení založené na IPsec, který poskytuje šifrování a ověřování pro servery a klienty. Je nativně podporováno jádrem Linuxu, ale konfigurace šifrovacích klíčů je ponechána na uživateli. Protokoly IKE (Internet Key Exchange) se používají v protokolech IPsec VPN pro vyjednávání bezpečné výměny klíčů pomocí různých prostředků, včetně certifikátů, předsdílených klíčů nebo obou. ESP (Encapsulating Security Payload) poskytuje autentizaci, integritu a důvěrnost, které chrání proti manipulaci s údaji a hlavně zajišťují ochranu obsahu zpráv Strongswan má neomezený počet typů klíčů pro autentizaci. Díky různým pluginům a krypto knihovnám Strongswan poskytuje široký výběr kryptografických algoritmů IKE protokolů (podporuje verze IKEv1 a IKEv2). Nativně podporuje většinu moderních klientů, včetně Linuxu, Windows 7, Apple iOS, Mac OSX, FreeBSD a BlackBerry OS.

Vybavení pracoviště

VirtualBox, CentOS7, Strongswan, iperf, Wireshark.

Schéma zapojení



Obr. 6.1: Schéma zapojení

Postup řešení

1. Otevřete program VirtualBox a nahrajte do něho disk s operačním systémem CentOS7, DVD ISO verzi. Během instalace zadejte a uložte všechny požadované parametry.

2. Až systém bude kompletně nainstalován, proveďte jeho klonování a pojmenujte jeden stroj jako Server a druhý jako Klient. V záložce Síť v nastavení změňte Adapter 1 na NAT, ostatní parametry neměňte. Přihlaste se do systému a připojte se do Internetu. Zkontrolujte, aby IP adresy na strojích nebyly stejné, jinak je třeba to změnit.

3. V terminálu se přihlaste jako správce (pomocí příkazu `su<user>` nebo `su`). Nainstalujte EPEL repositář a Strongswan na obou strojích:

```
# yum install epel-release
```

```
# yum install strongswan -y
```

4. Opravte konfigurační soubor `ipsec.conf` na Serveru podle příkladu dole. Kde je *left* ip adresa Serveru a *right* ip adresa Klienta:

```
# vi /etc/strongswan/ipsec.conf
config setup
    charondebug="all"
    uniqueids=yes
    strictcrpolicyn=no
conn %default
conn tunnel
    left=x.x.x.x/x
    leftsubnet=10.1.0.0/16
    right=y.y.y.y/y
    rightsubnet=11.1.0.0/16
    ike=aes256-sha2_256-modp1024!
    esp=aes256-sha2_256!
    keyingtries=0
    ikelifetime=1h
    lifetime=8h
    dpddelay=30
    dpdtimeout=120
    dpdaction=clear
    authby=secret
    keyexchange=ikev2
    auto=start
    type=tunnel
```

5. Opravte konfigurační soubor na Klientu podobným způsobem, rozdíl bude v tom, že místo *left* zadáte ip adresu Klienta, a místo *right* ip adresu Serveru. Takhle změňte i *leftsubnet* a *rightsubnet*.

6. Poté otevřete soubor *ipsec.secrets*, kam je třeba vložit sdílený klíč, který se bude používat mezi Serverem a Klientem při navázání spojení. Tento řádek musí být stejný na obou strojích ve stejném formátu, jak je ukázáno dole v příkladu. Kde *x.x.x.x* - ip adresa Serveru, *y.y.y.y* - ip adresa Klienta, *sharedsecret* – sdílený klíč.

```
# vi /etc/strongswan/ipsec.secrets
x.x.x.x y.y.y.y : PSK 'sharedsecret'
```

7. Když oba soubory jsou nakonfigurované, spusťte Strongswan na Serveru a Klientu a ověřte spojení:

```
# systemctl start strongswan
# ip xfrm state
```

8. Pro měření propustnosti VPN použijte jednoduchý nástroj *iperf*. Nainstalujte *iperf* na obou strojích a spusťte současně. Kde *x.x.x.x* – ip adresa Serveru.

Výsledky si uložte.

```
# yum install iperf
# iperf -s %na Serveru
# iperf -c x.x.x.x -d %na Klientu
```

9. Naistalujte Wireshark a spusťte aplikaci. Zachyťte pakety inicializace tunelu a autorizace mezi stroji, spočítejte dobu navázání spojení. Výsledky dosáhnete tak, že spočítáte časový rozdíl mezi prvním a posledním pakety *IKE_AUTH*.

```
# yum install wireshark-gnome
# wireshark
```

10. V konfiguračním souboru změňte šifrovací metodu IKE a ESP a opakujte testování pro IKE a ESP zvlášť. Katalog šifrovacích algoritmů najdete na oficiální stránce Strongswan. Výsledky si uložte a porovnejte.

Kontrolní otázky

1. Co je Strongswan? Jaký je rozdíl mezi Strongswan a IPsec?
2. Jaké pakety se vysílají mezi serverem a klientem před navázáním spojení a po ustanovení tunelu IPsec?

3. Jaký je rozdíl mezi IKE a ESP?
4. Jaké existují šifrovací metody IKE a ESP?

Závěr

Účelem této práce bylo popsat základní principy VPN a porovnat jednotlivé řešení dle některých parametrů. Řešení byly vybrány vzhledem k možnosti jejich realizace na operačním systému CentOS7 a dostupností softwaru. Existuje množství řešení VPN, ale tato práce má jen prezentační charakter, proto byly vybrány nejpoužívanější typy VPN, které lze implementovat v Linuxovém prostředí. Některé z testovaných řešení jsou dostupné i pro jiné operační systémy.

V první praktické části byly implementované některé typy VPN za účelem zjistit, jak VPN ovlivňuje práci operačního systému, a také změřit rychlost každého typu VPN. V dalším kroku byly popsány krypto knihovny a typy autentizace IPsec tunelu. Ve druhé praktické části byly vybrány různé metody šifrování pro IKE a ESP. Podle výsledků jsme zjistili jaká je nejrychlejší metoda pro určité mody.

Literatura

- [1] Compare VPN Protocols [online]. 2016 [cit. 14.12.2016]. Dostupné z URL: <http://www.giganews.com/vyprvpn/compare-vpn-protocols.html>
- [2] VPN everywhere: IPsec without L2TP with strongSwan [online] poslední aktualizace 03.02.2015 [cit. 14.12.2016]. Dostupné z URL: <https://www.lowendtalk.com/discussion/44964/vpn-everywhere-ipsec-without-l2tp-with-strongswan-even-inopenvz>
- [3] TAKELE DEGEFA, R. Bachelor's Thesis:VPN Scenarios, Configuration and Analysis[online]. 10.2015 [cit. 14.12.2016]. Dostupné z URL: <https://theses.fi/handle/10024/98730>
- [4] SUGRANES, A.B, BAQUERO G.M., Final Master Thesis: Telemedicine System in the South Atlantic. 07.2012 [cit. 14.12.2016]. Dostupné z URL: https://www.google.cz/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&cad=rja&uact=8&ved=0ahUKEwia hU2ht_TQAhWEBsAKHTecBHgQFggaMAA&url=https%3A%2F%2Fupcommons.upc.edu%2Fbitstream%2Fhandle%2F2099.1%2F15879%2Fmemoria.pdf&usq=AFQjCNGfx9DtR92vOM9QR2EBLu-2J3fkOg&bvm=bv.141320020,d.bGg
- [5] IPSEC VPN on Centos 7 with StrongSwan [online]. 2016, poslední aktualizace 30.12.2014 [cit. 14.12.2016]. Dostupné z URL: https://raymii.org/s/tutorials/IPSEC_vpn_with_CentOS_7.html
- [6] Configuration Files [online]. 2016, poslední aktualizace 20.10.2016 [cit. 14.12.2016]. Dostupné z URL: <https://wiki.strongswan.org/projects/strongswan/wiki/IpsecConf>
- [7] How to install OpenVPN Server and Client on CentOS 7 [online]. 2016 [cit. 14.12.2016]. Dostupné z URL: <https://www.howtoforge.com/tutorial/how-to-install-openvpn-on-centos-7/>
- [8] Openvpn [online]. 2016 [cit. 14.12.2016]. Dostupné z URL: <http://fedoraproject.org/wiki/Openvpn>
- [9] How To Install PPTP VPN On Centos 7]. 2016 [cit. 14.12.2016]. Dostupné z URL: <https://www.digitalocean.com/community/questions/how-to-install-pptp-vpn-on-centos-7>
- [10] Linux Free S/WAN [online]. 2017 [cit. 7.6.2017]. Dostupné z URL: <http://www.freeswan.org/>
- [11] Openswan [online]. 2016 [cit. 7.6.2017]. Dostupné z URL: <https://www.openswan.org/>
- [12] Libreswan VPN software [online]. 2016, poslední aktualizace 24 May 2016 [cit. 7.6.2017]. Dostupné z URL: <https://libreswan.org/>
- [13] Iperf Network Performance Measuring [online]. 2016 [cit. 14.12.2016]. Dostupné z URL: <http://centoshowtos.org/network-and-security/iperf/>
- [14] Libgcrypt knihovna [online]. 2017, poslední aktualizace 12.6.2017 [cit. 7.6.2017]. Dostupné z URL: <https://en.wikipedia.org/wiki/Libgcrypt>

- [15] OpenSSL Cryptography and SSL/TLS Toolkit [online]. 2016 [cit. 7.6.2017]. Dostupné z URL: <https://www.openssl.org/docs/man1.0.2/crypto/crypto.html>
- [16] Internet Key Exchange (IKE) [online]. Poslední aktualizace 2009 [cit. 7.6.2017]. Dostupné z URL: <http://searchsecurity.techtarget.com/definition/Internet-Key-Exchange>
- [17] IPsec configuration and connections [online]. 2011 [cit. 7.6.2017]. Dostupné z URL: <https://linux.die.net/man/5/ipsec.conf>
- [18] Configuring OpenSwan IPsec Server [online]. 2013 [cit. 7.6.2017]. Dostupné z URL: <http://blog.jameskyle.org/2012/07/configuring-openswan-ipsec-server/>
- [19] IKEv2 Cipher Suites [online]. 2017 [cit. 7.6.2017]. Dostupné z URL: <https://wiki.strongswan.org/projects/strongswan/wiki/IKEv2CipherSuites>

Seznam zkratek

| | |
|-----------|---|
| VPN | Virtual Private Networks |
| PPTP | Point-to-Point Protocol |
| MPPE | Microsoft Point-to-Point Encryption |
| PAP | Password Authentication Protocol |
| CHAP | Challenge Handshake Authentication Protocol |
| EAP-TLS | Extensible Authentication Protocol - Transport Layer Security |
| IKE | Internet Key Exchange |
| ESP | Encapsulating Security Payload |
| L2TP | Layer 2 Tunneling Protocol |
| FreeS/WAN | Free SecureWide-Area Networking |
| NAT | Network Address Translation |
| DES | Data Encryption Standard |
| PFS | Perfect Forward Secrecy |
| DH | Diffie Hellman |
| IETF | EngineeringTaskForce |