VŠB TECHNICKÁ UNIVERZITA OSTRAVA
FAKULTA ELEKTROTECHNIKY A INFORMATIKY

Kvalita služby v sítích IPsec VPN

Quality of Service in IPsec VPN Networks

Bc. Lukáš Bik

Diplomová práce

Ing. Petr Machník, Ph.D.

Ostrava, 2022

VŠB – Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra telekomunikační techniky

Zadání diplomové práce

Student: **Bc. Lukáš Bik**

Studijní program: N2647 Informační a komunikační technologie

Studijní obor: 2601T013 Telekomunikační technika

Téma: Kvalita služby v sítích IPsec VPN
Quality of Service in IPsec VPN Networks

Jazyk vypracování: čeština

Zásady pro vypracování:

Cílem diplomové práce je návrh, realizace a testování nástrojů pro podporu kvality služby (QoS) ve virtuálních privátních sítích IPsec v laboratorním prostředí s využitím směrovačů Huawei a Cisco.

Vypracování práce bude splňovat následující body zadání:
1. Popište nástroje pro implementaci kvality služby (QoS) v prostředí virtuálních privátních sítí.
2. Navrhněte a v laboratorních podmínkách realizujte počítačové sítě využívající směrovače Huawei a Cisco, v nichž jsou použity alespoň 3 různé nástroje pro podporu QoS. Ověřte funkčnost navržených řešení.
3. Ověřte kompatibilitu směrovačů Huawei a Cisco při implementaci těchto nástrojů.

Seznam doporučené odborné literatury:

[1] ODOM, Wendell a Michael J. CAVANAUGH. *Cisco QOS exam certification guide.* 2nd ed. Indianapolis, IN: Cisco Press, 2005. ISBN 978-1-58720-124-0.
[2] SZIGETI, Tim, Christina HATTINGH, Robert BARTON a Kenneth BRILEY. *End-to-end QoS network design.* 2nd edition. Indianapolis, IN: Cisco Press, 2014. Cisco Press networking technology series. ISBN 978-158-7143-694.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **Ing. Petr Machník, Ph.D.**

Datum zadání: 01.09.2021

Datum odevzdání: 30.04.2022

prof. Ing. Miroslav Vozňák, Ph.D.
vedoucí katedry

prof. Ing. Jan Platoš, Ph.D.
děkan fakulty

Abstrakt

Tato diplomová práce popisuje problematiku zavádění nástrojů kvality služeb v IPsec VPN sítích. V teoretické rovině nejprve popíše základní charakteristiku problematiky kvality služeb. V tomto segmentu diplomové práce budou popsány i konkrétní mechanismy, které bývají v rámci nástrojů kvality služby implementovány. Dále teoretická část této diplomové práce popisuje vlastnosti IPsec VPN technologie. Jsou zde vysvětleny principy na jejichž základě tato technologie funguje. V praktické části jsou rozebrány konkrétní problémy, které při součinnosti nástrojů kvality služeb a IPsec VPN technologie vznikají. Na IPsec VPN topologii s využitím zařízení výrobců Cisco a Huawei jsou postupně aplikovány nástroje kvality služby na klasifikaci, značkování, tvarování, omezování a prioritizaci provozu. Implementační část rovněž popisuje chybné značkování provozu na zařízeních Cisco s IOS verzí 12.4 a navrhuje alternativní postup zavedení IPsec VPN, který chybné značkování provozu řeší.

Klíčová slova

Cisco; Huawei; IPsec VPN; Klasifikace provozu; Kvalita služby; Omezování provozu; Prioritizace provozu; Tvarování provozu; Značkování provozu

Abstract

This thesis describes the implementation of quality-of-service tools in IPsec VPN networks. On the theoretical level, it first describes the basic characteristics of the quality-of-service. This segment of the thesis will also describe the specific mechanisms which tend to be implemented within the quality-of-service tools. Next, the theoretical part of this thesis describes the characteristics of IPsec VPN technology. The principles on the basis of which this technology works are explained.

The practical part discusses the specific problems which arise when quality-of-service tools and IPsec VPN technology interact. Quality-of-service tools for classification, marking, shaping, policing and prioritization are sequentially applied to an IPsec VPN topology using the equipment from Cisco and Huawei. The implementation section also describes the mismarking of the packets on Cisco devices running on IOS version 12.4 and proposes an alternative IPsec VPN implementation procedure which can be able to solve the mismarking of the packets.

Key words

Cisco; Huawei; IPsec VPN; Quality of service; Traffic classification; Traffic marking; Traffic policing; Traffic prioritisation; Traffic shaping;

Poděkování

Rád bych poděkoval vedoucímu této diplomové práce Ing. Petru Machníkovi, Ph.D. za odbornou pomoc a vstřícný přístup při řešení problémů týkajících se této práce.

Obsah

Seznam použitých zkratek

Zkratka	Význam
ACL	Access Control List
AES	Advanced Encryption Standard
AF	Assured Forwarding
AH	Authentication Header
Bc	Burst size
CAC	Call Admission Control
CBS	Commited Burst Size
CBWFQ	Class-Based Weighted Fair Queuing
CIR	Commited Information Rate
CQ	Custom Queuing
cRTP	Compressed Real-Time Protocol
CS	Class Selector
DF	Don't Fragment
DH	Diffie-Hellman Algorithm
DiffServ	Differentiated Services
DRE	Date Redundancy Elimination
DSCP	Differentiated Services Code Point
EBS	Excess Burst Size
EF	Expedited Forwarding
ECN	Explicit Congestion Notification
ESP	Encapsulating Security Payload
FIFO	First In First Out
GRE	Generic Routing Encapsulation
GSR	Gigabit Switch Router
HMAC	Hash-based Message Authentication Code
IKE	Internet Key Exchange
IntServ	Integrated Services
IOS	Internetwork Operating System
IP	Internet Protocol
IPsec	Internet Protocol Security
ISAKMP	Internet Security Association and Key Management Protocol
ISP	Internet Service Provider
LFI	Link Fragmentation and Interleaving
LLQ	Low Latency Queuing
MDRR	Modified Deficit Round-Robin
MOS	Mean Opinion Score
MPLS	Multiprotocol Label Switching
MSS	Maximum Segment Size
MTU	Maximum Transfer Unit

NBAR	Network-Based Application Recognition
PIR	Peak Information Rate
PQ	Priority Queuing
QoE	Quality of Experience
QoS	Quality of Service
RED	Random Early Detection
RSVP	Resource Reservation Protocol
RTP	Real-Time Protocol
RTT	Round-trip Time
SA	Security Association
SN	Sequence number
SPI	Security Parameter Index
SSL	Secure Sockets Layer
SSH	Secure Shell Protocol
TCP	Transmission Control Protocol
TLS	Transport Layer Security
ToS	Type of Service
UDP	User Datagram Protocol
VoIP	Voice over Internet Protocol
VPN	Virtual Private Network
VTI	Virtual Tunnel Interface
WAAS	Wide Area Application Services
WRED	Weighted Random Early Detection
WFQ	Weighted Fair Queuing

Seznam obrázků

Seznam tabulek

Úvod

V době, kdy se internet stával součástí každé domácnosti rostly i nároky na jeho služby. Jednoduché funkce, kterými internet při vzniku disponoval, se postupem času vyvíjely a stále se vyvíjí. Prvotní Best Effort model, který uživatelská data nikterak nerozlišoval, byl brzy nahrazen nástroji kvality služeb. Tyto nástroje jsou schopny jednotlivé typy provozu rozlišovat a na tyto různé typy aplikovat příslušná pravidla. Využití nástrojů kvality služeb umožňuje administrátorům zlepšit přenosové parametry pro určitý typ provozu na úkor přenosových parametrů jiného typu provozu. Tyto nástroje jsou na základě klasifikace schopny provoz různými způsoby i značit. Ve výsledku může nastat situace, ve které je směrovač, který v daný moment provádí klasifikaci paketu, odkázán pouze na tuto značku.

Kromě požadavků na zkvalitnění služeb pro určité typy provozu byly časem vzneseny i požadavky na zabezpečení provozu. K tomuto byla navržena technologie IPsec VPN. Jedná se o jednu z nejvyužívanějších VPN technologií. Zajišťuje veškeré parametry, které by kvalitní VPN síť měla splňovat. Mezi tyto parametry se řadí ochrana dat vůči odposlechům. Tato ochrana je zajištěna pomocí šifrování dat. Dále pomocí hashovacích funkcí zajišťuje integritu dat, tedy garantuje to, že přijatá data při průchodu sítí nebyla nijak změněna. Rovněž je pomocí IPsec VPN zajištěna autenticita obou komunikujících stran.

Jak již tomu tak bývá, řada výhod přináší i jisté komplikace. Jeden z problémů nastává v momentě, kdy je na zabezpečený provoz potřeba aplikovat mechanismy kvality služeb. I samotné směrovače, které navazují IPsec VPN spojení mají ztíženou klasifikaci, a to z toho důvodu, že VPN procesy často probíhají dříve než procesy kvality služeb. Nicméně, stále existují možná řešení této situace, jež tato diplomová práce popíše. Značně komplikovanější tuto klasifikaci provozu mají směrovače uvnitř tunelované sítě. Tyto směrovače jsou při klasifikaci provozu bez možného řešení odkázané pouze na výše uvedené značky.

Tento a další problémy vzniklé při implementaci dvou jmenovaných nástrojů popisuje tato diplomová práce. V teoretické části popisuje nejprve charakteristické vlastnosti nástrojů kvality služeb. Následně pak tato diplomová práce popisuje vlastnosti IPsec technologie. Zároveň představuje nejčastější problémy, které je nutné řešit při implementaci těchto nástrojů. V praktické části je popsána implementace nejprve samotné IPsec VPN technologie. Na této topologii jsou poté postupně popisovány postupy implementace jednotlivých nástrojů kvality služeb. Praktická část práce se rovněž zabývá problémy, které při součinnosti těchto dvou nástrojů mohou vzniknout. V závěru jsou navrženy, sestaveny a popsány tři scénáře s využitím zařízení dvou různých výrobců Cisco a Huawei.

1 Kvalita služby

Základním účelem služby QoS (Quality of Services) je správa spojení mezi síťovými prvky, která zajišťuje co nejefektivnější tok paketů datovou sítí. Protože ne všechny pakety jsou si rovny, můžeme s pakety různých služeb nakládat podle jejich důležitosti. Služba QoS tedy implementuje systém ,,řízené nespravedlnosti" v síti. Některé datové toky jsou ve výsledku zvýhodňovány na úkor ostatních datových toků. Například relace citlivé na zpoždění (video hovory, streaming) obcházejí fronty paketů, které jsou na zpoždění naopak méně náchylné. V momentě, kdy se některá z front pakety přeplní, může zase dojít k selekci těch, které je nutné v provozu zachovat, a naopak zahodit ty, kterých ztráta nemá citelný dopad na koncového účastníka sítě [2].

Důvody zavedení QoS jsou proto zřejmé. Systémy bez užití QoS se mohou projevovat různě, ve výsledku ale většinou negativně. V případě hlasových služeb může dojít k vypadávání hlasu nebo k úplnému výpadku hovoru. U služeb přenášejících video může zase dojít ke špatnému zobrazování obrazu, zpomalování videa nebo k přenosu videa, kde zvuk není synchronizován s videem. Všechny tyto negativní vlivy tvoří v součtu téměř nepoužitelnou datovou síť.

QoS se tedy zasluhuje nezměrnou měrou o zkvalitnění služeb datového přenosu. K tomuto zkvalitnění přenosu QoS využívá mechanismy, které přímo ovlivňují následující prvky sítě [1]:

- Propustnost (Bandwidth)
- Zpoždění (Delay)
- Variabilita hodnoty zpoždění (Jitter)
- Ztrátovost paketů (Packet loss)

Bohužel i zde vše souvisí se vším, takže zlepšením jednoho parametru přenosu degradujeme parametr jiný. Proto je nejdůležitější možnosti QoS využívat uvážlivě. Například konkrétní relaci vyžadující vylepšení jistého parametru toto vylepšení umožnit, zatímco jiné relaci, která ten samý parametr tolik nepotřebuje parametr degradovat [1].

1.1 Propustnost

Termín propustnost udává množství dat, které je přeneseno mezi dvěma zařízeními za určitou dobu. V některých případech se může hodnota šířky pásma rovnat přenosové rychlosti linky. V ostatních případech je propustnost hodnotou vždy nižší než přenosová rychlost dané linky. Propustnost je totiž vždy určena tou nejpomalejší linkou na dané trase. Při řešení potíží se šířkou pásma je za nejlepší řešení většinou považováno šířku pásma přidat [1]. Nicméně toto řešení bývá drahé a nemusí být trvalé.

Některé nástroje kvality služby redukují potřebný počet bitů k přenosu využitím datové komprese. Tato komprese zamezuje vzniku front v momentě, kdy je požadována hodnota propustnosti vyšší než hodnota, kterou linka nabízí. Další nástroje, které propustnost ovlivňují napřímo se nazývají call admission control. CAC nástroje rozhodují, zda je síť schopná vést další instanci, například video hovoru. Řekněme, že existuje linka, která kapacitou odpovídá třem hovorům. Bez použití CAC metod by nový, čtvrtý hovor zapříčinil degradaci kvality všech ostatních hovorů. Tomu CAC zabrání pozdržením nového, nadbytečného hovoru, nebo jeho přesměrováním prostřednictvím VoIP. Vedle CAC metod máme metody obsluhy paketových front. Tyto mechanismy umožňují prioritizaci provozu pro různé druhy paketů [1].

1.2 Přenosové zpoždění

Všechny pakety v datové sítí podléhají zpoždění. Existuje několik typů zpoždění, které v součtu tvoří přenosové zpoždění mezi dvěma zařízeními v síti [1].

1.2.1 Serializační zpoždění

Serializační zpoždění definuje čas nutný pro zakódování bitů paketu na fyzické rozhraní. Je přímo ovlivněno rychlostí linky a velikostí paketu. Serializační zpoždění lze popsat následující rovnicí [1]:

$$serializační\ zpoždění = \frac{počet\ odeslaných\ bitů}{rychlost\ linky}$$

1.2.2 Propagační zpoždění

Propagační zpoždění definuje čas nutný pro odeslání bitu z jednoho konce linky na konec druhý. Toto zpoždění je dáno fyzickými vlastnostmi přenosových médií a jediná proměnná která propagační zpoždění přímo ovlivňuje, je délka tohoto média. Propagační zpoždění lze v případě elektrického nebo optického signálu popsat následovně [1]:

$$propagační\ zpoždění = \frac{délka\ přenosového\ média\ [m]}{2{,}1 \cdot 10^8\ [ms^{-1}]}$$

1.2.3 Doba čekání v paketové frontě

Toto zpoždění vzniká v momentě, kdy odeslané pakety musí čekat na odeslání i zbylých paketů. Skládá se z času, který pakety stráví ve frontách v zařízení. Vzniká většinou pouze na směrovačích v odchozích směrech. Ve směrech příchozích je toto zpoždění zanedbatelné [1].

1.2.4 Zpoždění přesměrování

Zpoždění přesměrování je čas, který zařízením trvá zpracování příchozího paketu a jeho zařazení do odchozí fronty. Zpoždění tedy nezahrnuje čas, který paket stráví ve frontě [1].

1.2.5 Tvarovací zpoždění

Zpoždění vzniká variabilní rychlostí obsluhy jednotlivých front. V některých případech je totiž žádoucí rychlejší obsluha front nehledě na to, že je vyšší pravděpodobnost ztráty paketů. V opačných případech je zase potřeba pomalejší, ale spolehlivější obsluha front [1].

1.2.6 Zpoždění sítě

Zpoždění vzniklé v místě, kam jako administrátor nemáme přístup, např. síť jiného ISP. Zpoždění sítě se bude lišit v závislosti na různých proměnných jako je stav linek, nebo celkové přetížení dané sítě. Největší variabilitu zpoždění sítě způsobuje zpoždění vzniklé metodami obsluhy paketových front uvnitř nám nepřístupné sítě [1].

1.3 Nástroje ovlivňující zpoždění

Mezi nástroje ovlivňující přenosové zpoždění patří paketové fronty, fragmentace velkých paketů a jejich prokládání malými a datová komprese.

1.3.1 Metody obsluhy paketových front

Jedná se o nejpopulárnější QoS nástroj, který vytváří a efektivně obsluhuje paketové fronty. Do paketových front se na základě určených pravidel vkládají příchozí pakety. Ve výsledku jsou pak některé pakety na úkor jiných paketů odeslány rychleji. Tento mechanismus samozřejmě nesnižuje zpoždění

celkově, ale jenom pro vybrané pakety, které jsou citlivé na přenosové zpoždění. U těch méně prioritních paketů naopak toto zpoždění úměrně poroste [1].

1.3.1.1 **Metoda First In, First Out**

Jedná se o nejjednodušší metodu, která vytváří jedinou frontu fungující na principu FIFO. Paket, který do fronty dorazí jako první, je jako první z fronty také odeslán. Tato metoda slouží primárně k tomu, aby nedocházelo k zahazování paketů v momentě dočasného přehlcení směrovače. Přenosu citlivému na přenosové zpoždění není garantována žádná hodnota propustnosti. Rovněž i zpoždění může v některých momentech narůstat z důvodu času stráveného ve frontě. Při této metodě není nutné provádět žádnou klasifikaci příchozích paketů [1].

1.3.1.2 **Metoda Priority Queuing**

Tato metoda rozděluje pakety do čtyř front, přičemž každá fronta má odlišnou prioritu. Pakety z fronty s vyšší prioritou mají vždy přednost před pakety z fronty s nižší prioritou. Při každém odbaveném paketu dochází ke kontrole, zda do fronty s vyšší prioritou nedorazil nový paket. Metoda sice garantuje nejlepší možné přenosové parametry provozu s vysokou prioritou, nicméně vše na úkor paketů s nižší prioritou [1].

1.3.1.3 **Metoda Custom Queuing**

CQ umožňuje využití 16 front. Těmto frontám je správcem nakonfigurován počet bytů, který může být v jednom cyklu z fronty odeslán. V momentě, kdy dojde k překročení tohoto limitu, nebo ve frontě nezbývá žádný paket začne se obsluhovat fronta následující. Na rozdíl od metody Priority Queuing tato metoda jednotlivým frontám garantuje určité procento přenosové rychlosti linky. Na druhou stranu nedokáže patřičně upřednostnit provoz, který je citlivý na přenosové zpoždění [1][10].

1.3.1.4 **Metoda Modified Deficit Round-Robin**

MDRR byl speciálně navržen pro skupinu směrovačů typu Gigabit Switch Router (GSR). Metoda MDRR řeší nedostatky CQ metody v přesnosti rozdělení přenosové rychlosti. U předchozí metody snadno docházelo k překročení limitu počtu bytů pro jednotlivé fronty. MDRR tento problém řeší způsobem, že hodnotu, o kterou daná fronta překročí svůj nakonfigurovaný limit, v příštím cyklu této frontě odečte [1].

1.3.1.5 **Metoda Weighted Fair Queuing**

Metoda WFQ se oproti zmíněným metodám výrazně liší. Neumožňuje uživatelem definovanou klasifikaci provozu. Provoz je naopak rozdělen do jednotlivých datových toků jdoucích z určité IP adresy a portu aplikace na určitou IP adresu a port aplikace. Na jednom rozhraní může být maximálně 4096 front (resp. datových toků).

WFQ každé frontě přiděluje určité vážené procento z celkové přenosové rychlosti v závislosti na počtu datových toků. Každému paketu je přiděleno sekvenční číslo (SN). Na základě hodnoty SN se určuje, v jakém pořadí jsou fronty obsluhovány. Pakety s nižší hodnotou SN mají přednost oproti paketům s vyšším SN.

SN je určeno na základě tohoto výpočtu:

$$váha = \frac{32384}{IP\ Precedence + 1}$$

$$SN = SN\ předchozího\ paketu + (váha * velikost\ příchozího\ paketu)$$

Z uvedených rovnic vyplývá, že malé pakety s vyšší hodnotou IP Precedence budou označeny nižší hodnotou sekvenčního čísla než velké pakety s nízkou IP Precedence hodnotou. Nicméně, i přes tyto sofistikované mechanismy WFQ negarantuje přenosovou rychlost paketům citlivým na přenosové zpoždění [1][10].

1.3.1.6 **Metoda Class-Based Weighted Fair Queuing**

CBWFQ může garantovat jednotlivým frontám, kterých může být až 64, určité procento přenosové rychlosti odchozího rozhraní. V praxi se pro jednotlivé typy provozu vyčlení fronty, kterým je následně správcem přidělena potřebná hodnota přenosové rychlosti. Tento princip je stejný jako u metody CQ. CBWFQ navíc používá metodu WFQ pro neklasifikovaný provoz spadající do výchozí třídy ,,class-default". Rovněž tato metoda negarantuje potřebné přenosové vlastnosti provozu citlivému na přenosové zpoždění [1][10].

1.3.1.7 **Metoda Low Latency Queuing**

Tato metoda konečně implementuje mechanismus garantující potřebné přenosové parametry pro provoz citlivý na zpoždění. Problém řeší jednou (nebo i více) prioritní frontou, které je správcem nakonfigurovaná maximální přenosová rychlost. Po překročení nakonfigurované přenosové rychlosti dochází k zahození přebývajících paketů. Zbylé fronty jsou obsluhovány stejným způsobem jako u CBWFQ metody [1][10].

1.3.2 **Prokládání a fragmentace linky**

Čas nutný pro serializaci paketu na lince je závislý na rychlosti dané linky a velikosti paketu. V momentě, kdy směrovač začne odesílat paket, nehledě pak na jeho velikost, nepřestane, dokud celý paket neodešle. Metoda Link Fragmentation and Interleaving (LFI) pak umožňuje odesílání tohoto jednoho paketu rozdělit na několik menších částí, fragmentů. Tyto fragmenty lze následně prokládat prioritními pakety, které na směrovač dorazily později [1].

1.3.3 **Komprese**

Komprese, jak už název napovídá komprimuje data tak, aby zabírala méně bitů než původní paket. Komprese snižuje serializační zpoždění, a to díky snížení počtu bitů nutných k odeslání. Zároveň ale jisté zpoždění vzniká, a to dobou nutnou ke kompresi. Proto je nutné při použití komprese postupovat uvážlivě [1].

1.3.4 **Tvarování provozu**

Traffic Shaping je nástroj QoS, který pakety pozdrží v čekací frontě, pokud hrozí překročení povolené rychlosti na určitém rozhraní. V tom se liší od nástroje Traffic Policing, který tyto pakety rovnou zahazuje.

Jeli třeba přenášet data nižší rychlostí, než je fyzická přenosová rychlost rozhraní, dosáhne se toho střídáním krátkých intervalů, kdy se pakety přenášejí fyzickou přenosovou rychlostí rozhraní a kdy se nepřenáší nic. Traffic Shaping definuje, kolik bytů dat lze v určitém krátkém časovém intervalu přenést,

zbytek ve frontě čeká na další interval. Oproti tomuhle postupu metoda Traffic Policing definuje určitou minimální dobu mezi přenosem dvou paketů. Paket, který tuto dobu nesplní, je zahozen. Traffic Shaping je vhodné využit v případě nárazovitého provozu. V momentě, kdy by provoz byl dlouhodoběji vyšší, než je definovaný limit, rychle by došlo k zaplnění front a následnému zahazování paketů [1][10].

1.4 Variabilita zpoždění (Jitter)

Variabilita zpoždění vzniká náhlou změnou zpoždění v síti. Přenos, ve kterém je důležité pakety odesílat a přijímat v určitých a neměnných intervalech (hlas, video), může být variabilitou zpoždění negativně ovlivněn [1].

1.5 Ztrátovost paketů

Posledním QoS parametrem je ztrátovost paketů. Ke ztrátám paketů na směrovačích dochází z několika důvodů. S QoS pak přímo souvisí paketové ztráty, které jsou způsobeny přeplněnými frontami.

Paketové ztráty se QoS snaží redukovat Random Early Detection (RED) metodami. Ztrátovost paketů může být rovněž ovlivněna paketovými frontami. Zvětšením maximální velikosti fronty můžeme předejít ztrátám paketů, ale na druhou stranu poroste zpoždění důsledkem čekání paketů v paketových frontách [1].

1.6 Klasifikace a značkování paketů

QoS klasifikační nástroje kategorizují pakety po prozkoumání jejich obsahu. Nástroje značkování pak umožňují záhlaví příslušných paketů podle potřeby měnit. Většina QoS nástrojů se na základě klasifikace rozhoduje, jakým způsobem dané spojení zpracovávat. Například rozlišení paketů přenášejících hovorová data oproti paketům obyčejného datového přenosu a následné rozdělení do patřičných front. V rámci klasifikace a značkování se často užívají pojmy ToS (Type of Service) a DSCP (Differentiated Services Code Point*)*. Tyto hodnoty jsou obsaženy v záhlaví IP paketu a určují typ a prioritu daného provozu [1].

1.7 Modely kvality služby

K zajištění kvality služby lze přistupovat třemi různými způsoby. Úplně nejzákladnější model, na jehož principu vznikl internet, se nazývá Best Effort. Tento model data nijak nerozlišuje a s různými typy dat se zachází stejným způsobem.

Druhým modelem QoS je Integrated Services. Tento model zajišťuje služby QoS vyjednáním přenosových parametrů. Poslední model Differentiated Services rozlišuje data do tříd [1][10].

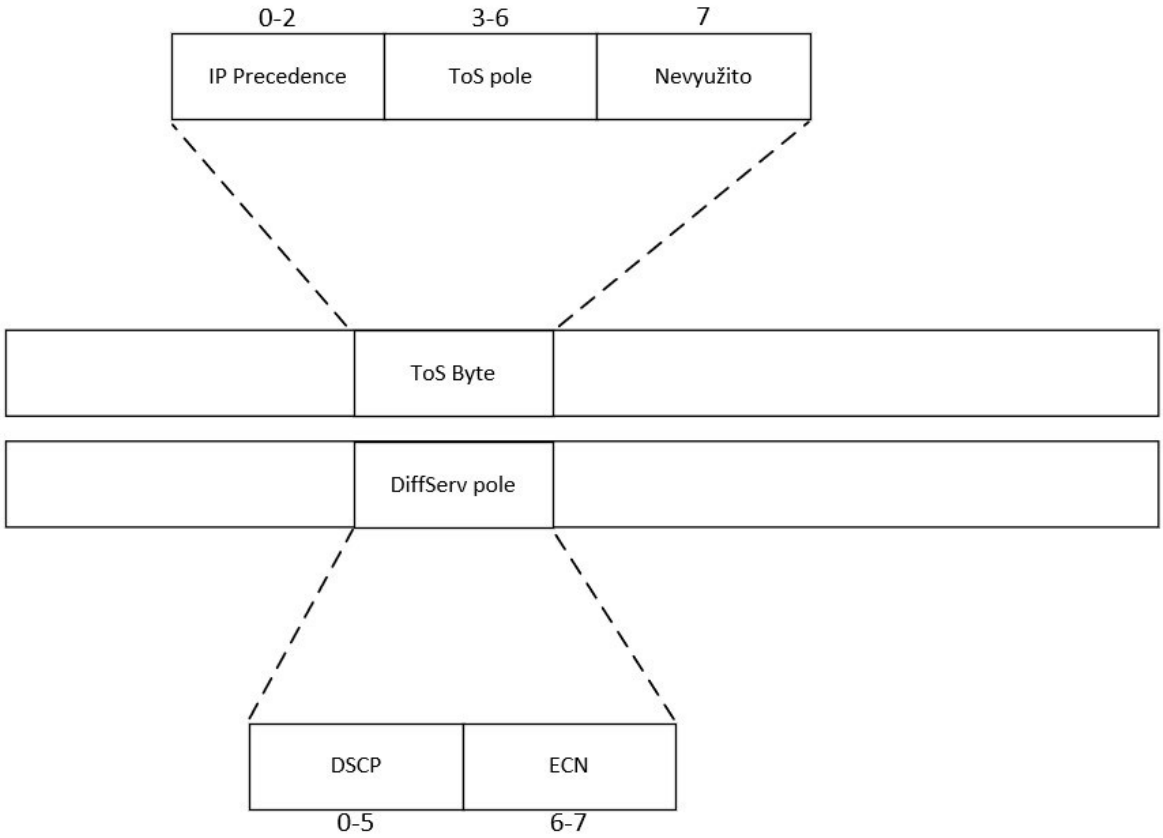
1.7.1 Integrated Services

Jedná se o model QoS, který pomocí signalizace pro jednotlivé datové toky rezervuje požadovanou propustnost nebo zpoždění. K zajištění rezervace parametrů se využívá RSVP (Resource Reservation Protocol) protokol. IntServ pomocí kontrolních mechanismů (admission control) může rozhodovat, zda je na trase dostatečná kapacita a na základě rozhodnutí rezervaci zdrojů odmítnout. V praxi je tento model využitelný pro VoIP hovory, jelikož garantuje potřebné parametry na trase. Nicméně je nepraktický pro široké využití v případě velkého množství datových toků napříč internetem. Koncová zařízení musí podporovat protokol RSVP. Pokud určitá část sítě RSVP nepodporuje, je možné RSVP zprávy buď transparentně přeposlat dál, nebo provést mapování z IntServ na Diffserv [1][10].

1.7.2 **Differentiated Services**

Nejnovější a v praxi široko využívaný je model DiffServ. Používá dělení provozu do tříd. Při vstupu do sítě využívající DiffServ dochází ke značení paketů. Na základě tohoto označení se pak zařízení v síti rozhodují, jakým způsobem budou s pakety zacházet. Tento způsob zacházení s pakety se stejnou DSCP hodnotou se nazývá Per-Hop Behavior. Pakety se stejnou DSCP hodnotu mají označení Behavior Aggregate.

Původní myšlenka při návrhu IP protokolu byla, že ke značení paketů podle typu dat bude sloužit ToS byte. Ten se skládá ze 3bitového pole IP Precedence, který slouží k odlišení typu provozu. Čím vyšší je IP Precedence hodnota, tím je provoz důležitější. Následující 4 bity, neboli ToS pole, nastavením na hodnotu 1 uvádí zvýšené požadavky na propustnost, zpoždění, spolehlivost, nebo cenu. Poslední bit v ToS bytu je nevyužit. Nicméně s příchodem DiffServ modelu došlo k modifikaci původního ToS bytu na pole DS (resp. DiffServ pole). IP Precedence a ToS pole bylo nahrazeno 6bitovou DSCP hodnotou. Poslední 2 bity v novém DiffServ poli v IP záhlaví slouží k signalizaci zahlcení v síti (Explicit Congestion Notification) [1][10].



Obrázek 1.1: ToS Byte a DiffServ pole v IP paketu [1]

Nicméně některá zařízení doteď nepodporují nové DiffServ pole s DSCP hodnotou. Z tohoto důvodu je možné zajistit zpětnou kompatibilitu DSCP vůči IP Precedence. To je možné vhodnou volbou prvních tří bitů u DSCP hodnoty. Takto vhodně zvolena DSCP hodnota se jmenuje class selector (CS). Například hodnota CS7 je vhodně zvolena DSCP = 111000 a IP Precedence = 111.

Pro úplné pochopení DSCP hodnot je třeba zmínit ještě dva pojmy. Expedited Forwarding (EF) a Assured Forwarding (AF) [1][10].

Expedited Forwarding

EF definuje typ provozu, pro který je požadováno nejnižší zpoždění, variabilita zpoždění, ztrátovost paketů a garantovaná propustnost. Zajištění nejlepších parametrů je dosaženo zvolením vhodných metod obsluhy paketových front. Přenosová rychlost je sice garantovaná, ale nesmí být překročena. K omezení se využívá traffic policing [1][10].

Assured Forwarding

AF třídí pakety do 4 tříd, kterým ve směrovačích přísluší 4 paketové fronty. Pakety se dále dělí podle pravděpodobnosti zahození v případě užití mechanismu bránícího zahlcení front (RED, WRED). Hodnota AF je ve formátu AFxy, kde x definuje číslo třídy (resp. fronty) a číslo y značí pravděpodobnost zahození (čím vyšší hodnota, tím vyšší pravděpodobnost zahození) [1][10].

2 Internet Protocol Security

IPsec VPN umožňuje obousměrnou autentizaci, šifrování přenášených dat a vyjednávání kryptografických metod a klíčů. Pro zabezpečení přenosu Internet Protokol Security používá 2 hlavní protokoly – Authentication Header (AH), Encapsulating Security Payload (ESP). Oba tyto protokoly jsou v praxi využívány většinou současně a oba rovněž podporují transportní a tunelovací mód.

Ze všech VPN technologií je IPsec protokol nejrozšířenější. Z toho důvodu lze jeho podporu najít na většině zařízeních i od různých výrobců, nejenom na Cisco zařízeních [2][3][4].

2.1 Základní vlastnosti virtuálních privátních sítí

Virtuální privátní sítě (VPN) umožňují vzdálené propojení účastníků sítě do jiné, zpravidla organizované sítě. Například pro dnešní dobu charakteristické připojení účastníka z jeho vlastní domácnosti do sítě jeho zaměstnavatele. Toto propojení by fyzickým způsobem pronajatými linkami bylo nejenom drahé, v některých případech ale i úplně nemožné. VPN technologie toto propojení řeší vytvořením privátního šifrovaného tunelu skrze běžně používanou internetovou infrastrukturu. Součástí VPN je řada protokolů. Samotná technologie se dělí na dva hlavní typy [2][3][4]:

- Site-to-Site VPN: Vzájemné propojení více sítí do jednoho celku.
- Remote Access VPN: Připojení klienta do sítě.

Druhé nejčastější dělení VPN technologie je podle primárně použitého protokolu:

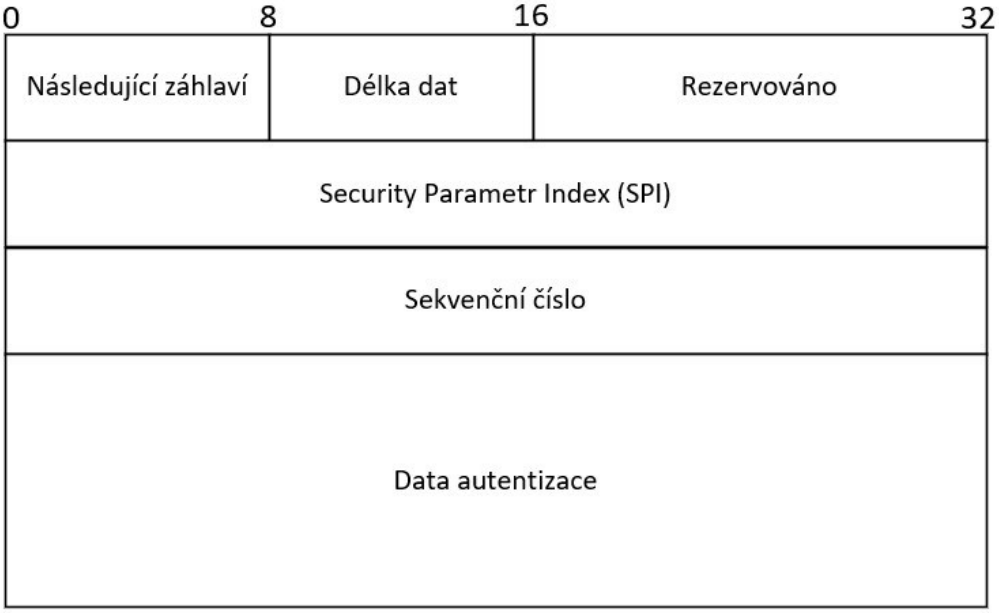
- IPsec VPN (Internet Protocol Security): Skupina protokolů pro zabezpečení IP komunikace mezi dvěma koncovými systémy.
- SSL VPN (Secure Socket Layer): Využívá Transport Layer Security (TLS) nebo Secure Socket Layer (SSL) pro zabezpečení komunikace.
- MPLS VPN (Multiprotocol Label Switching): VPN využitím MPLS technologie.

Kvalitní VPN síť by měla zajistit ochranu přenášených dat před odposloucháváním pomoci šifrování, ochranu dat proti jejich pozměňování (integrita dat) pomocí hashovacích funkcí, ochranu proti zfalšování identity komunikujícího partnera pomocí autentizace s využitím sdílených klíčů a ochranu proti útokům využívajících metodu opětovného zasílání již jednou přenesených paketů pomocí sekvenčních čísel [10].

2.2 Protokol Authentication Header

Protokol AH zajišťuje integritu a autentizaci zdroje dat. Využívá hashovací funkce a společný klíč, na kterém se při navázání spojení obě strany shodnou. Integritu dat protokol AH zajišťuje pomocí kontrolního součtu. Kvůli ochraně proti útokům využívajících opětovné posílání již jednou přenesených paketů používá v záhlaví pole se sekvenčními čísly. Pro ochranu informací AH používá hashovací algoritmy známé jako HMAC kódy (MD5, SHA, SHA256). Přestože se protokol stará o autentizaci paketu co možná nejvíce, nachází se zde pole, jejichž hodnota nelze předem příjemcem predikovat. Tato pole jsou nazývána jako proměnlivá pole. Proměnlivá pole jsou AH protokolem nechráněna. Při zpracování příchozího paketu se za původní IP záhlaví vkládá AH záhlaví, které musí být umístěno před záhlaví

jakéhokoli protokolu vyšší vrstvy. Zapouzdření pomocí AH může být provedeno v transportním nebo tunelovacím módu [4][5][6].

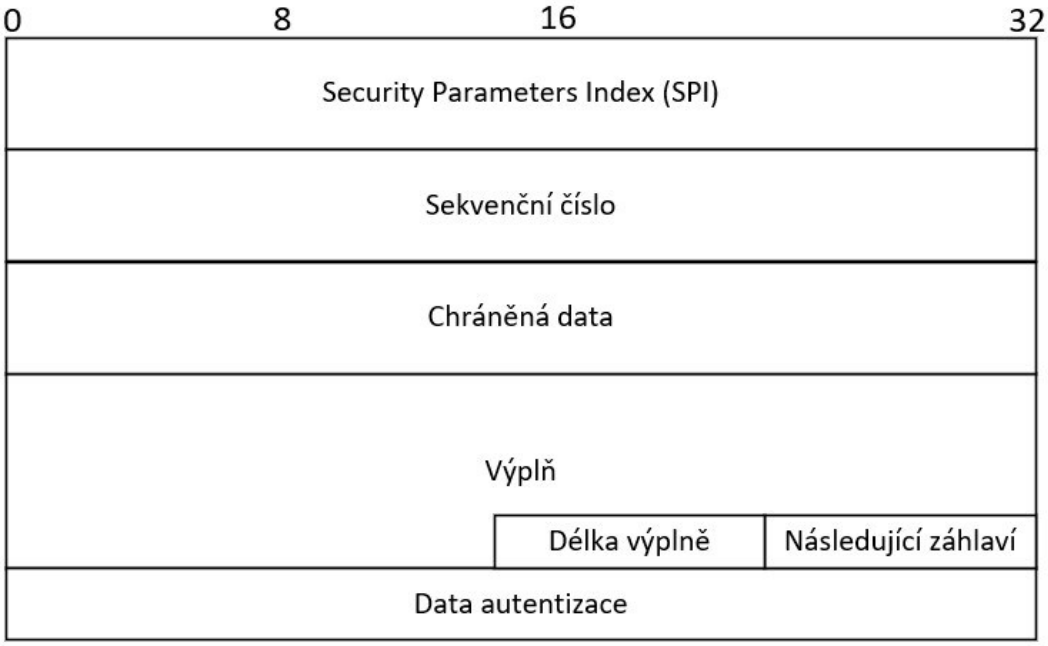


Obrázek 2.1: Záhlaví AH protokolu [5]

- Následující záhlaví: Pole maximálně 8 bitů dlouhé identifikující typ záhlaví, které následuje dané AH záhlaví.
- Délka dat: 8 bitů dlouhé pole, které udává délku Dat autentizace.
- Rezervováno: 16bitové pole rezervováno pro budoucí využití. Odesílatel nastavuje na hodnotu 0, příjemce ignoruje.
- Security Parametr Index (SPI): V kombinaci zdrojové adresy, cílové adresy a bezpečnostním protokolem (AH, ESP) identifikuje bezpečnostní asociaci (SA).
- Sekvenční číslo: Ochrana proti útokům opětovným posíláním paketů (replay attack).
- Data autentizace: Kontrolní součet přenášených dat zajišťující integritu přenosu. Generováno HMAC algoritmem.

2.3 Protokol Encapsulating Security Payload

Protokol ESP na rozdíl od protokolu AH poskytuje datům jistou důvěrnost. Kromě důvěrnosti ESP zajišťuje i autentizaci, kontrolu integrity a ochranu před replay útoky. V transportním módu ESP protokol datovou část paketu zašifruje, doplní jí ESP záhlavím a původní IP záhlaví přesune za ESP záhlaví. V tunelovacím módu protokol ESP zašifruje původní datovou část i s původním IP záhlavím, přidá ESP záhlaví a za ESP záhlaví přidá nové IP záhlaví obsahující adresy obou konců tunelů. V porovnání s AH může být protokol i rychlejší na zpracování. Umožňuje totiž kontrolu důvěrnosti šifrováním bez použití mechanismů zajišťující integritu spojení. ESP protokol může být implementován buď samostatně, nebo v kooperaci s AH protokolem [7].



Obrázek 2.2: Záhlaví ESP protokolu [7]

- Security Parameters Index (SPI): Identifikuje použité SA, se kterým je příchozí paket svázán.
- Sekvenční číslo: Kontrola před útoky opětovným posíláním paketů.
- Chráněná data: Obsahuje původní data paketu chráněná ESP protokolem.
- Výplň: Prostý text, který je vyžadován některými blokovými šifrovacími algoritmy.
- Délka výplně: Definuje délku výplně.
- Následující záhlaví: Popisuje, jakého typu jsou data v položce ,,chráněná data".
- Data autentizace (Integrity Check Value): Obsahují kontrolu integrity paketu.

2.4 IPsec Security Association

SA je soubor specifikací IPsec technologie. Tyto specifikace si mezi sebou vyjednávají zařízení, která navazují IPsec spojení. Soubor obsahuje informace o typu autentizace, šifrování a rovněž taky informace o IPsec protokolu zajišťující zapouzdření. IPsec SA se pro každý směr a pro každý protokol (ESP, AH) realizuje zvlášť [8].

2.4.1 Proces vyjednání IPsec Security Associaton

Kromě samotné IPsec SA je při procesu vytvořena rovněž Internet Key Exchange Security Associaton (IKE SA). V prvním kroku jedna strana (Alice) příjme paket, který má být zabezpečen pomocí IPsec. Zahájí se tedy vyjednávání IKE SA. Zde dochází ke vzájemné autentizaci obou stran (Alice – Bob) pomocí předsdílených klíčů nebo certifikátů a dokončení vytváření IKE SA mezi oběma stranami. Vytvořené zabezpečené IKE SA spojení slouží pro vyjednávání dvou protisměrných IPsec SA. Vyjednají se používané symetrické šifry, hashovací funkce, hodnota SPI a pomocí Diffie-Hellmanova algoritmu se vytvoří sdílený klíč pro symetrické šifrování. Alici odeslaný zašifrovaný paket Bob po přijetí přiřadí konkrétní SA pomocí SPI hodnoty. Díky přiřazení paketu k patřičné SA Bob ví, jakým způsobem má pomocí vyjednaného klíče paket dešifrovat. Platnost SA bývá v praxi omezená, a to buď časově, nebo množstvím přenesených dat. Poté je třeba vytvořit novou SA [8][10].

2.5 Internet Security Association and Key Management Protocol

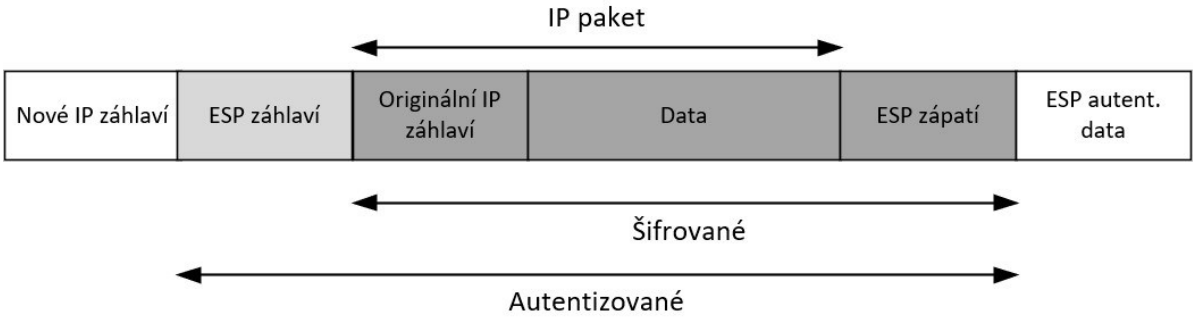
ISAKMP je obecný protokolový rámec, který zajišťuje autentizaci obou konců IPsec tunelu, vytvoření, následnou údržbu a ukončení IPsec SA. Rovněž také zajišťuje vytváření a výměnu dynamických klíčů (Diffie-Hellmanův algoritmus) [10].

2.6 Diffie-Hellmanův algoritmus

DH algoritmus slouží k vytvoření a bezpečné výměně sdíleného tajného klíče pro symetrické šifrování přenášených dat. Algoritmus se dělí do tří skupin (group 1, group 2, group 5) podle délky vytvořeného tajného klíče. Pátá skupina pak vytváří nejdelší klíč, který je využíván symetrickou šifrou AES [10].

2.7 Tunelovací mód

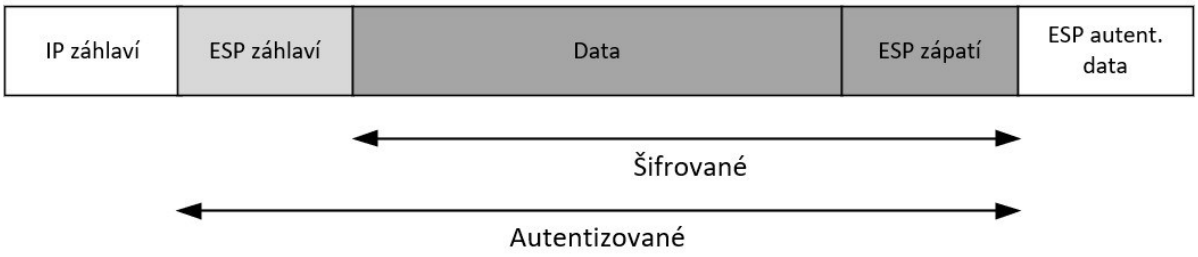
S tunelovacím módem je IPsec protokolem chráněn celý IP paket. To znamená, že odesílající VPN směrovač šifruje celý originální paket a následně přidává nové IP záhlaví obsahující adresy odesílatele a příjemce v rámci tunelu. Skrytí původních IP adres zvyšuje bezpečnost komunikace. Zpravidla se používá mezi výchozími bránami dvou sítí propojených skrze VPN (gateway-to-gateway) [2][5][9].



Obrázek 2.3: IPv4 paket zapouzdřen užitím ESP v tunelovacím módu [9]

2.8 Transportní mód

Transportní mód se nejčastěji využívá pro šifrovanou komunikaci mezi dvěma klienty (peer-to-peer). Na rozdíl od tunelovacího módu transportní mód nezabaluje původní IP paket do paketu nového. Místo toho pouze zašifruje (ESP) původní data paketu a originální záhlaví IP paketu přesune před ESP záhlaví. Tento mód je v porovnání s tunelovacím módem méně bezpečný, protože v rámci komunikace figurují skutečné IP adresy odesílatele a příjemce. Naopak se transportní mód často využívá ve spojení s Generic Routing Encapsulation (GRE) protokolem, kde se šifruje celý takto vytvoření GRE tunel [2].



Obrázek 2.4: IPv4 paket zapouzdřen užitím ESP v transportním módu [9]

2.9 Protokol Generic Routing Encapsulation

GRE je tunelovací protokol pracující na síťové vrstvě. Je často využíván ve spojení s IPsec protokolem, a to z toho důvodu, že na rozdíl od IPsec umožňuje tunelovat ne-unicastový provoz. Zapouzdřovaná data na druhou stranu ale nešifruje, takže nezajišťuje bezpečnost přenášených dat tak, jak to dělá IPsec. Princip fungování GRE protokolu je podobný jako IPsec protokol v tunelovacím módu. Přenášené pakety zapouzdří pomoci GRE záhlaví a přidá nové IP záhlaví, ve kterém figurují IP adresy začátku a konce tunelu. Následně je paket znova zapouzdřen pomocí IPsec protokolu v transportním módu s využitím ESP nebo AH. Využití transportního módu je umožněno díky tomu, že původní IP záhlaví je již skryto [10].

Obrázek 2.5: Struktura paketu zapouzdřeného pomocí GRE a ESP [10]

2.10 Klasifikace IPsec paketů

S užitím IPsec technologie přichází i jisté komplikace týkající se klasifikace přenosu. Jelikož IPsec protokol původní IP pakety celé šifrují, stávají se tak pro QoS klasifikační nástroje nečitelnými. Ve skutečnosti směrovače dokonce ztrácejí schopnost vůbec rozlišit, kde záhlaví originálního paketu začíná a končí.

Směrovače se s tímto problémem vypořádaly tak, že hodnotu ToS z paketu originálního automaticky kopírují a následně pak zapisují do záhlaví nového IPsec paketu. Stejný postup je použit i v případě GRE tunelu, s tím rozdílem, že se ToS hodnota kopíruje hned dvakrát [1][2].

2.11 Pre-classification IPsec paketů

Ve výchozím nastavení zařízení provádějící klasifikaci paketů pouze kopíruje hodnotu pole ToS do nového IP záhlaví, tudíž klasifikace na bázi DSCP funguje podle předpokládání. Nicméně v momentě, kdy klasifikace probíhá pomocí jiného parametru, než je ToS (například zdrojová nebo cílová IP adresa) je vyžadováno využití pre-classification mechanismu. V normálním případě je totiž veškerý jiný obsah, než ToS již zašifrovaný a klasifikační nástroje k nim nemají přístup. Proto dochází ve směrovačích ke klonování hlavičky příchozího paketu. Tato naklonovaná data si směrovače uchovávají v paměti, díky čemuž je možné přenos klasifikovat i po procesu tunelování a šifrování. V každém případě možnost před klasifikace je umožněna jen a pouze na zařízení, které šifrování originálního paketu provádí. Naklonovaná data tedy nejsou součástí přenášených dat a jakákoliv klasifikace po přenosu už je možná opět pouze na základě ToS parametru. Proto se jako osvědčený postup ukázal tuto před klasifikaci na VPN směrovačích nastavovat preventivně. Před klasifikace ve výsledku nemá žádný zásadní vliv na výkon směrovače [2].

2.12 Komplikace s MTU

Případné problémy vzniklé komplikacemi s MTU (Maximum transmission unit) mohou mít vážný dopad na konektivitu v síti. Proto je v praxi nutné implementovat ověřené postupy, které se s těmito

komplikacemi umějí vypořádat. Tyto komplikace se mohou projevit i ve zdánlivě fungující síti, kde většina služeb může fungovat bez jakýchkoliv problémů [2].

2.12.1 **GRE a komplikace s MTU**

V případě, že na rozhraní VPN směrovače dorazí paket, který se blíží k maximální hodnotě MTU = 1500 B směrovač přistoupí k zapouzdření originálního paketu přidáním několika nových hlaviček. Toto ve výsledku vede k navýšení celkové velikosti paketu nad rámec maximálního MTU. V momentě, kdy je v příchozím paketu nastaven bit DF (Don't Fragment) na 1, směrovač nemá jinou možnost než paket zahodit. MTU hodnota nelze jednoduše navýšit, jelikož maximální MTU ethernet rámce je 1518 B, což je možné řešit jumbo rámcem (9000 B), který ovšem ne každá síť podporuje [2].

2.12.2 **IPsec a komplikace s MTU**

IPsec se oproti GRE chová trochu odlišně v případě velkého příchozího paketu. Zatímco GRE tunel požaduje 24 B hlavičku IPsec přidává hlavičku s maximem 58 B. Příchozí paket se ale po zapouzdření fragmentuje na dvě části. Tyto části jsou pak na druhé straně opět složeny dohromady a odeslány v původní podobě. IPsec technologie má tedy zabudovaný mechanismus, který se s nadměrnými MTU umí vypořádat. V IOS navíc IPsec vždy provádí path MTU discovery (PMTUD). Tato metoda umožňuje IPsec VPN směrovači i v případě, že je odchozí MTU hodnota přijatelná zjistit, jestli se v cestě nenachází směrovač, který by mohl způsobit problémy. Na základě výsledku je pak směrovači umožněna preventivní fragmentace paketu, a to i přesto, že je DF bit nastaven na 1.

Mezi oběma přístupy je tedy zásadní rozdíl v tom, že IPsec fragmentovaný paket umožňuje na druhém konci znovu spojit dohromady. Proto je možná fragmentace paketu nehledě na hodnotu DF bitu [2].

2.12.3 **Úprava maximální velikosti segmentu**

Jedna z nejčastějších metod řešících komplikace s MTU je úprava parametru Maximum Segment Size (MSS). Tento parametr definuje maximální velikost užitečných dat, kterou je host schopný přijmout v jednom TCP/IP datagramu. Během navázání TCP spojení si obě strany sdělí hodnotu MSS parametru. Je pak zodpovědností každého hosta limitovat velikosti datagramů v každém TCP segmentu tak, aby byla splněna druhou stranou sdělena hodnota MSS. Parametry MSS a MTU jsou si dosti podobné. Rozdíl je v tom, že MSS udává velikost pouze užitečné zátěže IP paketu, zatímco MTU udává velikost celého IP paketu se všemi záhlavími.

Samotnou úpravu parametru MSS pak provádí jeden ze směrovačů v IPsec spojení. Při navazování spojení k tomuto směrovači přijde paket od konkrétního hosta s MSS o velikosti 1460 B. Směrovač pak cíleně hodnotu MSS sníží na hodnotu 1378 B. Na této snížené hodnotě MSS se obě strany shodnou. Tento postup tedy umožňuje krajním směrovačům dané IPsec linky navázat s hosty takové spojení, které je pak schopné projít IPsec/GRE tunelem bez nutností fragmentace paketů. Proces je to však výpočetně náročný. Proto je důležité postupovat uvážlivě při rozhodování, který ze směrovačů bude tuto úpravu MSS provádět. Rovněž je nutné zmínit, že úprava MSS lze provést pouze v případě TCP spojení. Na druhou stranu není příliš pravděpodobný scénář UDP pakety s MTU hodnotou vyšší, než je limit [2].

2.13 Komprese v IPsec VPN

Kompresní metody k celkovému zkvalitnění služeb a snížení odezvy systému je doporučeno používat kdykoliv a kdekoliv to situace umožňuje. Užité kompresní metody mezi dvěma systémy propojenými IPsec VPN technologií zvyšují propustnost a rovněž snižují odezvu VPN linek [2].

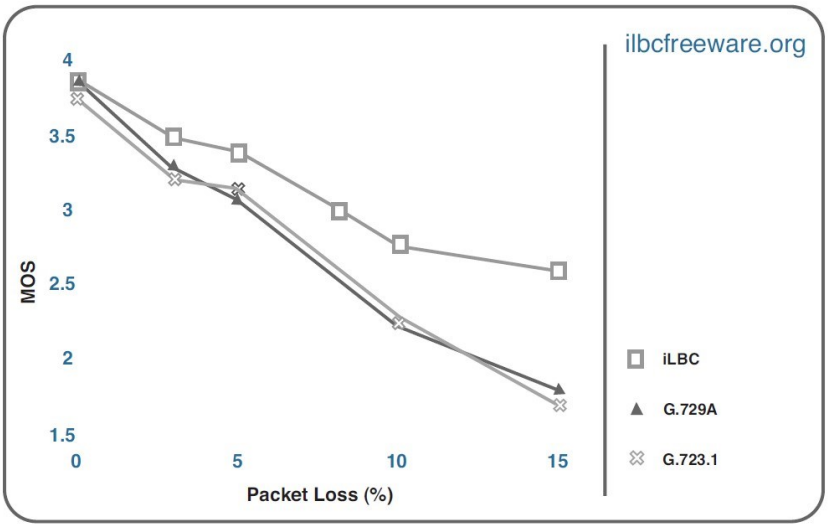
2.13.1 Optimalizace TCP pomocí WAAS

Jedná z klíčových technologií, která nezanedbatelným způsobem zkvalitňuje VPN WAN je Cisco Wide Area Application Services (WAAS). WAAS technologie používá různé druhy kompresních metod typu LZ komprese, Date Redundancy Elimination (DRE), které jsou založeny na kompresi redundantních dat. Tyto metody výrazně redukují množství dat, které je pak přes WAN VPN přeposíláno, což nejenom že umožňuje efektivnější využití šířky pásma, rovněž ale snižuje round-trip time (RTT) zpoždění. Zmíněné kompresní technologie mají výrazný dopad na celkovou uživatelskou QoE (Quality of Experience). I toto metody ale mají jistá omezení. Z výhod WAAS těží pouze komunikace na bázi TCP spojení. Rovněž je nutné dohlížet na správnou implementaci těchto služeb. Zvlášť pak na to, aby komprese probíhala pouze na užitečných datech paketu a nezasahovala do informací obsažených v záhlavích. Při kompresi těchto užitečných dat by totiž mohlo dojít ke znemožnění použití QoS klasifikačních metod [2].

2.13.2 Využití kodeků

Jelikož metody WAN akcelerátorů, jako jsou třeba metody WAAS ovlivňují pouze TCP přenos a žádným způsobem neovlivňují přenos UDP, je na zvážení užití hlasových kodeků. Ke zvětšení množství možných přenesených hlasových služeb administrátoři často volí hlasové kompresní kodeky např. ILBC nebo G.729. Tyto dva kompresní kodeky v porovnání s typickým G.711 kodekem redukují požadovanou šířku pásma z původních 80 kbit/s na 24 kbit/s. Ve výsledku pak VPN linkou může projít až třikrát více hovorů. Jak už to tak ale bývá, řada výhod přináší i jisté nevýhody. V momentě komprese velkého množství hovorových dat, do malého množství paketů se tento přenášený hovor stává velmi náchylným na chyby vzniklé ztrátou některých přenášených paketů. Problémy vzniklé ne úplně spolehlivou sítí internet řeší implementace jistých mechanismů. Například kodek G.729 tyto potíže řeší vyplněním mezery vzniklé ztrátou paketu. Řekněme, že z přenášených čtyř paketů hovoru při přenosu ztratíme v pořadí třetí paket. Mechanismus pak na místo třetího, ztraceného paketu přehraje znova paket druhý. Tímto způsobem lze oklamat zařízení na přijímací straně vyplněním mezery nehledě na to, že takto nahrazená data nejsou korektní. Díky této metody jsme schopní tolerovat až průměrnou pětiprocentní ztrátovost paketů hovoru.

Ještě o něco lépe si v praxi vede nový open source kodek Internet Low Bitrate Codec (iLBC), navržen společností Global IP Solutions (Google). iLBC s nabízenou přenosovou rychlostí 15,2 kbit/s nebo 13,33 kbit/s je vhodnou alternativou G.729 kodeku v případě ztrátových VPN linek. V případě nulové ztrátovosti (Packet Loss) dosahuje podobné Mean Opinion Score (MOS) hodnoty jako G.729. S rostoucí paketovou ztrátovosti si ale iLBC vede výrazně lépe [2].



Obrázek 2.6: MOS v závislosti na paketové ztrátovosti [2]

2.14 cRTP a IPsec

S šifrovaným přenosem v IPsec sítích souvisí nekompatibilita služeb, které pro správnou funkčnost vyžadují přístup k originálnímu paketu. Ten je ale kvůli šifrovanému přenosu v IPsec zcela nečitelný. Jedna z funkcí, která se spoléhá na transparentní pakety je Compressed Real-Time Protocol (cRTP). V době, kdy je volána komprese je původní RTP paket již zašifrovaný. Takto zašifrovaný paket není možné komprimovat. Z tohoto důvodu je cRTP a IPsec zcela nekompatibilní [2].

2.15 Antireplay ochrana

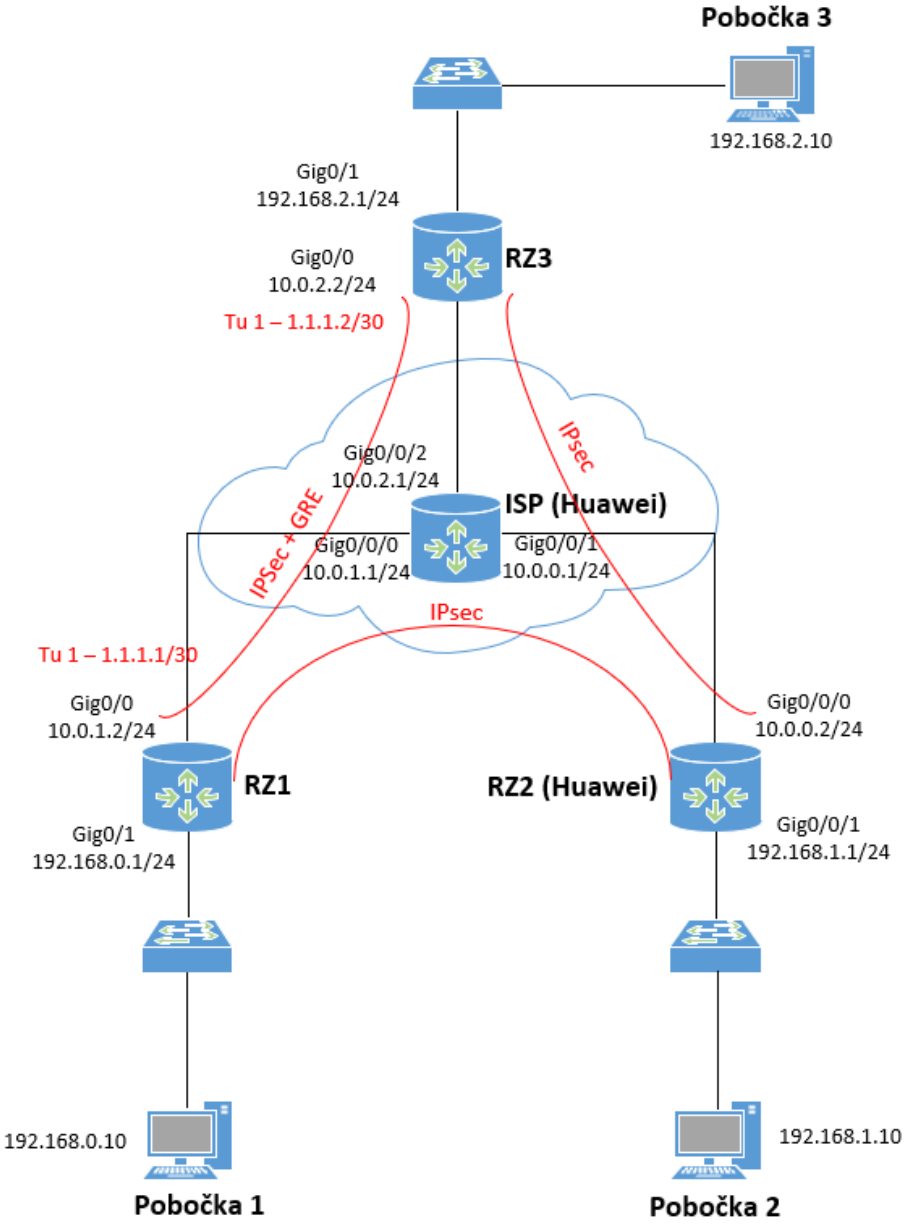
Technologie IPsec zajišťuje integritu zpráv, čímž zajišťuje ochranu (connectionless integrity) proti útokům duplicitními pakety. Rovněž poskytuje částečnou integritu sekvence, čímž těmto duplikacím předchází. V případě autentizace ESP protokolu přijímací strana IPsec spojení každé SA verifikuje, jestli je paket přijatý právě jednou. K tomu slouží 64 paketů dlouhé posuvné okno (sliding window), které vymezuje paměť směrovače právě pro kontrolu příchozích paketů. Odesílající směrovač všem zašifrovaným paketům přiřadí unikátní sekvenční číslo. Při plnění posuvného okna vždy dochází ke kontrole, zda se sekvenční číslo příchozího paketu v okně již nachází nebo ne. V momentě, kdy je kontrolován paket s vyšší hodnotou, než je nejvyšší hodnota v okně je paket přijat a okno se posune doprava. Přijde-li naopak paket, který má sekvenční číslo menší, než je nejmenší možná hodnota tohoto posuvného okna, paket se zahodí.

Komplikace ale přicházejí s konvergovanými IPsec VPN sítěmi užívající QoS služby. Zde probíhá prioritizace provozu různých služeb, což vede k rozhození pořadí paketů. Proto může dojít k zahození paketu, a to i přesto, že výše uvedené podmínky splňuje. V jistých případech se tomuto nežádoucímu vlivu vyhnout. Například v implementacích čistých IPsec tunelů bez použití GRE technologie můžeme pro každý typ přenosu navázat separátní SA [2].

3 Implementace IPsec VPN

3.1 Návrh a sestavení topologie

Sestavená topologie pracuje se scénářem, kdy zákazník potřebuje skrze veřejnou síť propojit šifrovanými tunely své tři vzdálené pobočky. Jak již bylo zmíněno, IPsec je široce využívaný otevřený standard, který řada výrobců implementuje do svých zařízení. Různá zařízení různých výrobců jsou pak schopny navazovat tato zabezpečená spojení i mezi sebou. Tři podsítě zákazníka budou navzájem propojeny pomocí IPsec protokolu s využitím ESP protokolu na zapouzdření provozu. Dva hraniční směrovače zákazníka RZ1 a RZ2 jsou od dodavatele Cisco, konkrétně se jedná o modely Cisco 2901. Třetí hraniční směrovač RZ2 bude od dodavatele Huawei, model AR3260. Ve středu topologie se nachází Huawei AR2200, který reprezentuje síť poskytovatele internetu. Síť rovněž obsahuje tři Cisco přepínače a jeden Cisco rozbočovač, který bude využíván pouze pro testovací účely.



Obrázek 3.1: IPsec + GRE topologie

Cílem této testovací topologie bude zejména dokázání vzájemné kompatibility při sestavování IPsec VPN spojení mezi zařízeními dvou různých výrobců. Takto sestavená VPN spojení se v praxi běžně využívají. Při konfiguraci IPsec VPN je potřeba dbát na to, aby byly mechanismy zajišťující zabezpečení provozu na obou koncích nakonfigurovány shodně. V případě, kdy konfigurace parametrů probíhá na obou zařízeních odlišným způsobem a zároveň musí dát stejný výsledek mohou nastat komplikace.

Mezi směrovači RZ1 a RZ3 bude dále pro testovací účely navázán GRE tunel.

3.2 Konfigurace IPsec VPN topologie

Pro zkrácení výstupu práce zde nebude popsán postup konfigurace jednotlivých rozhraní nebo konfigurace směrovacího protokolu. Mezi hraničním směrovačem zákaznické sítě a sítě poskytovatele bude připojen rozbočovač s počítačem, který bude pomocí Wireshark programu odchytávat pakety, které následně poslouží k ověření správnosti konfigurace. Rovněž zde nebude uvedena konfigurace směrovače RZ3, jelikož se oproti konfiguraci směrovače RZ1 bude lišit pouze v konfigurovaných IP adresách. Celý konfigurační soubor bude následně přiložen v příloze, viz [A].

3.2.1 Konfigurace směrovače RZ1

V prvé řadě je nutné definovat ISAKMP politiku. Zde budou nastaveny parametry pro šifrování, vzájemnou autentizaci obou konců a použita skupina Diffie-Hellmanova algoritmu. V případě testovací topologie je využito šifrování AES s 256bitovým klíčem. Autentizace obou konců bude probíhat pomocí předem sdíleného hesla. Toto heslo musí být následně na obou stranách nastaveno rovněž stejné. V poslední řadě je pak nakonfigurováno využití páté skupiny Diffie-Hellmanova algoritmu, a to hlavně z důvodu využití šifry AES-256. ISAKMP politika rovněž umožňuje konfiguraci časového intervalu, po jehož uplynutí dojde k vyjednání nových bezpečnostních asociací.

```
RZ1(config)#crypto isakmp policy 10
RZ1(config-isakmp)#encr aes 256
RZ1(config-isakmp)#authentication pre-share
RZ1(config-isakmp)#group 5
```

Dále je třeba uvést heslo pro autentizaci a IP adresu rozhraní druhého konce tunelu, ke kterému se bude zařízení autentizovat. V případě RZ1 směrovače to bude rozhraní směrovače RZ2 a RZ3.

```
RZ1(config)#crypto isakmp key heslo address 10.0.0.2
RZ1(config)#crypto isakmp key heslo address
10.0.2.2
```

Následně se pomocí transform-set příkazu definují parametry pro zabezpečování průchozích dat. Dále je zde možné nastavit, zda bude IPsec fungovat v transportním, nebo tunelovacím módu. Ve výchozím stavu je nastaven tunelovací mód. V případě GRE tunelu bude použit transportní mód.

```
RZ1(config)#crypto ipsec transform-set myset esp-aes 256 esp-sha-hmac
RZ1(config)#crypto ipsec transform-set myset1 esp-aes 256 esp-sha-hmac
RZ1(cfg-crypto-trans)#mode transport
```

Provoz, který bude procházet šifrovanými tunely je nutné definovat pomocí access-listu. V případě GRE tunelu to budou uvedeny IP adresy fyzických rozhraní směrovačů, na kterých je navázán GRE tunel. V případě tunelu RZ1 a RZ2 budou uvedeny IP adresy daných poboček.

```
RZ1(config)#access-list 100 permit gre host 10.0.1.2 host 10.0.2.2
RZ1(config)#access-list 101 permit ip 192.168.0.0 0.0.0.255 192.168.1.0 0.0.0.255
```

Následně se pomocí příkazu *crypto-map* spojí předešlé konfigurace dohromady. Ustanoví se adresa druhého konce tunelu, transform-set, který definuje parametry zabezpečení provozu a access-list, který definuje, který provoz má byt zabezpečen.

```
RZ1(config)#crypto map ipsec_map 10 ipsec-isakmp
RZ1(config-crypto-map)#set peer 10.0.2.2
RZ1(config-crypto-map)#set transform-set myset1
RZ1(config-crypto-map)#match address 100

RZ1(config)#crypto map ipsec_map 20 ipsec-isakmp
RZ1(config-crypto-map)#set peer 10.0.0.2
RZ1(config-crypto-map)#set transform-set myset
RZ1(config-crypto-map)#match address 101
```

Pro GRE tunel je třeba vytvořit tunelovací rozhraní s IP adresou pro daný virtuální tunel. Je vhodné upravit maximální velikost segmentu. To z toho důvodu, že po připojení nového záhlaví by paket s maximální hodnotou MTU mohl překročit maximální limit daného spoje. Paket by byl následně zahozen. Dále je zde uvedena zdrojová a cílová IP adresa fyzických rozhraní, na kterých dochází k vytvoření tunelu.

```
RZ1(config)#interface Tunnel1
RZ1(config-if)#ip address 1.1.1.1 255.255.255.252
RZ1(config-if)#tunnel source GigabitEthernet0/0
RZ1(config-if)#tunnel destination 10.0.2.2
```

Nyní je třeba nakonfigurovat statickou cestu pro síť pobočky 3. Jako next hop IP adresa bude uvedena IP adresa druhého konce GRE tunelu.

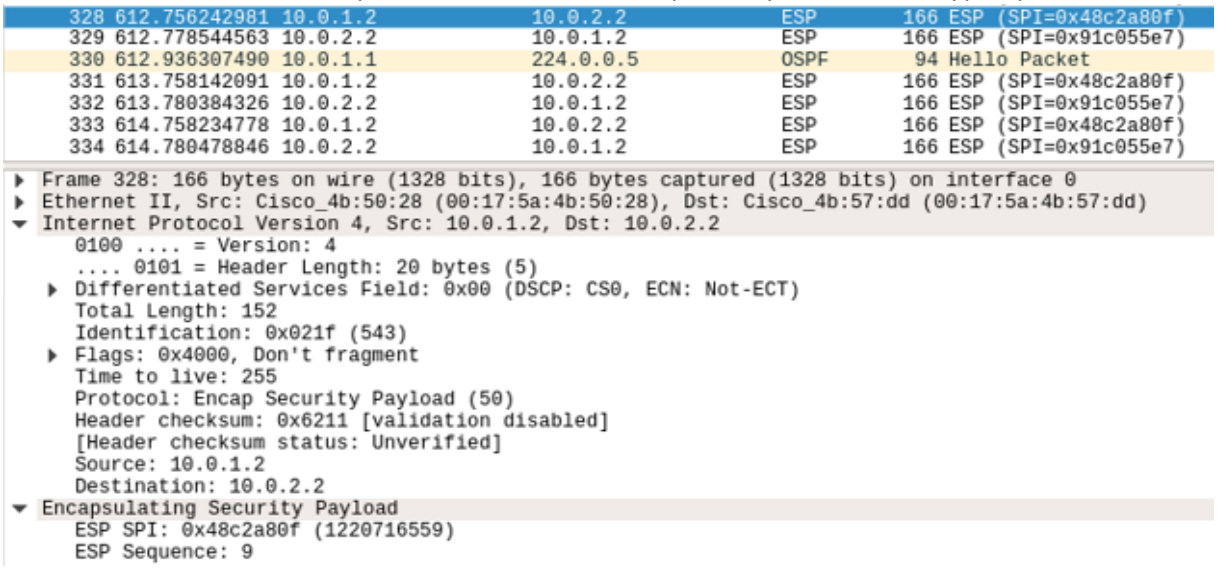
```
RZ1(config)#ip route 192.168.2.0 255.255.255.0 1.1.1.2
```

Jako poslední krok je nutné nakonfigurovat vytvořenou crypto-map na fyzické rozhraní směrovače RZ1.

```
RZ1(config)#interface GigabitEthernet0/0
RZ1(config-if)#crypto map ipsec_map
```

3.2.2 Ověření konfigurace IPsec VPN na RZ1 směrovači

Mezi směrovače RZ1 a RZ3 je připojen rozbočovač s počítačem, který pomocí Wireshark programu potvrdí, že přenášené pakety jsou zapouzdřeny a zašifrovány pomocí ESP protokolu. Rovněž lze ověřit funkčnost GRE protokolu. Průchozí provoz bude mít zdrojovou IP adresu začátku GRE tunelu 10.0.1.2 a cílovou IP adresu 10.0.2.2. Správnost lze rovněž ověřit pomocí příkazu *show crypto ipsec sa*.



Obrázek 3.2: Ověření odchyceného provozu RZ1-RZ3

3.2.3 Konfigurace směrovače RZ2

Pro zkrácení výstupu této práce bude na RZ2 směrovači uvedena pouze konfigurace IPsec VPN tunelu mezi RZ2 a RZ1 směrovačem. Konfigurace v případě tunelu RZ2 a RZ3 se bude lišit pouze v konfigurovaných IP adresách.

Na úvod opět dojde nejprve ke konfiguraci parametrů, na základě kterých se mezi zařízeními vyjednají IKE SA. Jedná se o šifrovací metodu AES s 256bitů dlouhým klíčem a skupinu Diffie-Hellmanova algoritmu.

```
[RZ2]ike proposal 1
[RZ2-ike-proposal-1]encryption-algorithm aes-cbc-256
[RZ2-ike-proposal-1]dh group5
```

Následně dojde ke konfiguraci parametrů, které budou využívány ve vyjednaném IPsec SA spojení. Jedná se o šifrování AES a hashovací funkce SHA. V tomto segmentu konfigurace lze rovněž uvést, zda se jedná o transportní, nebo tunelovací mód. Tunelovací mód je nastaven ve výchozím stavu.

```
[RZ2]ipsec proposal tran1
[RZ2-ipsec-proposal-tran1]esp authentication-algorithm sha1
[RZ2-ipsec-proposal-tran1]esp encryption-algorithm aes-256
```

Nyní stejně jako v případě Cisco směrovačů je třeba pomocí access-listu definovat, který provoz bude zabezpečován pomocí IPsec tunelů.

```
[RZ2]acl 3000
[RZ2-acl-adv-3000]rule permit ip source 192.168.1.0 0.0.0.255 destination 192.168.0.0 0.0.0.255
```

V této části konfigurace dojde ke konfiguraci druhého konce tunelu. Je nutné uvést typ autentizace, předsdílené heslo a také IP adresu konce tunelu, se kterým bude docházet k autentizaci a k následnému

navázání tunelu. Tomuto nastavení následně přidáme již připravený ike proposal. Na obou stranách tunelů musí být nastavené stejné předsdílené heslo.

```
[RZ2]ike peer cisco2 v1
[RZ2-ike-peer- cisco2]pre-shared-key simple heslo
[RZ2-ike-peer- cisco2]ike-proposal 1
[RZ2-ike-peer- cisco2]remote-address 10.0.1.2
```

Dosud provedenou konfiguraci sloučíme do ipsec policy, kterou následně přiřadíme na konkrétní fyzické rozhraní RZ2 směrovače.

```
[RZ2]ipsec policy map1 2 isakmp
[RZ2-ipsec-policy-isakmp-map1-1]security acl 3000
[RZ2-ipsec-policy-isakmp-map1-1]ike-peer cisco2
[RZ2-ipsec-policy-isakmp-map1-1]proposal tran1
```

```
[RZ2] interface GigabitEthernet 0/0/0
[RZ2-GigabitEthernet0/0/0] ipsec policy map1
```

3.2.4 **Ověření konfigurace IPsec VPN na RZ2 směrovači**

Mezi směrovači RZ2-RZ1 a RZ2-RZ3 byly navázány tunely uvedeným způsobem. Správnost uvedené konfigurace můžeme ověřit několika *display* příkazy a rovněž pomocí programu Wireshark na počítači, který je připojen rozbočovačem.

Navázané IKE spojení lze ověřit následovně.

```
[Huawei]display ike sa
  Conn-ID  Peer         VPN   Flag(s)            Phase
 -----------------------------------------------------------------
    19    10.0.2.2      0    RD|ST              2
    17    10.0.2.2      0    RD|ST              1
    21    10.0.1.2      0    RD|ST              2
    20    10.0.1.2      0    RD|ST              1
Flag Description:
 RD--READY   ST--STAYALIVE   RL--REPLACED   FD--FADING   TO--TIMEOUT
 HRT--HEARTBEAT   LKG--LAST KNOWN GOOD SEQ NO.   BCK--BACKED UP
```

Všechna navázaná SA spojení a jejich příslušné hodnoty parametru SPI, využitý protokol pro zapouzdření dat, hashovací i šifrovací metodu lze zobrazit tímto způsobem.

```
[Huawei]display ipsec sa brief
Number of SAs:4
  Src address    Dst address      SPI    VPN  Protocol     Algorithm
-------------------------------------------------------------------------------
    10.0.2.2      10.0.0.2 2767409705     0    ESP   E:AES-256 A:SHA1-96
    10.0.0.2      10.0.1.2  226413462     0    ESP   E:AES-256 A:SHA1-96
    10.0.1.2      10.0.0.2 2940199604     0    ESP   E:AES-256 A:SHA1-96
```

```
10.0.0.2        10.0.2.2 1764986959     0    ESP   E:AES-256 A:SHA1-96
```

```
    12 3.006735886   10.0.0.2          10.0.1.2          ESP       166 ESP (SPI=0x0d7ecb96)
    13 4.008365881   10.0.1.2          10.0.0.2          ESP       166 ESP (SPI=0xaf3fe2b4)
    14 4.008950526   10.0.0.2          10.0.1.2          ESP       166 ESP (SPI=0x0d7ecb96)
    15 5.009346382   10.0.1.2          10.0.0.2          ESP       166 ESP (SPI=0xaf3fe2b4)
    16 5.009898585   10.0.0.2          10.0.1.2          ESP       166 ESP (SPI=0x0d7ecb96)

▸ Frame 16: 166 bytes on wire (1328 bits), 166 bytes captured (1328 bits) on interface 0
▸ Ethernet II, Src: Cisco_4b:57:dd (00:17:5a:4b:57:dd), Dst: Cisco_4b:50:28 (00:17:5a:4b:50:28)
▾ Internet Protocol Version 4, Src: 10.0.0.2, Dst: 10.0.1.2
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▸ Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)
    Total Length: 152
    Identification: 0x0017 (23)
  ▸ Flags: 0x0000
    Time to live: 253
    Protocol: Encap Security Payload (50)
    Header checksum: 0xa819 [validation disabled]
    [Header checksum status: Unverified]
    Source: 10.0.0.2
    Destination: 10.0.1.2
▾ Encapsulating Security Payload
    ESP SPI: 0x0d7ecb96 (226413462)
    ESP Sequence: 23
```

Obrázek 3.3: Ověření odchyceného provozu RZ2-RZ1

Na přiloženém snímku lze potvrdit použití ESP protokolu pro zapouzdření a šifrování provozu. Dále lze zpozorovat, že v polích zdrojové a cílové IP adresy figurují adresy rozhraní, na kterých je navázán IPsec tunel. IP adresy komunikujících klientů jsou skryté. Na snímku lze zpozorovat hodnota v poli ESP SPI: 226413462. Tuto hodnotu můžeme srovnat s posledním přiloženým výpisem zobrazující navázána IPsec SA spojení.

3.3 Konfigurace maximálního MTU

Při implementaci samotné IPsec VPN technologie, nebo IPsec + GRE je často žádoucí úprava MTU, nebo maximální velikosti segmentu. V teoretické části práce bylo naznačeno, že zařízení výrobce Cisco mají implementovaný mechanismus, který je schopen tyto potíže řešit samostatně. Nicméně toto řešení funguje pouze na čisté IPsec tunely. Při implementaci IPsec s GRE je nutná úprava buď MTU, nebo maximální velikosti segmentu (MSS). Tento mechanismus směrovače Huawei neobsahují, zde je tedy nutná úprava MSS i pro čistou IPsec implementaci.

3.3.1 Úprava MTU na Cisco zařízeních

Konfigurace MTU v případě Cisco zařízení je tedy nutná v momentě, kdy s IPsec technologií dochází k implementaci GRE. Konfigurace probíhá na tunelovacím rozhraní, nebo na fyzickém rozhraní. Upravit lze rovněž MSS pomocí *ip tcp adjust-mss* příkazu.

```
RZ1(config)#interface Tunnel1
RZ1(config-if)#ip mtu 1440
```

3.3.2 Úprava MTU na Huawei zařízeních

V případě zařízení výrobce Huawei je nutné tuto úpravu provést i v případě implementace čistého IPsec VPN tunelu. MSS lze rovněž upravit pomocí *tcp adjust-mss* příkazu

```
[RZ2] interface GigabitEthernet 0/0/0
[RZ2-GigabitEthernet0/0/0]mtu 1445
```

4 Nástroje kvality služeb v IPsec VPN

S implementací QoS nástrojů v sítích IPsec VPN je třeba řešit několik problémů. Primární potíž nastává už při samotné klasifikaci paketu. Veškeré QoS nástroje jsou totiž aplikovány až po tunelovacím procesu ESP protokolu. Směrovač má tedy při klasifikaci k dispozici již zašifrovaný paket, který je navíc doplněn o nové IP záhlaví. Originální IP záhlaví směrovač již nemá k dispozici, a proto je mu znemožněna klasifikace na základě původních hodnot.

4.1 Klasifikace provozu v IPsec VPN sítích

Tuto komplikaci se postupem času výrobci rozhodli řešit nástrojem qos pre-classify. Tento nástroj již byl v textu této práce detailněji popsán v kapitole 2.11. Využití tohoto nástroje se může lišit v závislosti na konfiguraci. Administrátor je schopen navázat IPsec tunel nejenom pomocí crypto-map, jak je tomu v případě této práce, ale rovněž může využít virtuální tunelovací rozhraní. Tato rozhraní pracují na podobném principu jako nakonfigurovaný GRE tunel. Na nakonfigurované IPsec topologii byl navržen test, který demonstruje využití qos-preclassify nástroje. Tento test popisuje, jakou část paketu má směrovač RZ3 k dispozici při klasifikaci v závislosti na způsobu využití *qos pre-classify* příkazu.

Nejprve dojde ke konfiguraci access-listu pro jednotlivé protokolové části paketů.

```
RZ3(config)#access-list 121 permit esp any any
RZ3(config)#access-list 122 permit gre any any
RZ3(config)#access-list 123 permit icmp any any
```

Následně je potřeba nakonfigurovat příkazem klasifikační třídu, která odchytí veškerý provoz splňující podmínku uvedenou v access-listu.

```
RZ3(config)#class-map match-all test_esp
RZ3(config-cmap)#match access-group 121
RZ3(config)#class-map match-all test_gre
RZ3(config-cmap)#match access-group 122
RZ3(config)#class-map match-all test_icmp
RZ3(config-cmap)#match access-group 123
```

Dále je třeba vytvořit servisní politiku, která bude obsahovat všechny tři vytvořené třídy.

```
RZ3(config)#policy-map test_output
RZ3(config-pmap)#class test_esp
RZ3(config-pmap)#class test_gre
RZ3(config-pmap)#class test_icmp
```

Vytvořenou provozní třídu je třeba aplikovat na rozhraní v odchozím směru

```
RZ3(config)#interface GigabitEthernet0/0
RZ3(config-if)#service-policy output test_output
```

4.1.1 Bez použití QoS pre-classify

Bez použití nástroje qos pre-classify bude provoz ve směru RZ3-RZ1 i RZ3-RZ2 klasifikován stejným způsobem. V obou případech má směrovač při klasifikaci k dispozici paket zašifrovaný a zapouzdřený pomocí ESP protokolu. Kontrolu provedeme pomocí příkazu *show policy-map interface*.

```
 Service-policy output: test_output
   Class-map: test_esp (match-all)
     30 packets, 4680 bytes
     5 minute offered rate 1000 bps
     Match: access-group 121
   Class-map: test_gre (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps
     Match: access-group 122
   Class-map: test_icmp (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps
     Match: access-group 123
```

4.1.2 **QoS pre-classify a crypto-map**

Nyní se příkaz *qos pre-classify* aplikuje na obě vytvořené crypto-mapy.

```
RZ3(config)#crypto map ipsec_map 10 ipsec-isakmp
RZ3(config-crypto-map)#qos pre-classify
RZ3(config)#crypto map ipsec_map 20 ipsec-isakmp
RZ3(config-crypto-map)#qos pre-classify
```

Tento test bude mít dva výsledné efekty. Jedná-li se o obyčejný IPsec tunel, tedy směr RZ3-RZ2, bude mít směrovač k dispozici původní originální IP záhlaví. Dojde k úspěšné klasifikaci ICMP provozu.

```
Service-policy output: test_output
   Class-map: test_esp (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps
     Match: access-group 121
   Class-map: test_gre (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps
     Match: access-group 122
   Class-map: test_icmp (match-all)
     30 packets, 2640 bytes
     5 minute offered rate 0 bps
     Match: access-group 123
```

Nicméně v případě, kdy provoz prochází nejprve GRE a následně až IPsec tunelem nemá použití qos pre-classify na crypto-mapě žádný význam. V momentě, kdy nástroj qos pre-classify klonuje IP záhlaví je paket již zapouzdřen GRE protokolem.

```
Service-policy output: test_output
   Class-map: test_esp (match-all)
     0 packets, 0 bytes
     5 minute offered rate 0 bps
     Match: access-group 121
   Class-map: test_gre (match-all)
```

```
    30 packets, 3360 bytes
    5 minute offered rate 0 bps
    Match: access-group 122
  Class-map: test_icmp (match-all)
    0 packets, 0 bytes
    5 minute offered rate 0 bps
    Match: access-group 123
```

4.1.3 QoS pre-classify a tunelovací rozhraní

V případě, kdy je IPsec tunel navázán pomocí virtuálního tunelovacího rozhraní nebo je využito tunelovací rozhraní pro GRE protokol, je nutné příkaz *qos-preclassify* použít na tomto tunelovacím rozhraní.

```
RZ3(config)#interface Tunnel1
RZ3(config-if)#qos pre-classify
```

Provoz zapouzdřený pomocí GRE a následně pomocí ESP protokolu je nyní klasifikován na základě originálních hodnot v původním IP záhlaví.

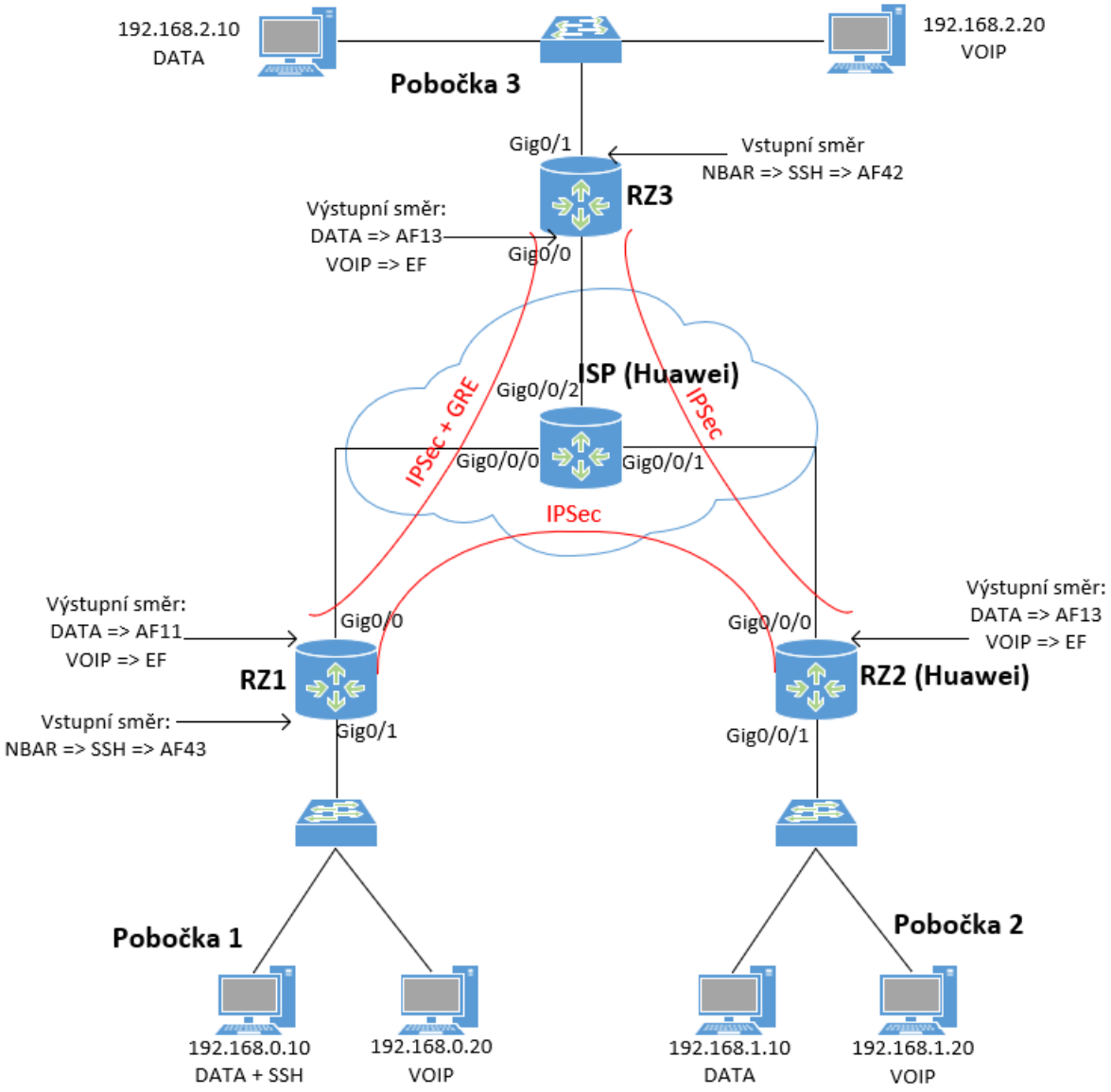
```
 Service-policy output: test_output
  Class-map: test_esp (match-all)
    0 packets, 0 bytes
    5 minute offered rate 0 bps
    Match: access-group 121
  Class-map: test_gre (match-all)
    0 packets, 0 bytes
    5 minute offered rate 0 bps
    Match: access-group 122
  Class-map: test_icmp (match-all)
    30 packets, 3360 bytes
    5 minute offered rate 0 bps
    Match: access-group 123
```

4.2 Klasifikace a značení provozu v IPsec VPN

V případě implementace QoS nástrojů v IPsec VPN sítích bývá často potřebné značení provozu pomocí pole DSCP v IP záhlaví paketu. Značení je potřeba z toho důvodu, že směrovače, které provádějí tunelovací a šifrovací procesy, jsou na cestě paketu poslední zařízení, která mohou na základě originálních hodnot v IP záhlaví provádět klasifikaci. Na základě této klasifikace následně provádí značkování pomocí DSCP hodnot. Směrovače, které jsou součásti tunelované sítě následně mohou provoz rozlišovat pouze na základě DSCP značek.

Na navržené IPsec VPN topologii bude v každé podsíti zákazníka přidáno jedno klientské zařízení. Definujeme pravidlo, že klientské zařízení s IP adresou 192.168.X.10 bude zařízení generující obyčejná data. Naopak zařízení s adresou 192.168.X.20 bude vždy zařízení generující prioritní data citlivá na zpoždění, například VoIP provoz. Obyčejná data klientských zařízení s adresou končící .10 budou označena DSCP značkou AF11, respektive AF13. Prioritní provoz klientského zařízení s adresou končící .20 bude vždy označen DSCP značkou EF. Dále bude na směrovačích RZ1 a RZ3 využit nástroj NBAR pro

hlubší inspekci paketů a následnou detekci SSH provozu. Scénář je navržen tak, že zařízení v pobočce 1 s adresou 192.168.0.10 patří správci infrastruktury sítě zákazníka. U tohoto zařízení jsou tedy častá SSH spojení se zařízeními v pobočce 3. Tento typ provozu bude odlišován od provozu obyčejných dat.



Obrázek 4.1: Schéma značkování v IPsec VPN topologii

4.2.1 Konfigurace klasifikace a značkování na RZ1 směrovači

Na úvod nakonfigurujeme nástroj NBAR. Konfigurace tohoto nástroje je velmi jednoduchá a jeho využití naopak velmi široké. Použití nástroje NBAR je vhodné v momentě, kdy je nutné klasifikovat provoz protokolu, který relaci navazuje pomocí dynamického portu. Klasifikace provozu takového protokolu by bylo složité řešit pomocí access-listu.

```
RZ1(config)#interface GigabitEthernet0/1
RZ1(config-if)#ip nbar protocol-discovery
```

Následuje konfigurace access-listu, který využijeme při detekci provozu jdoucího z adresy 192.168.0.10.

```
RZ1(config)# access-list 110 permit ip host 192.168.0.10 any
```

Nyní je nutné pomocí příkazu *class-map* vytvořit klasifikační třídu, která bude odchytávat SSH provoz, který je generován zařízením správce síťové infrastruktury.

```
RZ1(config)#class-map match-all match_SSH
RZ1(config-cmap)#match protocol ssh
RZ1(config-cmap)#match access-group 110
```

Takto odchycený provoz je třeba označkovat DSCP hodnotou AF43.

```
RZ1(config)#policy-map rz1_toLocal_input
RZ1(config-pmap)#class match_SSH
RZ1(config-pmap-c)#set dscp af43
```

Vytvořenou provozní politiku aplikujeme na rozhraní.

```
RZ1(config)#interface GigabitEthernet0/1
RZ1(config-if)#service-policy input rz1_toLocal_input
```

Podobná konfigurace probíhá na zařízení RZ3. Liší se pouze ve dvou aspektech. SSH provoz definovaný třídou na RZ3 směrovači nemusí splňovat podmínku uvedenou pomocí access-listu. Provoz se tedy nevztahuje na konkrétní IP adresu v podsítí. Dále se liší při značení DSCP hodnotou. Pro zkrácení výstupu této práce konfigurace nebude uvedena.

Nyní dojde ke konfiguraci klasifikace a značení provozu jednotlivých zařízení. Provoz hosta s IP adresou 192.168.0.10, jenž zároveň nesplňuje podmínku, že se jedná o SSH provoz, bude označen DSCP hodnotou AF11. Provoz, který je generován hostem s IP adresou 192.168.0.20 bude označen hodnotou EF.

Na úvod konfigurace je opět nutné nakonfigurovat třídy, které budou zajišťovat klasifikaci na základě nakonfigurovaných access-listů pro IP adresy jednotlivých zařízení.

```
RZ1(config)#access-list 110 permit ip host 192.168.0.10 any
RZ1(config)#access-list 111 permit ip host 192.168.0.20 any
```

```
RZ1(config)#class-map match-all match_DATA
RZ1(config-cmap)#match access-group 110
RZ1(config-cmap)#match not dscp af43
RZ1(config)#class-map match-all match_VOIP
RZ1(config-cmap)#match access-group 111
```

Provozní politika naopak zajistí potřebné značení pro klasifikovaný provoz.

```
RZ1(config)#policy-map marking
RZ1(config-pmap)#class match_DATA
RZ1(config-pmap-c)#set dscp af11
```

```
RZ1(config-pmap)#class match_VOIP
RZ1(config-pmap-c)#set dscp ef
```

Po aplikování provozní politiky na rozhraní bude zajištěna klasifikace a následné značení pro veškerý provoz generovaný pobočkou 1.

```
RZ1(config)#interface GigabitEthernet0/0
RZ1(config-if)#service-policy output marking
```

Konfigurace směrovače RZ3 i v tomto případě probíhá obdobným způsobem, viz [B].

4.2.2 **Konfigurace klasifikace a značkování na RZ2 směrovači**

V této podkapitole bude popsán způsob klasifikace a následného značkování na zařízení RZ2 výrobce Huawei. Koncept značkování bude založen na stejném principu jako u konfigurace zařízení RZ1.

Na úvod bude nutné nakonfigurovat access-list pro provoz s konkrétní zdrojovou IP adresou.

```
[RZ2]acl number 3021
[RZ2-acl-adv-3022]rule 5 permit ip source 192.168.1.10 0
[RZ2]acl number 3022
[RZ2-acl-adv-3022]rule 5 permit ip source 192.168.1.20 0
```

Následně bude vytvořena klasifikační třída, která bude klasifikovat provoz jdoucí z jednotlivých zařízení.

```
[RZ2]traffic classifier match_DATA
[RZ2-classifier-match_DATA]if-match acl 3021
[RZ2]traffic classifier match_VOIP
[RZ2-classifier-match_VOIP]if-match acl 3022
```

Nyní je nutné nakonfigurovat, jakým způsobem se s takto klasifikovaným provozem bude zacházet.

```
[RZ2]traffic behavior beh_DATA
[RZ2-behavior-beh_ DATA]remark dscp af13
[RZ2]traffic behavior beh_VOIP
[RZ2-behavior-beh_ VOIP]remark dscp ef
```

V poslední řadě dojde ke konfiguraci provozní politiky, kterou následně aplikujeme na rozhraní.

```
[RZ2]traffic policy marking
[RZ2-trafficpolicy-rz2_toNet_output]classifier match_ DATA behavior beh_ DATA
[RZ2-trafficpolicy-rz2_toNet_output]classifier match_ VOIP behavior beh_ VOIP
```

```
[RZ2]interface GigabitEthernet0/0/0
[RZ2- GigabitEthernet0/0/0]traffic-policy marking outbound
```

4.2.3 **Ověření značkování provozu**

Pro názornou ukázku značkování je proveden ping ze zařízení s adresou 192.168.0.10 na adresu 192.168.2.10. Testovací segment topologie vypadá stejně jako je uvedeno v předchozím testovacím případě. Ve směru RZ1-RZ3 dojde k označení paketu DSCP hodnotou AF11. V opačném směru RZ3-RZ1 bude paket označen DSCP hodnotou AF13.

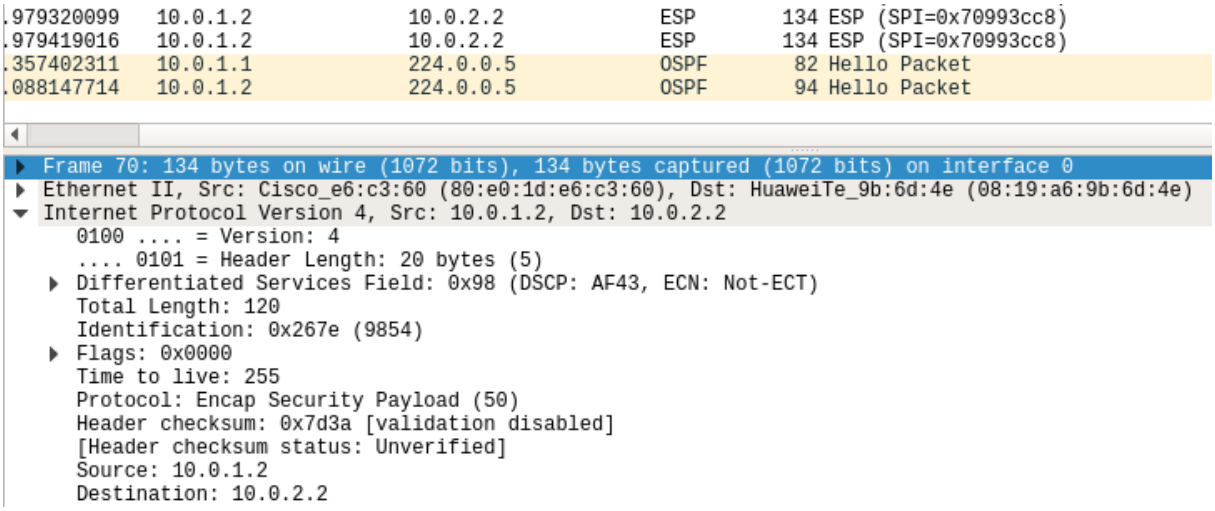
```
      7 2.388537    10.0.1.2                  10.0.2.2                  ESP      166 ESP (SPI=0x7c5492e7)
      8 2.390277    10.0.2.2                  10.0.1.2                  ESP      166 ESP (SPI=0xb589f0fb)
      9 3.223618    fe80::f032:e461:1e70:bc85 ff02::fb                  MDNS     107 Standard query 0x0000 PTR _i
     10 3.390024    10.0.1.2                  10.0.2.2                  ESP      166 ESP (SPI=0x7c5492e7)

▸ Frame 2: 166 bytes on wire (1328 bits), 166 bytes captured (1328 bits)
▸ Ethernet II, Src: Cisco_4b:50:28 (00:17:5a:4b:50:28), Dst: HuaweiTe_9b:6d:4e (08:19:a6:9b:6d:4e)
▾ Internet Protocol Version 4, Src: 10.0.1.2, Dst: 10.0.2.2
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▸ Differentiated Services Field: 0x28 (DSCP: AF11, ECN: Not-ECT)
    Total Length: 152
    Identification: 0x001d (29)
  ▸ Flags: 0x0000
    Time to live: 255
    Protocol: Encap Security Payload (50)
    Header checksum: 0xa3eb [validation disabled]
```

Obrázek 4.2: Ověření značkování ve směru RZ1-RZ3

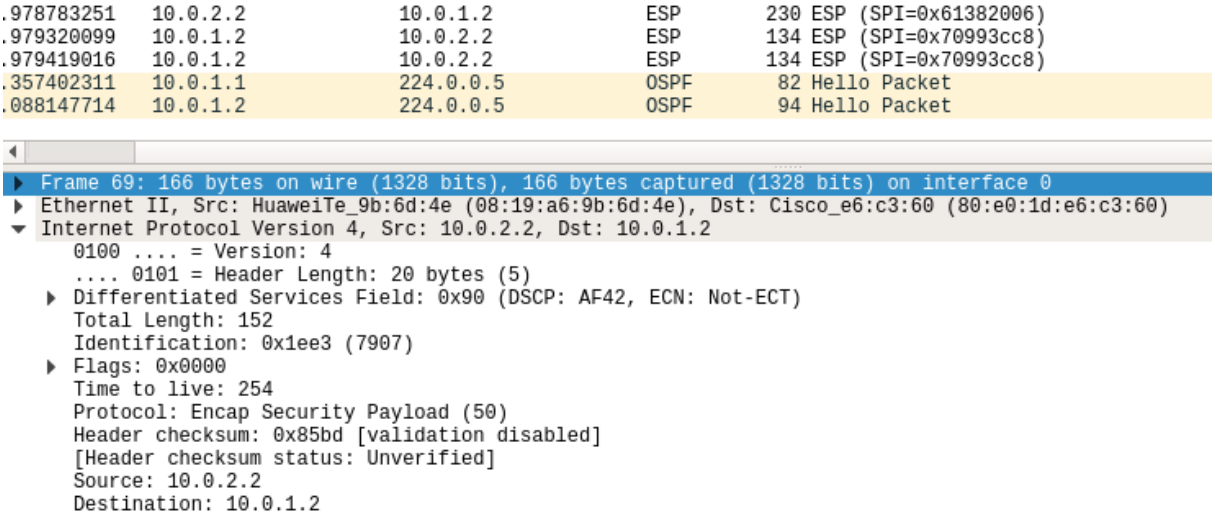
```
      6 1.388735    10.0.2.2                  10.0.1.2                  ESP      166 ESP (SPI=0xb589f0fb)
      7 2.388537    10.0.1.2                  10.0.2.2                  ESP      166 ESP (SPI=0x7c5492e7)
      8 2.390277    10.0.2.2                  10.0.1.2                  ESP      166 ESP (SPI=0xb589f0fb)
      9 3.223618    fe80::f032:e461:1e70:bc85 ff02::fb                  MDNS     107 Standard query 0x0000 PTR _i
     10 3.390024    10.0.1.2                  10.0.2.2                  ESP      166 ESP (SPI=0x7c5492e7)

▸ Frame 3: 166 bytes on wire (1328 bits), 166 bytes captured (1328 bits)
▸ Ethernet II, Src: HuaweiTe_9b:6d:4e (08:19:a6:9b:6d:4e), Dst: Cisco_4b:50:28 (00:17:5a:4b:50:28)
▾ Internet Protocol Version 4, Src: 10.0.2.2, Dst: 10.0.1.2
    0100 .... = Version: 4
    .... 0101 = Header Length: 20 bytes (5)
  ▸ Differentiated Services Field: 0x38 (DSCP: AF13, ECN: Not-ECT)
    Total Length: 152
    Identification: 0x001c (28)
  ▸ Flags: 0x0000
    Time to live: 254
    Protocol: Encap Security Payload (50)
    Header checksum: 0xa4dc [validation disabled]
```

Obrázek 4.3: Ověření značkování ve směru RZ3-RZ1

Rovněž dojde k otestování značkování SSH provozu. Bude navázáno SSH spojení ze zařízení správce v pobočce 1 s libovolným zařízením v pobočce 2. Provoz této SSH relace bude ve směru RZ1-RZ3 značkován DSCP značkou AF43. Zpětně pak bude provoz značen DSCP značkou AF42.



Obrázek 4.4: Ověření značkování SSH provozu RZ1-RZ3



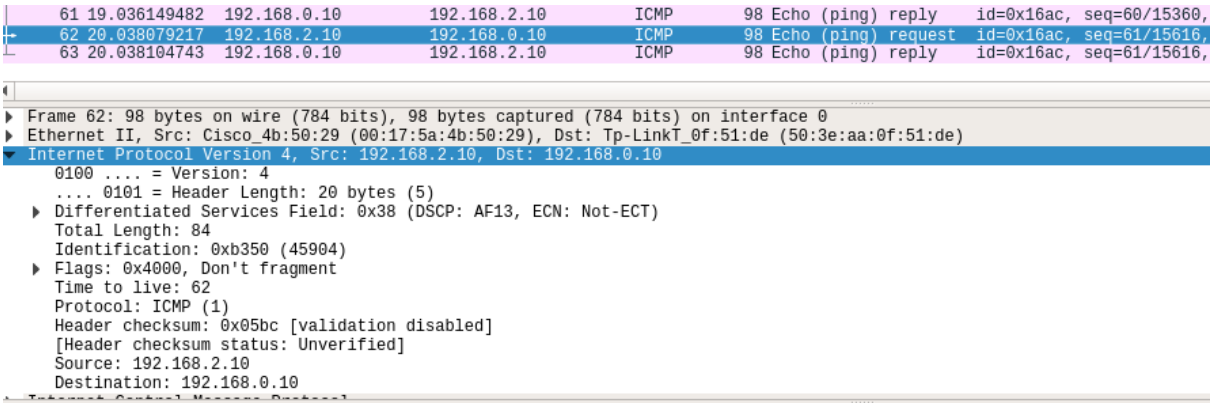
Obrázek 4.5: Ověření značkování SSH provozu RZ3-RZ1

4.2.4 Cisco IOS verze 12.4 a chybné značkování

Při testování se ukázalo, že značkování, které má na základě klasifikace naklonovaného originálního IP záhlaví přepisovat DSCP hodnotu vnějšího IP záhlaví nefunguje zcela správně. S vnějším záhlavím odejmuté DSCP značky byly nakonec viditelné i na klientských zařízeních. To znamená, že procesem značkování nebylo označeno pouze pole DSCP ve vnějším IP záhlaví, ale rovněž také pole DSCP v IP záhlaví vnitřním. Toto se ukázalo, že je chybou Cisco směrovačů s verzi IOS 12.4. Tato skutečnost realizaci této diplomové práce nijak neovlivňuje, v praxi ale bývá běžné, že značkování probíhá dříve než na směrovačích, které navazují IPsec tunel. Mohla by tedy nastat situace, kdy by směrovač zpracovával paket, který ve vnitřním IP záhlaví již DSCP hodnotu má předem vyplněnou. Tato hodnota by při zapouzdření pomocí ESP protokolu v tunelovacím módu byla překopírovaná do nově vytvořeného IP záhlaví. Na základě klasifikace by pak tuto hodnotu ve vnějším IP záhlaví administrátor mohl nechat přepsat. Nicméně zde by správce zamýšlel přepsání pouze DSCP značky ve vnějším IP

záhlaví. Chyba procesu klasifikace a následného značkování ve verzi IOS 12.4. by zapříčinila chybné přepsání DSCP hodnoty i ve vnitřním IP záhlaví.

Na následujícím snímku lze zpozorovat vliv této chyby na provoz. Je generován ping ze zařízení s IP adresou 192.168.2.10 na zařízení s IP adresou 192.168.0.10, tedy z pobočky 3 na pobočku 1. Na RZ3 směrovači je paket klasifikován na základě hodnot z naklonovaného originálního IP záhlaví. Na základě této klasifikace je mu přidělena DSCP značka AF13. Tato značka by při správném chování RZ3 směrovače měla být viditelná pouze uvnitř tunelované sítě. Nicméně níže uvedený snímek dokazuje, že provoz odchycený programem Wireshark na klientském zařízení v pobočce 1 má označkované i vnitřní IP záhlaví.



Obrázek 4.6: Chybné značkování vnitřního IP záhlaví

4.2.4.1 **Řešení problémového značkování**

Řešením tohoto problému může být upgrade verze operačního systému IOS Cisco směrovačů. Uvedená topologie byla rovněž otestována na Cisco zařízeních IOS verze 15.5. Zde se problém nevyskytuje a značkování funguje správným způsobem.

Problém lze řešit i navázáním IPsec tunelů pomocí virtuálních tunelovacích rozhraní (VTI). Konfigurace je podobná popsané konfiguraci v podkapitole 3.2. Použití příkazů *crypto isakmp policy* a *crypto ipsec transform-set* zůstává zachováno. Změna přichází v nahrazení crypto-map takzvaným IPsec profilem.

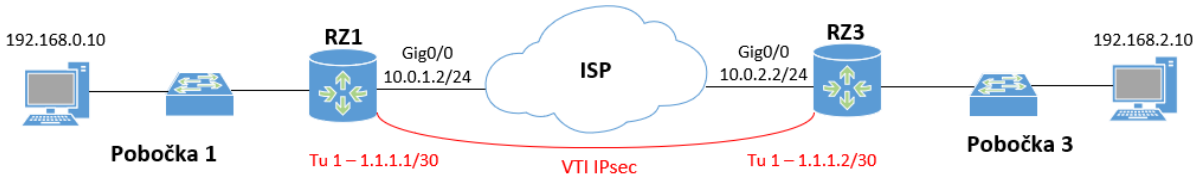
```
RZ1(config)#crypto ipsec profile profile
RZ1(ipsec-profile)#set transform-set myset
```

Následně je nutná konfigurace tunelovacího rozhraní. V této části konfigurace je rovněž nutné definovat, že se jedná o rozhraní, které má být zabezpečeno IPsec VPN. Rovněž je nutné uvést v předchozím kroku nakonfigurovaný IPsec profil.

```
RZ1(config)#interface Tunnel 0
RZ1(config-if)#ip address 1.1.1.1 255.255.255.252
RZ1(config-if)#tunnel source 10.0.1.2
RZ1(config-if)#tunnel destination 10.0.2.2
RZ1(config-if)#tunnel mode ipsec ipv4
RZ1(config-if)#tunnel protection ipsec profile profile
```

V posledním kroku je nutné uvést statickou cestu pro síť na opačné straně tunelu.

```
RZ1(config)#ip route 192.168.2.0 255.255.255.0 Tunnel0
```

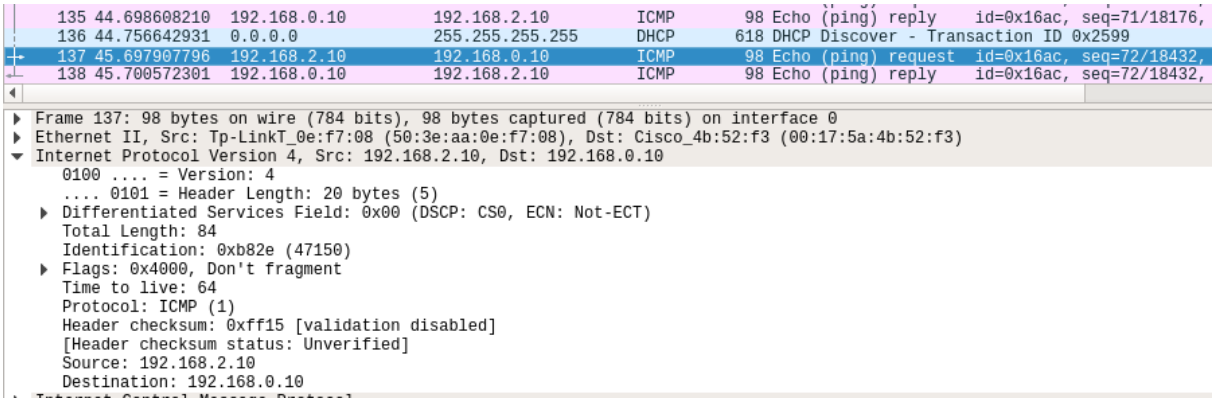


Obrázek 4.7: Topologie s využitím VTI

Toto navržené řešení nyní umožňuje aplikaci provozní politiky, která zajišťuje značkování dvěma způsoby. Provozní politiku lze nakonfigurovat na tunelovací rozhraní. V tomto případě je paket klasifikován na základě údajů v originálním IP záhlaví.

```
RZ1(config)#interface Tunnel 0
RZ1(config-if)#service-policy output marking
```

Rovněž je možné provozní politiku zajišťující značkování aplikovat na fyzické rozhraní. V takovémto případě klasifikace probíhá na základě údajů ve vnějším IP záhlaví přidaném po procesu ESP zapouzdření v tunelovacím módu. V případě využití qos pre-classify nástroje na tunelovacím rozhraní lze opět klasifikovat na základě originálního IP záhlaví a následně označkovat vnější IP záhlaví DSCP hodnotou. Správnost lze opět ověřit generováním ping paketů z pobočky 3 do pobočky 1. Na RZ3 opět dorazí paket, který je klasifikován na základě naklonovaného IP záhlaví. Na základě této klasifikace je paketu označkováno vnější IP záhlaví DSCP značkou. Takto označený paket následně putuje tunelovanou sítí poskytovatele služeb. Na RZ1 směrovači je paketu odstraněno vnější IP záhlaví a do klientské sítě pobočky 1 je odeslán paket pouze s původním IP záhlavím. Toto záhlaví, díky metodě navázání IPsec tunelu pomoci virtuálních tunelovacích rozhraní má správně výchozí CS0 značku.



Obrázek 4.8: Opravené značkování po využití VTI

5 Implementace nástrojů kvality služeb v IPsec VPN

Při implementaci jednotlivých nástrojů ovlivňující přenosové parametry nastal další problém. Jednalo se o problém týkající se licence SECURITY/K9, která je aktivní na směrovačích, které jsou ve školní laboratoři k dispozici. Tento problém je možné řešit právě pomocí nástrojů kvality služeb.

5.1 SECURITY/K9 Licence

Všechna Cisco zařízení exportována do zahraničí jsou exportována buď s licencí SEC-K9, nebo HSEC-K9. První zmíněná aplikuje na vytvářená VPN spojení přísná omezení. S touto licencí je maximální možný počet navázaných VPN tunelů 225. Zároveň pak má každý navázaný VPN tunel omezenou propustnost na 85 Mbit/s. Nicméně, při orientačním měření nástrojem Iperf3 na zařízeních Cisco 2811 s IOS verzí 12.4 se propustnost pohybovala okolo 20 Mbit/s. V momentě, kdy generovaná propustnost překročila limit, došlo k drastickému omezení propustnosti daného zařízení. Podobné chování bylo zaznamenáno na zařízeních Cisco 2901 s IOS verzí 15.5. Zde se naměřená propustnost pohybovala okolo hodnoty 40 Mbit/s. Tato hodnota je stále hluboko pod limitem, který je dán licencí. Situaci, kdy dochází k překročení limitu a následnému výraznému omezení propustnosti, lze předcházet právě využitím nástrojů kvality služeb.

```
Router#show platform cerm-information
Crypto Export Restrictions Manager(CERM) Information:
 CERM functionality: ENABLED

 ---------------------------------------------------------------
 Resource                    Maximum Limit          Available
 ---------------------------------------------------------------
 Tx Bandwidth(in kbps)       85000                  85000
 Rx Bandwidth(in kbps)       85000                  85000
 Number of tunnels           225                    224
 Number of TLS sessions      1000                   1000

 Resource reservation information:
 D - Dynamic
 ---------------------------------------------------------------------------
 Client         Tx Bandwidth    Rx Bandwidth    Tunnels    TLS Sessions
                 (in kbps)       (in kbps)
 ---------------------------------------------------------------------------
 VOICE          0               0               0          0
 IPSEC          D               D               1          N/A
 SSLVPN         D               D               0          N/A
```

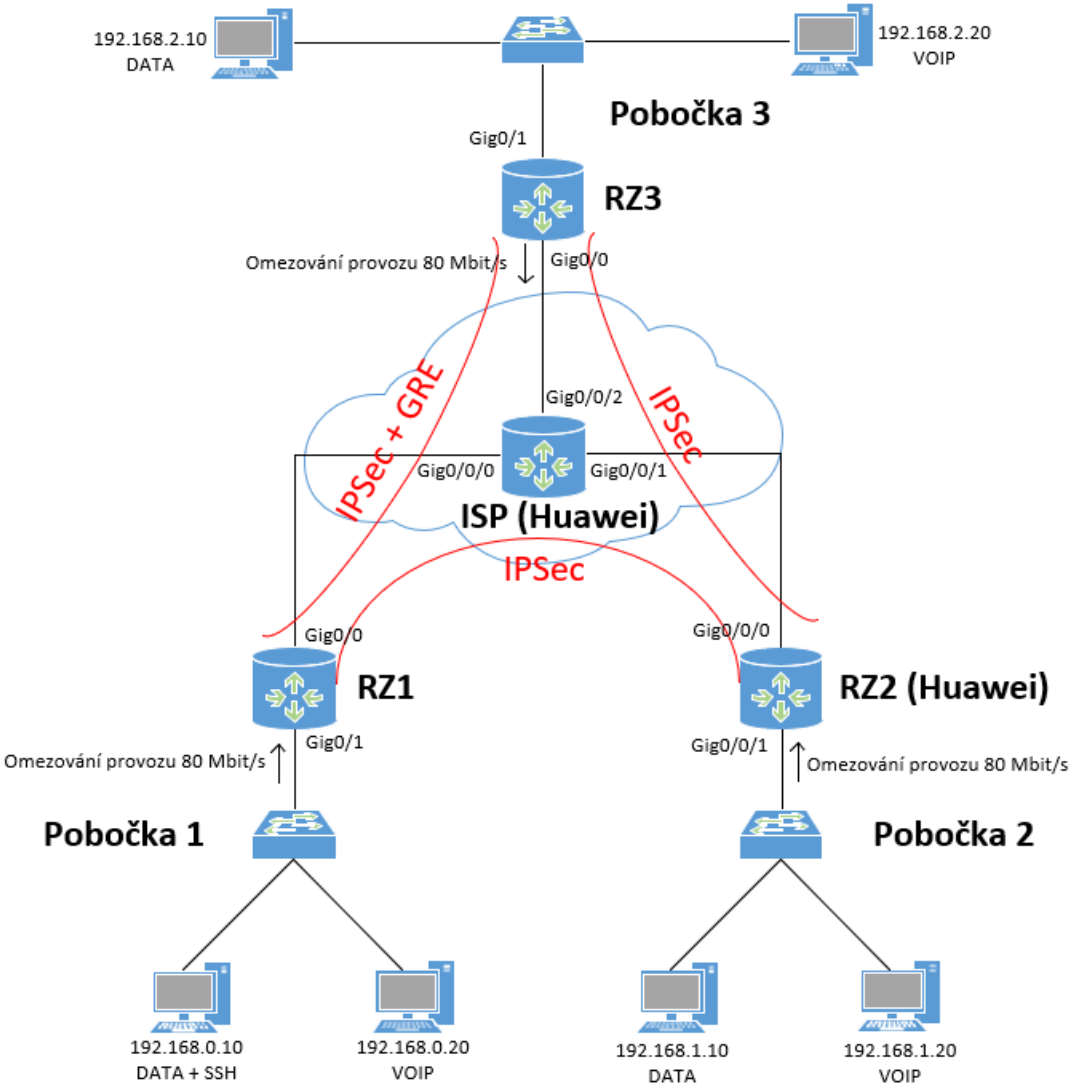
Obrázek 5.1: Licence Security-K9

```
student@pc14:~$ iperf3 -c 192.168.0.10
Connecting to host 192.168.0.10, port 5201
[  4] local 192.168.2.10 port 56464 connected to 192.168.0.10 port 5201
[ ID] Interval           Transfer     Bandwidth       Retr  Cwnd
[  4]   0.00-1.00   sec  6.10 MBytes  51.2 Mbits/sec   54   33.8 KBytes
[  4]   1.00-2.00   sec  4.46 MBytes  37.4 Mbits/sec   53   36.5 KBytes
[  4]   2.00-3.00   sec  4.89 MBytes  41.0 Mbits/sec   83   35.2 KBytes
[  4]   3.00-4.00   sec  4.28 MBytes  35.9 Mbits/sec   52   35.2 KBytes
[  4]   4.00-5.00   sec  5.50 MBytes  46.2 Mbits/sec   80   33.8 KBytes
[  4]   5.00-6.00   sec  5.51 MBytes  46.2 Mbits/sec   79   36.5 KBytes
[  4]   6.00-7.00   sec  3.82 MBytes  32.0 Mbits/sec   80   29.8 KBytes
[  4]   7.00-8.00   sec  5.36 MBytes  44.9 Mbits/sec   55   35.2 KBytes
^C[  4]   8.00-8.97   sec  3.86 MBytes  33.5 Mbits/sec   79   33.8 KBytes
- - - - - - - - - - - - - - - - - - - - - - - - -
[ ID] Interval           Transfer     Bandwidth       Retr
[  4]   0.00-8.97   sec  43.8 MBytes  41.0 Mbits/sec  615             sender
[  4]   0.00-8.97   sec  0.00 Bytes  0.00 bits/sec                  receiver
```

Obrázek 5.2: Naměřená propustnost při překročení limitu

5.2 Omezování provozu

Tato kapitola bude problém vzniklý licencí řešit pomocí využití metody omezování provozu. Tato metoda umožňuje administrátorovi nastavení maximální hodnoty propustnosti, po jejímž překročení dojde k zahazování paketů. Tímto způsobem lze předcházet aktivaci licenčního omezení.



Obrázek 5.3: Schéma implementace omezování provozu

Dle uvedeného schématu bude provedeno omezování provozu na zařízeních RZ1, RZ2 a RZ3. Zde v textu bude popsána implementace omezování provozu pouze na směrovačích RZ2 a RZ3. Na těchto směrovačích se implementace bude lišit ve směru, ve kterém bude omezování aktivováno.

5.2.1 **Algoritmus kupónový kyblík**

Zařízení obou výrobců využívají algoritmus kupónový kyblík. Do kupónového kyblíku se přidávají kupóny průměrnou přenosovou propustností (CIR-Commited Information Rate). Kyblík má omezenou velikost nazvanou Bc (Burst size). V případě příchodu paketu jsou z kyblíku odebrány kupóny (1 kupón = 1 bajt paketu) a následně se paket odešle. V případě nedostatku kupónů je paket zahozen.

Zařízení Huawei využívají nástroj CAR, který pracuje se dvěma kupónovými kyblíky. Kyblík C je doplňován kupóny s přenosovou rychlostí CIR a má maximální velikost CBS (Commited Burst Size). Kyblík P je doplňován kupóny maximální povolenou přenosovou rychlostí PIR (Peak Information Rate) a má maximální velikost EBS (Excess Burst Size). V případě, že na směrovač dorazí provoz s přenosovou rychlostí B a kyblík C i P mají dostatek kupónů, provoz je označen zeleně. Pokud má kyblík P dostatek kupónu pro přenos paketů, ale kyblík C nikoliv, pakety jsou označeny žlutě. Ve zbylých případech jsou pakety označeny červeně [11].

5.2.2 **Konfigurace RZ2 směrovače**

Na RZ2 směrovači bude omezování aktivováno na rozhraní GigabitEthernet0/0/1 a to ve vstupním směru. K omezování a potenciálnímu zahazování paketů dojde tedy v momentě příchodu paketu z pobočky 2 na směrovač RZ2. Omezován bude veškerý provoz jdoucí v tomto směru na zařízení RZ2.

Na úvod bude nakonfigurována klasifikační třída, do které bude spadat veškerý provoz.

```
[RZ2]traffic classifier match_any
[RZ2-classifier-match_any]if-match any
```

Následně dojde ke konfiguraci třídy, která zajistí samotné omezování provozu na 80 Mbit/s.

```
[RZ2]traffic behavior police
[RZ2-behavior-police]car cir 80000 green pass yellow discard red discard
[RZ2-behavior-police]statistic enable
```

Nyní je nutné nakonfigurovat provozní politiku, která spojí dohromady nakonfigurovanou klasifikační třídu a třídu zajišťující samotné omezování.

```
[RZ2]traffic policy police
[RZ2-policy-police]classifier match_any behavior police
```

Tuto provozní politiku následně tak, jak je uvedeno na schématu aplikujeme na rozhraní GigabitEthernet0/0/1 ve vstupním směru.

```
[RZ2]interface GigabitEthernet0/0/1
[RZ2- GigabitEthernet0/0/1]traffic-policy police inbound
```

5.2.3 **Konfigurace RZ3 směrovače**

Na směrovači RZ3 bude omezování provozu nastaveno na rozhraní GigabitEthernet0/0 v odchozím směru. Při této konfiguraci je nutné poukázat na neefektivnost tohoto nastavení. Problém spočívá v tom, že k omezování provozu, a tedy k zahazování paketů překračující nastavenou mez dochází až po

procesu šifrování. Toto nastavení, byť je z hlediska konfigurace možné je v porovnání s konfiguraci RZ2 směrovače ryze neefektivní.

Při konfiguraci RZ3 směrovače není nutné konfigurovat klasifikační třídu jako tomu bylo v předchozí konfiguraci. Při nakonfigurování servisní politiky se v této politice vyskytuje výchozí třída. Do této výchozí třídy spadá automaticky veškerý provoz jdoucí přes rozhraní, kde je tato servisní politika aplikována. Na výchozí třídu se aplikuje omezování provozu na 80 Mbit/s. Součástí této výchozí třídy bude podtřída zajišťující značkování.

```
RZ3(config)#policy-map police
RZ3(config-pmap)#class class-default
RZ3(config-pmap-c)#police 80000000 conform-action transmit exceed-action drop
RZ3(config-pmap-c)#service-policy marking
```

Následně tuto třídu aplikujeme na rozhraní. Jak již bylo zmíněno, konfigurace proběhne na rozhraní v odchozím směru k poskytovateli. Zahazované budou pakety, které již předtím prošly procesem šifrování. Výpočetní výkon směrovače bude nadužíván z důvodu neefektivní konfigurace.

```
RZ3(config)#interface GigabitEthernet0/0
RZ3(config-if)#service-policy output police
```

5.2.4 **Ověření omezování provozu**

Na směrovači RZ2 došlo při implementaci třídy zajišťující samotné omezování rovněž k aktivování statistik příkazem *statistic enable*. Díky tomuto příkazu můžeme nyní ověřit funkčnost omezování provozu rovnou na RZ2 směrovači. Z následného výpisu lze vyčíst počet paketů, který byl zařazen pod vytvořenou klasifikační třídu. Na tento počet paketů byla následně aplikována metoda omezování provozu. Z celkového počtu 83727 paketů bylo odesláno 82282 paketů. Zbylých 1445 paketů bylo metodou omezování provozu zahozeno.

```
Interface: GigabitEthernet0/0/1
 Traffic policy inbound: police
 Rule number: 1
 Current status: OK!
Item                  Sum(Packets/Bytes)        Rate(pps/bps)
-------------------------------------------------------------------------------
Matched                    83,727/                  0/
                           122,629,066              0
 +--Passed                 82,282/                  0/
                           120,507,806              0
 +--Dropped                1,445/                   0/
                           2,121,260                0
  +--Filter                0/                       0/
                           0                        0
  +--CAR                   1,445/                   0/
                           2,121,260                0
 +--Car                    83,727/                  0/
                           122,629,066              0
  +--Green packets         82,282/                  0/
                           120,507,806              0
```

+--Yellow packets	1,152/ 1,691,136	0/ 0
+--Red packets	293/ 430,124	0/ 0

Pro přesnější naměření přenosových parametrů byl připojen generátor provozu ParaScope GigE. Na tomto přístroji byla použita aplikace Traffic Test. Pro testovací účely byl generován 100 Mbit/s provoz ze zařízení s IP adresou 192.168.0.10 na zařízení s IP adresou 192.168.1.10. Ve výsledném výpisu tohoto měření lze zpozorovat účinky aplikované metody omezování provozu. Rovněž lze pozorovat ztrátovost paketů, která při aplikaci této metody vznikla zahazováním paketů.

Tabulka 5.1: Naměřené hodnoty přenosových parametrů (Omezování provozu)

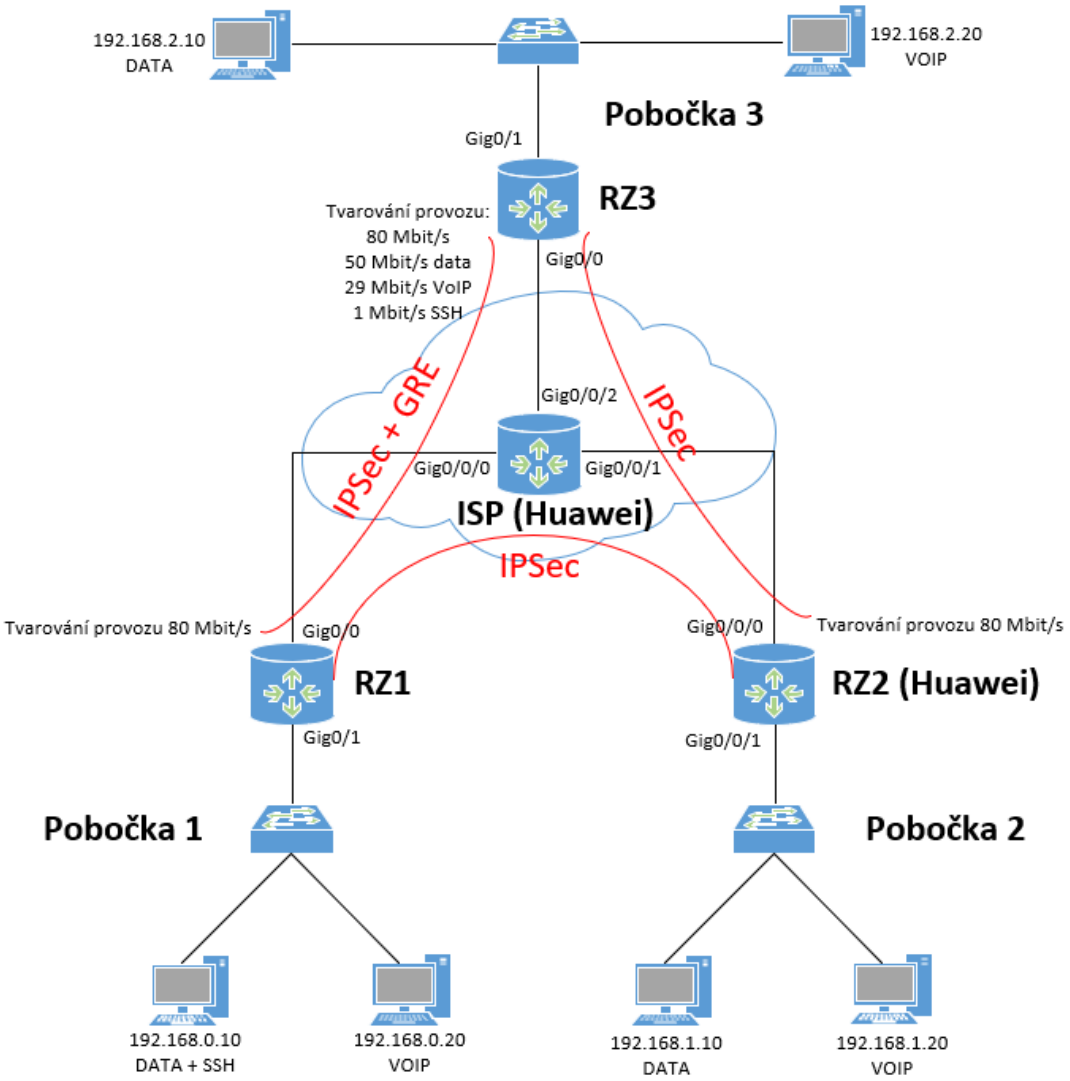
	Odesláno	Přijato
Propustnost (Mbps)	100.00	76.75
Počet paketů	925365	690642
Počet bajtů	925364156	690642000

Při generování 100 Mbit/s bylo zpětně přijímáno průměrně 76 Mbit/s. Z celkového počtu 925365 odeslaných paketů bylo zpětně přijato 690642. Pakety byly generovány s velikostí 1000b. Konečná ztrátovost se dá vyčíslit na 25,36 %. Tato hodnota zhruba odpovídá očekávánému výsledku. Zároveň lze říct, že problém vzniklý licenčním limitem je vyřešen.

5.3 Tvarování provozu

Implementace omezování provozu sice řešila problém vzniklý licencí Security-K9, nicméně za cenu toho, že nadbývající provoz byl rovnou zahozen. Toto řešení z hlediska uživatele není úplně ideální. Proto efektivnější nástroj na zajištění toho, aby hodnota propustnosti nepřekročila limit 85 Mbit/s, který je dán licencí SEC-K9, je tvarování provozu. Tato metoda provoz převyšující limit na krátkou dobu pozdrží ve frontě, aby jej v momentě uvolnění linky mohla poslat.

Následující podkapitoly popíšou postup implementace nástroje tvarování provozu na všech třech směrovačích zákazníka.



Obrázek 5.4: Implementace tvarování provozu

5.3.1 Konfigurace tvarování provozu RZ1 směrovače

Na RZ1 směrovači bude opět využita výchozí třída class-default, do které spadá bez výjimky veškerý provoz jdoucí z pobočky 1. Jako podtřída bude využita třída *marking*, která zajistí značkování provozu DSCP značkami.

```
RZ1(config)#policy-map shapingv2
RZ1(config-pmap)#class class-default
RZ1(config-pmap-c)#shape average 80000000
RZ1(config-pmap-c)#service-policy marking
```

Tuto provozní politiku je třeba aplikovat na rozhraní GigabitEthernet0/0 v odchozím směru. Rovněž je třeba upozornit, že metodu tvarování provozu na rozhraní ve vstupním směru nelze aplikovat.

```
RZ1(config)#interface GigabitEthernet0/1
RZ1(config-if)#service-policy output shapingv2
```

5.3.2 **Konfigurace tvarování provozu RZ2 směrovače**

V případě směrovače Huawei bude konfigurace tvarování provozu velmi jednoduchá. Tento výrobce má nástroj tvarování provozu nastavitelný i na samotných fyzických rozhraních. Využitím tohoto nástroje dojde ke tvarování veškerého provozu procházejícího tímto rozhraním.

```
[RZ2]interface GigabitEthernet0/0/0
[RZ2- GigabitEthernet0/0/0]qos gts cir 80000
```

5.3.3 **Konfigurace tvarování provozu RZ3 směrovače**

Směrovač RZ3 bude mít tvarování provozu řešen nejkomplexněji. Řešení bude složeno z jedné rodičovské třídy zajišťující tvarování veškerého provozu na 80 Mbit/s. Dále se bude skládat ze tří podtříd, které budou na základě klasifikace provozu provoz dále tvarovat na konkrétní hodnoty. Provoz splňující kritéria obyčejných dat bude tvarován na hodnotu 50 Mbit/s. Provoz spadající do klasifikační třídy pro VoIP bude tvarován na hodnotu 29 Mbit/s. Případný SSH provoz, klasifikován nástrojem NBAR, bude tvarován na 1 Mbit/s. Pro klasifikaci provozu budou opět využity již vytvořené klasifikační třídy.

Na úvod vytvoříme provozní politiku, která bude zajišťovat klasifikaci provozu a jeho následné rozřazení do jednotlivých podtříd. Těmto třídám bude následně přiřazena hodnota, na kterou bude daný typ provozu tvarován. Jednotlivé typy provozu na základě klasifikace označkujeme příslušnou DSCP hodnotou podle konceptu z podkapitoly 4.2.

```
RZ3(config)#policy-map shaping_Child
RZ3(config-pmap)#class match_DATA
RZ3(config-pmap-c)#shape average 50000000
RZ3(config-pmap-c)#set dscp af13
RZ3(config-pmap)#class match_VOIP
RZ3(config-pmap-c)#shape average 29000000
RZ3(config-pmap-c)#set dscp ef
RZ3(config-pmap)#class match_SSH_response_DSCP
RZ3(config-pmap-c)#shape average 1000000
```

Nyní je potřeba vytvořit hlavní, rodičovskou provozní politiku, která bude veškerý provoz tvarovat na hodnotu 80 Mbit/s. Tato třída bude obsahovat v předchozím kroku vytvořenou podtřídu.

```
RZ3(config)#policy-map shapingv2
RZ3(config-pmap)#class class-default
RZ3(config-pmap-c)#shape average 80000000
RZ3(config-pmap-c)#service-policy shaping_Child
```

V posledním kroku je potřeba tuto provozní politiku aplikovat na rozhraní GigabitEthernet0/0 v odchozím směru, tedy ve směru k poskytovateli.

```
RZ3(config)#interface GigabitEthernet0/0
RZ3(config-if)#service-policy output shapingv2
```

5.3.4 **Ověření tvarování provozu**

Zda je metoda tvarování provozu aktivována lze ověřit tímto příkazem. Ověření je provedeno na směrovači RZ1. Počet paketů, který byl zpracován provozní politikou shapingv2. Téměř všechny pakety takto zpracované byly následně klasifikovány jako obyčejná data a v podtřídě bylo zajištěno jejich značkování DSCP hodnotou af11.

```
RZ1#show policy-map interface
 GigabitEthernet0/0
  Service-policy output: shapingv2
    Class-map: class-default (match-any)
      1089382 packets, 1091033547 bytes
      5 minute offered rate 0000 bps, drop rate 0000 bps
      Match: any
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 1089382/1155796491
      shape (peak) cir 80000000, bc 320000, be 320000
      target shape rate 160000000
      Service-policy : marking
        Class-map: match_DATA (match-all)
          1088208 packets, 1090913952 bytes
          5 minute offered rate 0000 bps, drop rate 0000 bps
          Match: access-group 110
          Match: not  dscp af43 (38)
          QoS Set
            dscp af11
              Packets marked 1088208
```

Pro ověření chování konfigurace lze na směrovačích využít následující příkaz. Ve výpisu lze vidět množství paketů, které bylo na RZ3 směrovači nejprve zachyceno rodičovskou třídou zajišťující tvarování na 80 Mbit/s. Následně, jelikož šlo o provoz obyčejných dat byl provoz zachycen klasifikační třídou *match_DATA* a následně tvarován na hodnotu 50 Mbit/s.

```
RZ3#show policy-map interface
 GigabitEthernet0/0
  Service-policy output: shapingv2
    Class-map: class-default (match-any)
      26930 packets, 1801863 bytes
      5 minute offered rate 36000 bps, drop rate 0000 bps
      Match: any
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/0/0
      (pkts output/bytes output) 26930/3639007
      shape (average) cir 80000000, bc 320000, be 320000
      target shape rate 80000000
```

```
Service-policy : shaping_Child
  Class-map: match_DATA (match-all)
    26911 packets, 1800114 bytes
    5 minute offered rate 36000 bps, drop rate 0000 bps
    Match: access-group 111
    Match: not  dscp af42 (36)
    Queueing
    queue limit 64 packets
    (queue depth/total drops/no-buffer drops) 0/0/0
    (pkts output/bytes output) 26911/3637258
    shape (average) cir 50000000, bc 200000, be 200000
    target shape rate 50000000
    QoS Set
      dscp af13
        Packets marked 26911
```

Následně byly přenosové parametry v takto nastavené topologii otestovány generátorem provozu ParaScope GigE. Nejprve byl generován 100 Mbit/s provoz z pobočky 1 do pobočky 2. Zde bylo v obou směrech aktivní tvarování provozu na hodnotu 80 Mbit/s. Při interpretaci tohoto výsledku je nutné brát v potaz to, jakým způsobem metoda tvarování provozu funguje. Metodu je vhodná využít pro zpracování provozu, který sítí putuje ve shlucích. V případě, kdy generátorem dlouhodobě generujeme provoz 100 Mbit/s, dojde k rychlému zahlcení fronty a následnému zahazování paketů stejně jako je tomu u metody omezování provozu. I zde ale lze pozorovat rozdíl ve výsledku oproti předchozí metodě.

Tabulka 5.2: Naměřené hodnoty přenosových parametrů (Tvarování provozu – RZ1)

	Odesláno	Přijato
Propustnost (Mbps)	100.00	78.48
Počet paketů	511557	404144
Počet bajtů	511556332	404144000

I při tomto měření byla naměřena ztrátovost paketů přibližně 21 %. Jak již bylo řečeno, při kontinuálním překračování se pakety po přeplnění front začnou zahazovat stejně jako v případě využití metody omezování provozu.

Dále byla provedena další dvě měření aplikovaného tvarování provozu na směrovači RZ3. Nejprve byl generován 100 Mbit/s provoz simulující provoz jdoucí ze zařízení s adresou 192.168.2.10. Tedy v konceptu této práce se jedná o provoz obyčejný dat. Na tomto směrovači je aplikováno tvarování provozu prostřednictvím hlavní, rodičovské třídy na 80 Mbit/s. Dále pak je aktivována podtřída, která tvaruje provoz obyčejných dat na hodnotu 50 Mbit/s.

Tabulka 5.3: Naměřené hodnoty přenosových parametrů (Tvarování obyčejných dat – RZ3)

	Odesláno	Přijato
Propustnost (Mbps)	100.00	48.03
Počet paketů	1110280	533165
Počet bajtů	1110279280	533165000

Jelikož došlo k překročení nastaveného limitu větší mírou než v případě měření na RZ1 směrovači, je naměřená ztrátovost paketů 51,98 %.

Následně bylo provedeno měření, které odpovídá provozu citlivému na přenosové parametry sítě. Jedná se o VoIP provoz, tedy o provoz generovaný zařízením s adresou 192.168.2.20. Na tento provoz je aktivovaná opět hlavní třída, která veškerý provoz tvaruje na hodnotu 80 Mbit/s. Dále dojde k aktivování podtřídy, která klasifikuje VoIP provoz a k následnému tvarování na hodnotu 29 Mbit/s.

Tabulka 5.4: Naměřené hodnoty přenosových parametrů (Tvarování VoIP provozu – RZ3)

	Odesláno	Přijato
Propustnost (Mbps)	100.00	27.82
Počet paketů	960366	267512
Počet bajtů	960365244	267512000

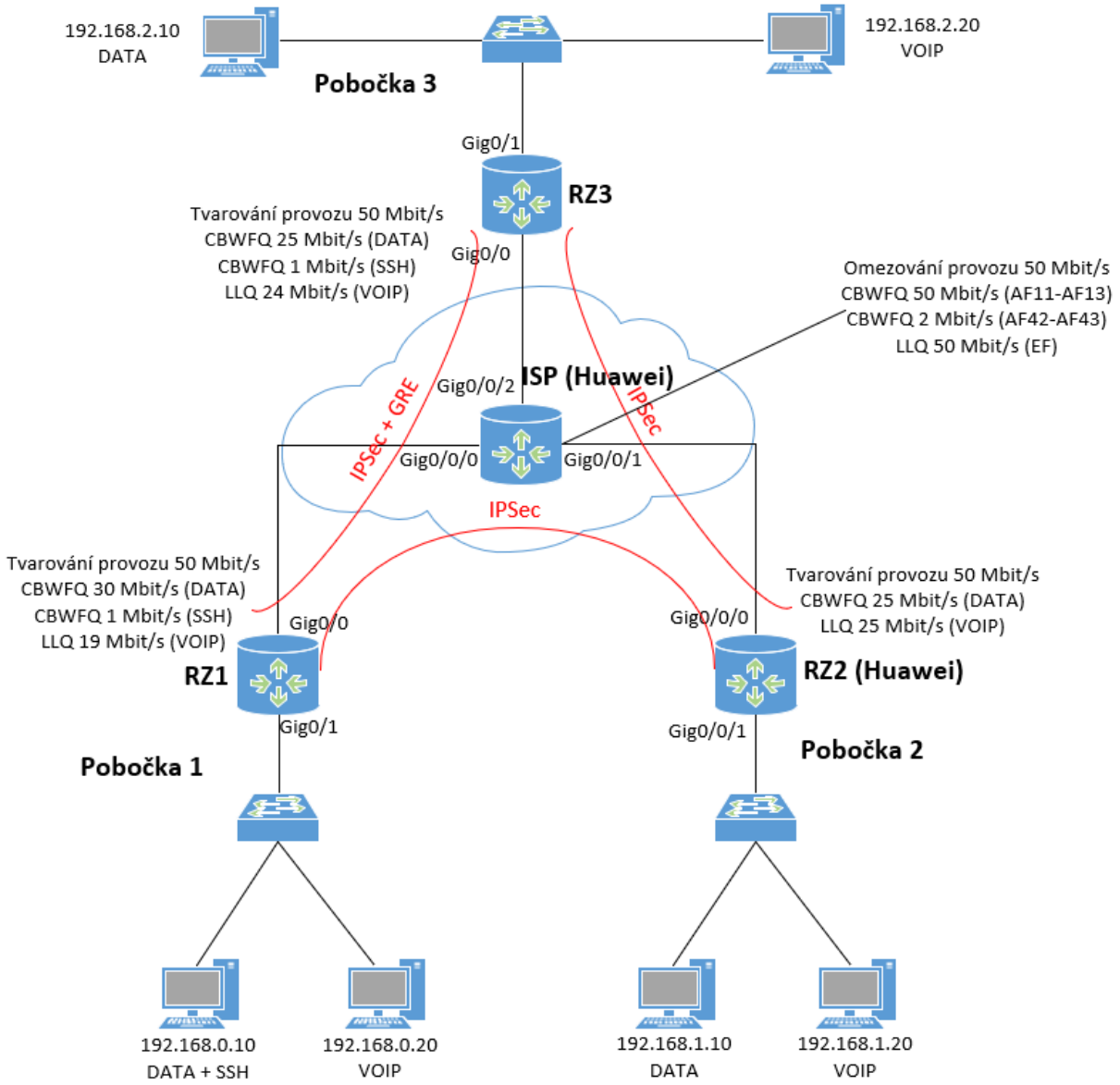
Naměřená propustnost zde činila průměrně 28 Mbit/s. Ztrátovost paketů byla 72,14 %.

5.4 Tvarování, omezování a prioritizace provozu

Tato podkapitola používá dvě předešlé metody tvarování a omezování provozu. Mimo to zde bude popsaná implementace dvou metod obsluhy paketových front CBWFQ a LLQ. Provoz bude omezován a zároveň tvarován na hodnotu 50 Mbit/s. Následně bude rozdělen na základě klasifikace do několika front. Dvě fronty budou typu CBWFQ. Bude do nich spadat provoz klasifikován jako obyčejná data, nebo SSH. Další fronta bude určena pro prioritní provoz citlivý na přenosové zpoždění, tedy VoIP provoz. Tato fronta bude typu LLQ. Bude tedy takto klasifikovanému provozu garantovat nejenom hodnotu propustnosti, ale rovněž okamžité odbavení. LLQ fronta zároveň zaručí, že VoIP provoz, který by překročil tuto garantovanou hodnotu, bude zahozen.

5.4.1 Návrh a sestavení topologie

Tato navržená úloha se opírá o reálný scénář, který je často aplikován v praxi. Jedná se o situaci, kdy je na směrovači poskytovatele služeb aplikovaná metoda omezování provozu. Tato situace nastává často v momentě, kdy si zákazník hradí nižší hodnotu propustnosti, než je linka poskytovatele schopná zákazníkovi dodat. Zároveň bude směrovač poskytovatele provoz klasifikovat na základě DSCP značek, které jsou paketům přiřazeny na hraničních směrovačích zákazníka. Klasifikovaný provoz bude rozdělovat do patřičných CBWFQ a LLQ front. V momentě, kdy je na směrovači poskytovatele aktivována metoda omezování, je více než vhodné na směrovače zákazníka aplikovat metodu tvarování provozu. Kromě tvarování provozu budou směrovače na hranicích zákaznických poboček provoz klasifikovat. Na základě této klasifikace dojde ke značkování jednotlivých paketů podle konceptu daného v podkapitole 4.2, tedy data budou označena značkou AF11, respektive AF13, VoIP provoz značkou EF, SSH provoz značkou AF42 nebo AF43.



Obrázek 5.5: Schéma implementace QoS nástrojů

5.4.2 Konfigurace ISP směrovače

Na směrovači poskytovatele služeb budou postupně implementovány dva přístupy k datům zákazníkových poboček. Nejprve dojde k aplikaci omezování provozu na hodnotu 50 Mbit/s. Toto omezování bude aktivní na všech rozhraních ve vstupním směru. Při aplikaci omezování bude využita klasifikační třída *match_any*, do které bude spadat bez výjimky všechen provoz.

```
[ISP]traffic classifier match_any
[ISP-classifier-match_any]if-match any
```

Následně bude vytvořeno provozní chování, kde bude aplikována metoda omezování provozu na hodnotu 50 Mbit/s.

```
[ISP]traffic behavior police
[ISP-behavior-police]car cir 50000 green pass yellow discard red discard
```

V poslední řadě dojde ke konfiguraci provozní politiky, která předešlou konfiguraci spojí dohromady. Zároveň je nutné provozní politiku aplikovat na všechna rozhraní ISP směrovače ve vstupním směru.

```
[ISP]traffic policy police
[ISP-policy-police]classifier match_any behavior police
```

```
[ISP]interface GigabitEthernet0/0/0
[ISP-GigabitEthernet0/0/0]traffic-policy police inbound
[ISP]interface GigabitEthernet0/0/1
[ISP-GigabitEthernet0/0/1]traffic-policy police inbound
[ISP]interface GigabitEthernet0/0/2
[ISP-GigabitEthernet0/0/2]traffic-policy police inbound
```

Předešlý blok konfigurace zajistí omezování veškerého provozu na 50 Mbit/s jdoucího na ISP směrovač z poboček zákazníka. Nyní je potřeba nakonfigurovat klasifikaci. ISP směrovač bude provoz klasifikovat na základě DSCP značek. Pro provoz obyčejných dat bude následně vytvořena CBWFQ fronta, které bude přiděleno 50 Mbit/s. Druhá CBWFQ fronta bude vytvořena pro SSH provoz a bude garantovat propustnost 2 Mbit/s. Poslední LLQ fronta bude zajišťovat provozu s DSCP značkou EF okamžité odbavení. Této frontě bude přiděleno 50 Mbit/s.

Na úvod je nutné nakonfigurovat klasifikační třídy zajišťující rozlišení provozu na základě DSCP značek.

```
[ISP]traffic classifier match_DATA
[ISP-classifier- match_DATA]if-match dscp af11 af13
[ISP]traffic classifier match_VOIP
[ISP-classifier- match_DATA]if-match dscp ef
[ISP]traffic classifier match_SSH
[ISP-classifier- match_DATA]if-match dscp af42 af43
```

Pro klasifikovaný provoz je nutné vytvořit provozní chování, kde budou nakonfigurovány jednotlivé fronty. Těmto frontám bude přiřazena určitá hodnota propustnosti.

```
[ISP]traffic behavior DATA_CBWFQ
[ISP-behavior-DATA_CBWFQ]queue af bandwidth 50000
[ISP]traffic behavior SSH_CBWFQ
[ISP-behavior- SSH_CBWFQ]queue af bandwidth 2000
[ISP]traffic behavior VOIP_LLQ
[ISP-behavior-VOIP_LLQ]queue llq bandwidth 50000
```

Klasifikační třídy a provozní chování budou v posledním kroku spojeny do jedné provozní politiky, která bude následně aplikována na všechna rozhraní v odchozím směru. Tato konfigurace zajistí, že klasifikovaný provoz jdoucí do jedné z poboček bude mít garantovanou určitou hodnotu propustnosti. Provoz citlivý na přenosové zpoždění bude mít navíc garantované bezodkladné odbavení pomocí fronty typu LLQ.

```
[ISP]traffic policy CBWFQ_LLQ
[ISP-policy-CBWFQ_LLQ]classifier match_DATA behavior DATA_CBWFQ
[ISP-policy-CBWFQ_LLQ]classifier match_VOIP behavior VOIP_LLQ
[ISP-policy-CBWFQ_LLQ]classifier match_SSH behavior SSH_CBWFQ
```

```
[ISP]interface GigabitEthernet0/0/0
[ISP-GigabitEthernet0/0/0]traffic-policy CBWFQ_LLQ outbound
[ISP]interface GigabitEthernet0/0/1
[ISP-GigabitEthernet0/0/1]traffic-policy CBWFQ_LLQ outbound
[ISP]interface GigabitEthernet0/0/2
[ISP-GigabitEthernet0/0/2]traffic-policy CBWFQ_LLQ outbound
```

5.4.3 **Konfigurace RZ1 směrovače**

Na směrovači RZ1 bude aplikováno tvarování provozu. Provoz, který by byl v případě překročení hodnoty 50 Mbit/s směrovačem ISP zahozen bude nástrojem tvarování provozu pozdržen ve frontě. Bude tedy nakonfigurována rodičovská provozní politika zajišťující tvarování provozu na 50 Mbit/s. Součásti této politiky budou tři podtřídy zajišťující rozdělení provozu na základě klasifikace do dvou CBWFQ front a do jedné LLQ fronty. Těmto frontám bude garantována určitá hodnota propustnosti. V rámci podtříd bude rovněž nakonfigurováno značkování provozu dle zavedeného konceptu. Pro klasifikaci budou využity již nakonfigurované třídy z kapitoly 4.2 popisující klasifikaci a značkování provozu.

Nejprve bude nakonfigurována sekundární provozní politika, jejíž obsahem budou tři podtřídy zajišťující klasifikaci provozu. Provoz rozlišen těmito třídami bude rozřazen do dvou CBWFQ a jedné LLQ fronty. CBWFQ frontě určené pro data bude přiděleno 30 Mbit/s. SSH provozu, který je přiřazen do druhé CBWFQ fronty, bude přiděleno 1 Mbit/s. Poslední LLQ fronta bude mít přiřazeno 19 Mbit/s. Kromě rozřazení do jednotlivých front bude rovněž zajištěno značkování provozu.

```
RZ1(config)#policy-map CBWFQ_LLQ
RZ1(config-pmap)#class match_DATA
RZ1(config-pmap-c)#bandwidth 30000
RZ1(config-pmap-c)#set dscp af11
RZ1(config-pmap)#class match_VOIP
RZ1(config-pmap-c)#priority 19000
RZ1(config-pmap-c)#set dscp ef
RZ1(config-pmap)#class match_SSH_by_DSCP
RZ1(config-pmap-c)#bandwidth 1000
```

Následně bude vytvořena hlavní provozní politika, která bude zajišťovat tvarování veškerého provozu jdoucího z pobočky 1 na RZ1 směrovač na 50 Mbit/s.

```
RZ1(config)#policy-map shaping
RZ1(config-pmap)#class class-default
RZ1(config-pmap-c)#shape average 50000000
RZ1(config-pmap-c)#service-policy CBWFQ_LLQ
```

Provozní politiku je potřeba aplikovat na rozhraní GigabitEthernet0/0 v odchozím směru. Pro správnou funkčnost je rovněž nutné mít na směrovači RZ1 aplikovanou provozní politiku rz1_toLocal_input z kapitoly 4.2 zajišťující klasifikaci a značkování SSH provozu.

```
RZ1(config)#interface GigabitEthernet0/0
RZ1(config-if)#service-policy output shaping
RZ1(config)#interface GigabitEthernet0/1
RZ1(config-if)#service-policy input rz1_toLocal_input
```

5.4.4 **Konfigurace RZ2 směrovače**

Na RZ2 směrovači výrobce Huawei bude rovněž aplikované tvarování provozu na 50 Mbit/s. Dále bude aktivní jedna CBWFQ fronta, která bude provozu obyčejných dat garantovat propustnost 25 Mbit/s. Tento provoz bude značkován DSCP značkou AF13. Dále bude aktivní jedna LLQ fronta pro provoz citlivý na přenosové zpoždění. Tento provoz bude označkován DSCP značkou EF.

Pro klasifikaci provozu budou využity klasifikační třídy z kapitoly 4.2. Je tedy potřeba nakonfigurovat provozní chování, které bude jednak vytvářet CBWFQ a LLQ fronty a zároveň bude zajišťovat značkování provozu.

```
[RZ2]traffic behavior DATA_CBWFQ
[RZ2-behavior-DATA_CBWFQ]queue af bandwidth 25000
[RZ2-behavior-DATA_CBWFQ]remark dscp af13
[RZ2]traffic behavior VOIP_LLQ
[RZ2-behavior-VOIP_LLQ]queue llq bandwidth 25000
[RZ2-behavior-VOIP_LLQ] remark dscp ef
```

Nyní je potřeba nakonfigurovat provozní politiku, která spojí nakonfigurované klasifikační třídy a provozní chování.

```
[RZ2]traffic policy rz2_toNet_output
[RZ2-policy-rz2_toNet_output]classifier match_DATA behavior DATA_CBWFQ
[RZ2-policy-rz2_toNet_output]classifier match_VOIP behavior VOIP_LLQ
```

V posledním kroku dojde k připojení provozní politiky na fyzické rozhraní GigabitEthernet0/0/0. Zároveň dojde ke konfiguraci tvarování provozu.

```
[RZ2]interface GigabitEthernet0/0/0
[RZ2- GigabitEthernet0/0/0]qos gts cir 50000
[RZ2- GigabitEthernet0/0/0]traffic-policy rz2_toNet_output outbound
```

5.4.5 **Konfigurace RZ3 směrovače**

Konfigurace RZ3 směrovače bude podobná konfiguraci směrovače RZ1. Bude nakonfigurovaná hlavní provozní politika zajišťující samotné tvarování provozu. V roli potomka bude aplikovaná provozní

politika, která s pomocí klasifikačních tříd zajišťuje rozřazení provozu do jednotlivých front. Opět budou aktivní dvě CBWFQ fronty pro data a SSH provoz a jedna LLQ fronta pro provoz citlivý na přenosové zpoždění. Při rozřazení provozu do jednotlivých front dojde i k jeho označkování DSCP značkami.

```
RZ3(config)#policy-map CBWFQ_LLQ
RZ3(config-pmap)#class match_DATA
RZ3(config-pmap-c)#bandwidth 25000
RZ3(config-pmap-c)#set dscp af13
RZ3(config-pmap)#class match_VOIP
RZ3(config-pmap-c)#priority 24000
RZ3(config-pmap-c)#set dscp ef
RZ3(config-pmap)#class match_SSH_response_DSCP
RZ3(config-pmap-c)#bandwidth 1000
```

Hlavní provozní politika bude zajišťovat tvarování na 50 Mbit/s.

```
RZ3(config)#policy-map shaping
RZ3(config-pmap)#class class-default
RZ3(config-pmap-c)#shape average 50000000
RZ3(config-pmap-c)#service-policy CBWFQ_LLQ
```

Provozní politiku shaping je nutné aplikovat na rozhraní GigabitEthernet0/0 v odchozím směru. Rovněž je nutné zajistit klasifikaci a značkování SSH provozu pomocí již vytvořené třídy rz3_toLocal_input.

```
RZ3(config)#interface GigabitEthernet0/0
RZ3(config-if)#service-policy output shaping
RZ3(config)#interface GigabitEthernet0/1
RZ3(config-if)#service-policy input rz3_toLocal_input
```

5.4.6 **Ověření omezování, tvarování a prioritizace provozu**

Nejprve byl pro orientační měření použit nástroj Iperf3. Chování směrovače ke generovaném provozu lze ověřit následovně.

```
RZ3#show policy-map interface
 GigabitEthernet0/0
  Service-policy output: shaping
    Class-map: class-default (match-any)
      42900 packets, 61413457 bytes
      5 minute offered rate 1623000 bps, drop rate 1000 bps
      Match: any
      Queueing
      queue limit 64 packets
      (queue depth/total drops/no-buffer drops) 0/25/0
      (pkts output/bytes output) 42875/63946926
      shape (average) cir 51000000, bc 204000, be 204000
      target shape rate 51000000
      Service-policy : CBWFQ_LLQ
        queue stats for all priority classes:
```

```
    Queueing
    queue limit 64 packets
    (queue depth/total drops/no-buffer drops) 0/0/0
    (pkts output/bytes output) 15/1290
   Class-map: match_DATA (match-all)
    42857 packets, 61409555 bytes
    5 minute offered rate 1623000 bps, drop rate 2000 bps
    Match: access-group 111
    Match: not  dscp af42 (36)
    Queueing
    queue limit 64 packets
    (queue depth/total drops/no-buffer drops) 0/25/0
    (pkts output/bytes output) 42832/63942844
    bandwidth 25000 kbps
    QoS Set
     dscp af13
       Packets marked 42859
```

Z výpisu je patrné, že malé množství paketů bylo v důsledku tvarování provozu zahozeno. I přesto ale použití tohoto nástroje na straně pobočky zákazníka předešlo přísnějšímu omezování provozu, které je aplikováno na straně poskytovatele.

```
[ISP]display traffic policy statistics interface GigabitEthernet 0/0/2 inbound
Interface: GigabitEthernet0/0/2
Traffic policy inbound: police
 Rule number: 1
 Current status: OK!
 Item              Sum(Packets/Bytes)          Rate(pps/bps)
 -----------------------------------------------------------------------------
 Matched                42,896/64,977,772            1/200
   Passed               42,896/64,977,772            1/200
   Dropped                      0/0                  0/0
     Filter                     0/0                  0/0
     CAR                        0/0                  0/0
   Queue Matched                0/0                  0/0
     Enqueued                   0/0                  0/0
     Discarded                  0/0                  0/0
   CAR                  42,896/64,977,772            1/200
     Green packets      42,896/64,977,772            1/200
     Yellow packets             0/0                  0/0
     Red packets                0/0                  0/0
```

Při ověřování výsledků byl následně přístrojem ParaScope GigE generován provoz z pobočky 2 do pobočky 3. Nejprve byl generován 100 Mbit/s, který simuloval tok obyčejných dat. Klasifikovaný provoz tedy spadal do CBWFQ fronty, které bylo přiděleno 25 Mbit/s. Generovanému provozu na RZ2 stálo v

cestě tvarování provozu na 50 Mbit/s. Jelikož nebyl generován žádný VoIP provoz, byla přidělena propustnost LLQ frontě předána CBWFQ frontě.

Tabulka 5.5: Naměřené hodnoty přenosových parametrů (Omezování, tvarování, CBWFQ, LLQ – AF13)

	Odesláno	Přijato
Propustnost (Mbps)	100.00	46.96
Počet paketů	738045	352438
Počet bajtů	738044952	352438000

Následně byl generován provoz simulující data náchylná na přenosové parametry. Jednalo se tedy o provoz, který byl klasifikován jako VoIP, označen značkou EF a následně přiřazen do LLQ fronty, které byla nastavena propustnost 25 Mbit/s. Pro LLQ fronty je sice garantované bezprostřední odbavení, nastavená propustnost je ale rovněž propustnost maximální.

Tabulka 5.6: Naměřené hodnoty přenosových parametrů (Omezování, tvarování, CBWFQ, LLQ – EF)

	Odesláno	Přijato
Propustnost (Mbps)	100.00	23.56
Počet paketů	683191	161090
Počet bajtů	683190880	161090000

Závěr

Cílem této práce byla implementace mechanismů kvality služby v sítích IPsec VPN na zařízeních výrobců Cisco a Huawei. Na navržených scénářích došlo k ověření jak kompatibility zařízení dvou výrobců, ale rovněž k ověření samotné implementace pomocí zapůjčeného měřícího přístroje. Diplomová práce popisuje implementaci samotné IPsec VPN, postup při konfiguraci klasifikace a značkování provozu a následně dojde k návrhu a realizaci topologií využívající konkrétní nástroje kvality služeb. Diplomová práce poukazuje a následně řeší problémy, které vznikají buď součinností dvou implementovaných technologií, nebo například chybnou implementací nástroje v použité verzi operačního systému na zařízeních Cisco.

Veškeré použité nástroje byly na zařízeních dvou výrobců kompatibilní. K odlišnostem docházelo pouze ve způsobu konfigurace jednotlivých nástrojů. Tato skutečnost může být zdrojem problémů už při konfiguraci samotné IPsec VPN. Při konfiguraci IPsec VPN je většinou nutné na obou stranách navazujících IPsec spojení nastavit totožné parametry. Nicméně, konfigurace na zařízeních výrobců Cisco a Huawei probíhá odlišným způsobem. Liší se jak jednotlivé příkazy, tak i postup, který je nutné při konfiguraci dodržet. Zde jednoduše dochází k chybám, kvůli kterým nedojde ke správnému navázání IPsec tunelu. Tuto komplikaci si ale uvědomují i samotní výrobci, a proto Huawei nabízí vzorové konfigurace, kde na tyto odlišnosti poukazuje a názorně je vysvětluje. Při následné konfiguraci ať už klonování záhlaví pomocí nástroje qos pre-classify, nebo konfiguraci klasifikace a značkování se postupuje podobným způsobem. Při klasifikaci byl v jednom případě na zařízeních Cisco využit nástroj NBAR. Ekvivalent tohoto nástroje nabízí pod speciální licencí i zařízení výrobce Huawei pod názvem Service Awareness nebo Smart Application Control. K využití tohoto nástroje nedošlo z důvodu absence této licence na zařízeních ve školní laboratoři. Komplikace s licencí byly řešeny i na zařízeních výrobce Cisco. Licence, která byla na zařízeních k dispozici omezovala propustnost každého navázaného IPsec tunelu na 85 Mbit/s. Naměřená propustnost v případě překročení tohoto limitu byla až o polovinu nižší. Tato komplikace byla následně řešena dvěma způsoby. Nejprve došlo na zařízeních Cisco i Huawei k aplikaci omezování provozu na hodnotu nižší, než je hodnota daná licencí. Naměřená propustnost už odpovídala hodnotě předpokládané. Následně bylo popsáno řešení i pomocí metody tvarování provozu. Na závěr byl navržen scénář, který obsahoval využití jak omezování a tvarování provozu, ale rovněž také využití mechanismů obsluhy paketových front CBWFQ a LLQ.

Přínosem této práce je popis implementace mechanismů kvality služby v prostředí IPsec VPN sítích s využitím zařízení Cisco a Huawei. V diplomové práci byly vyobrazeny rozdíly při konfiguraci těchto zařízení. Implementační část kromě návrhu třech scénářů s využitím jednotlivých nástrojů kvality služby řeší i primární problémy, které při součinnosti nástrojů kvality služby a IPsec VPN technologie mohou vzniknout.

Použitá literatura

[1] ODOM, Wendell. a Michael J. CAVANAUGH. *Cisco QOS exam certification guide*. 2nd ed. Indianapolis, IN: Cisco Press, 2005. ISBN 978-1-58720-124-0.

[2] SZIGETI, Tim, Christina HATTINGH, Robert BARTON a Kenneth BRILEY. *End-to-end QoS network design*. 2nd edition. Indianapolis, IN: Cisco Press, 2014. Cisco Press networking technology series. ISBN 978-158-7143-694.

[3] BOUŠKA, Petr. VPN - 1 IPsec VPN a Cisco [online]. [cit. 2021-5-4]. Dostupné z: https://www.samuraj-cz.com/clanek/vpn-1-ipsec-vpn-a-cisco/

[4] RFC 2401 [online]. [cit. 2021-5-4]. Dostupné z: https://tools.ietf.org/html/rfc2401#section-1

[5] RFC 4302 [online]. [cit. 2021-5-4]. Dostupné z: https://tools.ietf.org/html/rfc4302

[6] Protokol AH [online]. [cit. 2021-5-4]. Dostupné z: https://www.ibm.com/docs/cs/i/7.3?topic=protocols-authentication-header

[7] RFC 4303 [online]. [cit. 2021-5-4]. Dostupné z: https://tools.ietf.org/html/rfc4303#section-1

[8] IPsec Security Associations Overview [online]. [cit. 2021-5-4]. Dostupné z: https://www.juniper.net/documentation/us/en/software/junos/security-services/topics/topic-map/ipsec-security-associations-overview.html

[9] Transport mode and tunnel mode [online]. [cit. 2021-5-7]. Dostupné z: https://www.ibm.com/docs/en/zos/2.4.0?topic=encapsulation-transport-mode-tunnel-mode

[10] MACHNIK, Petr. Pokročilé síťové technologie. Ostrava, 2021. Skripta. VŠB-Technická univerzita Ostrava.

[11] LAUTERBACH, Filip. Kvalita služby v sítích MPLS VPN [online]. Ostrava, 2021 [cit. 2022-04-15]. Dostupné z: http://hdl.handle.net/10084/143838. Diplomová práce. Vysoká škola báňská – Technická univerzita Ostrava.

Seznam příloh

Příloha A

Konfigurace IPsec VPN topologie

Konfigurace směrovače RZ1

```
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RZ1
!
boot-start-marker
boot-end-marker
!
no aaa new-model
bsd-client server url https://cloudsso.cisco.com/as/token.oauth2
!
ip cef
no ipv6 cef
!
multilink bundle-name authenticated
!
cts logging verbose
!
license udi pid CISCO2901/K9 sn FCZ1937C19J
!
redundancy
!
crypto isakmp policy 10
 encr aes 256
 authentication pre-share
 group 5
crypto isakmp key heslo address 10.0.0.2
crypto isakmp key heslo address 10.0.2.2
!
crypto ipsec transform-set myset esp-aes 256 esp-sha-hmac
 mode tunnel
crypto ipsec transform-set myset1 esp-aes 256 esp-sha-hmac
 mode transport
```

```
!
crypto map ipsec_map 10 ipsec-isakmp
 set peer 10.0.2.2
 set transform-set myset1
 match address 100
crypto map ipsec_map 20 ipsec-isakmp
 set peer 10.0.0.2
 set transform-set myset
 match address 101
!
interface Tunnel1
 ip address 1.1.1.1 255.255.255.252
 tunnel source GigabitEthernet0/0
 tunnel destination 10.0.2.2
 crypto map ipsec_map
 ip mtu 1440
!
interface GigabitEthernet0/0
 ip address 10.0.1.2 255.255.255.0
 duplex auto
 speed auto
 crypto map ipsec_map
!
interface GigabitEthernet0/1
 ip address 192.168.0.1 255.255.255.0
 duplex auto
 speed auto
!
router ospf 1
 network 10.0.1.0 0.0.0.255 area 0
 network 192.168.0.0 0.0.0.255 area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 10.0.1.1
ip route 192.168.2.0 255.255.255.0 1.1.1.2
!
!
!
```

```
access-list 100 permit gre host 10.0.1.2 host 10.0.2.2
access-list 101 permit ip 192.168.0.0 0.0.0.255 192.168.1.0 0.0.0.255
!
control-plane
!
!
!
line con 0
line aux 0
line 2
 no activation-character
 no exec
 transport preferred none
 transport output pad telnet rlogin lapb-ta mop udptn v120 ssh
 stopbits 1
line vty 0 4
 login
 transport input none
!
scheduler allocate 20000 1000
!
end
```

Konfigurace směrovače RZ2

```
[V200R003C00SPC200]
#
 snmp-agent local-engineid 800007DB030819A69A8275
 snmp-agent
#
cwmp
 cwmp cpe connect retry 0
#
 http timeout 3
#
 drop illegal-mac alarm
#
dhcp enable
#
undo dhcp server bootp
#
acl number 3000
 rule 5 permit ip source 192.168.1.0 0.0.0.255 destination 192.168.0.0 0.0.0.255
acl number 3001
 rule 5 permit ip source 192.168.1.0 0.0.0.255 destination 192.168.2.0 0.0.0.255
#
ipsec proposal tran1
 esp authentication-algorithm sha1
 esp encryption-algorithm aes-256
#
ike proposal 1
 encryption-algorithm aes-cbc-256
 dh group5
#
ike peer cisco v1
 pre-shared-key simple heslo
 ike-proposal 1
 remote-address 10.0.2.2
ike peer cisco2 v1
 pre-shared-key simple heslo
 ike-proposal 1
 remote-address 10.0.1.2
#
ipsec policy map1 1 isakmp
 security acl 3001
 ike-peer cisco
```

```
 proposal tran1
ipsec policy map1 2 isakmp
 security acl 3000
 ike-peer cisco2
 proposal tran1
#
aaa
 authentication-scheme default
 authorization-scheme default
 accounting-scheme default
 domain default
 domain default_admin
 local-user admin password cipher %$%$=i~>Xp&aY+*2cEVcS-A23Uwe%$%$
 local-user admin service-type http
#
firewall zone Local
 priority 128
#
interface GigabitEthernet0/0/0
 ip address 10.0.0.2 255.255.255.0
 ipsec policy map1
 mtu 1445
#
interface GigabitEthernet0/0/1
 ip address 192.168.1.1 255.255.255.0
#
interface GigabitEthernet0/0/2
#
interface Cellular0/0/0
 link-protocol ppp
#
interface Cellular0/0/1
 link-protocol ppp
#
interface NULL0
#
ospf 1
 area 0.0.0.0
  network 10.0.0.0 0.0.0.255
  network 192.168.1.0 0.0.0.255
#
user-interface con 0
```

```
 authentication-mode password
 set authentication password cipher
%$%$Is3>#""U3)`w':%"+:z),2f\0gpi>q[R,&GX.xR7i%T%2f_,%$%$
 user-interface vty 0 4
 #
 wlan ac
 #
 voice
 #
 diagnose
 #
Return
```

Konfigurace směrovače RZ3

```
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RZ3
!
boot-start-marker
boot-end-marker
!
no aaa new-model
bsd-client server url https://cloudsso.cisco.com/as/token.oauth2
!
ip cef
no ipv6 cef
!
multilink bundle-name authenticated
!
cts logging verbose
!
license udi pid CISCO2901/K9 sn FCZ1937C19F
!
redundancy
!
crypto isakmp policy 10
 encr aes 256
 authentication pre-share
 group 5
crypto isakmp key heslo address 10.0.0.2
crypto isakmp key heslo address 10.0.1.2
!
crypto ipsec transform-set myset esp-aes 256 esp-sha-hmac
 mode tunnel
crypto ipsec transform-set myset1 esp-aes 256 esp-sha-hmac
 mode transport
!
crypto map ipsec_map 10 ipsec-isakmp
 set peer 10.0.0.2
 set transform-set myset
 match address 100
crypto map ipsec_map 20 ipsec-isakmp
```

```
 set peer 10.0.1.2
 set transform-set myset1
 match address 101
!
interface Tunnel1
 ip address 1.1.1.2 255.255.255.252
 tunnel source GigabitEthernet0/0
 tunnel destination 10.0.1.2
 crypto map ipsec_map
!
interface GigabitEthernet0/0
 ip address 10.0.2.2 255.255.255.0
 duplex auto
 speed auto
 crypto map ipsec_map
!
interface GigabitEthernet0/1
 ip address 192.168.2.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/1/0
 no ip address
 shutdown
 clock rate 2000000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 2000000
!
router ospf 1
 network 10.0.2.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 10.0.2.1
ip route 192.168.0.0 255.255.255.0 1.1.1.1
```

```
 !
 access-list 100 permit ip 192.168.2.0 0.0.0.255 192.168.1.0 0.0.0.255
 access-list 101 permit gre host 10.0.2.2 host 10.0.1.2
 !
 control-plane
 !
 line con 0
 line aux 0
 line 2
  no activation-character
  no exec
  transport preferred none
  transport output pad telnet rlogin lapb-ta mop udptn v120 ssh
  stopbits 1
 line vty 0 4
  login
  transport input none
 !
 scheduler allocate 20000 1000
 !
end
```

Konfigurace směrovače ISP

```
[V200R005C20SPC200]
#
 drop illegal-mac alarm
#
dhcp enable
#
pki realm default
 enrollment self-signed
#
aaa
 authentication-scheme default
 authorization-scheme default
 accounting-scheme default
 domain default
 domain default_admin
 local-user admin password irreversible-cipher
%@%@}*u6!8e,8+a4YR42e^>ULT'[`Kc!Iba$>AIT"~>;W{GCT'^L%@
 local-user admin service-type http
#
firewall zone Local
 priority 64
#
interface GigabitEthernet0/0/0
 ip address 10.0.1.1 255.255.255.0
#
interface GigabitEthernet0/0/1
 ip address 10.0.0.1 255.255.255.0
#
interface GigabitEthernet0/0/2
 ip address 10.0.2.1 255.255.255.0
#
interface Cellular0/0/0
#
interface Cellular0/0/1
#
interface NULL0
#
ospf 1
 area 0.0.0.0
  network 10.0.0.0 0.0.0.255
  network 10.0.1.0 0.0.0.255
```

```
 network 10.0.2.0 0.0.0.255
#
 snmp-agent local-engineid 800007DB030819A69B6D4D
#
user-interface con 0
 authentication-mode password
 set authentication password cipher %@%@!hfe<ML7M"j]f=BgO,S4,2S3kgyk-
+6)9~L6p24+EBO,2S6,%@%@
user-interface vty 0 4
#
wlan ac
#
voice
 #
 diagnose
#
return
```

Příloha B

QoS pre-classify test na RZ3 směrovači

Konfigurace RZ3 směrovače

```
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RZ3
!
boot-start-marker
boot-end-marker
!
no aaa new-model
bsd-client server url https://cloudsso.cisco.com/as/token.oauth2
!
ip cef
no ipv6 cef
!
multilink bundle-name authenticated
!
cts logging verbose
!
license udi pid CISCO2901/K9 sn FCZ1937C19F
!
redundancy
!
class-map match-all test_esp
 match access-group 121
class-map match-all test_gre
 match access-group 122
class-map match-all test_icmp
 match access-group 123
!
policy-map test_output
 class test_esp
 class test_gre
 class test_icmp
!
```

```
crypto isakmp policy 10
 encr aes 256
 authentication pre-share
 group 5
crypto isakmp key heslo address 10.0.0.2
crypto isakmp key heslo address 10.0.1.2
!
crypto ipsec transform-set myset esp-aes 256 esp-sha-hmac
 mode tunnel
crypto ipsec transform-set myset1 esp-aes 256 esp-sha-hmac
 mode transport
!
crypto map ipsec_map 10 ipsec-isakmp
 set peer 10.0.0.2
 set transform-set myset
 match address 100
 qos-preclassify
crypto map ipsec_map 20 ipsec-isakmp
 set peer 10.0.1.2
 set transform-set myset1
 match address 101
!
interface Tunnel1
 ip address 1.1.1.2 255.255.255.252
 tunnel source GigabitEthernet0/0
 tunnel destination 10.0.1.2
 crypto map ipsec_map
 qos-preclassify
 ip mtu 1440
!
interface GigabitEthernet0/0
 ip address 10.0.2.2 255.255.255.0
 duplex auto
 speed auto
 crypto map ipsec_map
 service-policy output test_output
!
interface GigabitEthernet0/1
 ip address 192.168.2.1 255.255.255.0
 duplex auto
 speed auto
!
```

```
interface Serial0/1/0
 no ip address
 shutdown
 clock rate 2000000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 2000000
!
router ospf 1
 network 10.0.2.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 10.0.2.1
ip route 192.168.0.0 255.255.255.0 1.1.1.1
!
access-list 100 permit ip 192.168.2.0 0.0.0.255 192.168.1.0 0.0.0.255
access-list 101 permit gre host 10.0.2.2 host 10.0.1.2
access-list 121 permit esp any any
access-list 122 permit gre any any
access-list 123 permit icmp any any
!
control-plane
!
line con 0
line aux 0
line 2
 no activation-character
 no exec
 transport preferred none
 transport output pad telnet rlogin lapb-ta mop udptn v120 ssh
 stopbits 1
line vty 0 4
 login
 transport input none
!
```

```
scheduler allocate 20000 1000
!
end
```

Příloha C

Konfigurace klasifikace a značkování

Konfigurace RZ1 směrovače

```
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RZ1
!
boot-start-marker
boot-end-marker
!
no aaa new-model
bsd-client server url https://cloudsso.cisco.com/as/token.oauth2
!
ip cef
no ipv6 cef
!
multilink bundle-name authenticated
!
cts logging verbose
!
license udi pid CISCO2901/K9 sn FCZ1937C19J
!
redundancy
!
class-map match-all match_VOIP
 match access-group 111
class-map match-all match_DATA
 match access-group 110
 match not dscp af43
class-map match-all match_SSH
 match protocol ssh
 match access-group 110
class-map match-all match_SSH_by_DSCP
 match dscp af43
!
policy-map rz1_toLocal_input
```

```
 class match_SSH
  set dscp af43
policy-map marking
 class match_DATA
  set dscp af11
 class match_VOIP
  set dscp ef
!
crypto isakmp policy 10
 encr aes 256
 authentication pre-share
 group 5
crypto isakmp key heslo address 10.0.0.2
crypto isakmp key heslo address 10.0.2.2
!
crypto ipsec transform-set myset esp-aes 256 esp-sha-hmac
 mode tunnel
crypto ipsec transform-set myset1 esp-aes 256 esp-sha-hmac
 mode transport
!
crypto map ipsec_map 10 ipsec-isakmp
 set peer 10.0.2.2
 set transform-set myset1
 match address 100
crypto map ipsec_map 20 ipsec-isakmp
 set peer 10.0.0.2
 set transform-set myset
 match address 101
 qos pre-classify
!
interface Tunnel1
 ip address 1.1.1.1 255.255.255.252
 ip mtu 1440
 ip nbar protocol-discovery
 qos pre-classify
 tunnel source GigabitEthernet0/0
 tunnel destination 10.0.2.2
 crypto map ipsec_map
!
interface Embedded-Service-Engine0/0
 no ip address
 shutdown
```

```
!
interface GigabitEthernet0/0
 ip address 10.0.1.2 255.255.255.0
 duplex auto
 speed auto
 crypto map ipsec_map
 service-policy output marking
!
interface GigabitEthernet0/1
 ip address 192.168.0.1 255.255.255.0
 ip nbar protocol-discovery
 duplex auto
 speed auto
 service-policy input rz1_toLocal_input
!
interface Serial0/1/0
 no ip address
 shutdown
 clock rate 2000000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 2000000
!
router ospf 1
 network 10.0.1.0 0.0.0.255 area 0
 network 192.168.0.0 0.0.0.255 area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 10.0.1.1
ip route 192.168.2.0 255.255.255.0 1.1.1.2
!
!
!
access-list 100 permit gre host 10.0.1.2 host 10.0.2.2
access-list 101 permit ip 192.168.0.0 0.0.0.255 192.168.1.0 0.0.0.255
access-list 110 permit ip host 192.168.0.10 any
```

```
 access-list 111 permit ip host 192.168.0.20 any
 !
 control-plane
 !
 !
 !
 line con 0
 line aux 0
 line 2
  no activation-character
  no exec
  transport preferred none
  transport output pad telnet rlogin lapb-ta mop udptn v120 ssh
  stopbits 1
 line vty 0 4
  login
  transport input none
 !
 scheduler allocate 20000 1000
 !
end
```

Konfigurace RZ2 směrovače

```
[V200R003C00SPC200]
#
 snmp-agent local-engineid 800007DB030819A69A8275
 snmp-agent
#
cwmp
 cwmp cpe connect retry 0
#
 http timeout 3
#
 drop illegal-mac alarm
#
dhcp enable
#
undo dhcp server bootp
#
acl number 3000
 rule 5 permit ip source 192.168.1.0 0.0.0.255 destination 192.168.0.0 0.0.0.255
acl number 3001
 rule 5 permit ip source 192.168.1.0 0.0.0.255 destination 192.168.2.0 0.0.0.255
acl number 3021
 rule 5 permit ip source 192.168.1.10 0
acl number 3022
 rule 5 permit ip source 192.168.1.20 0
#
ipsec proposal tran1
 esp authentication-algorithm sha1
 esp encryption-algorithm aes-256
#
ike proposal 1
 encryption-algorithm aes-cbc-256
 dh group5
#
ike peer cisco v1
 pre-shared-key simple heslo
 ike-proposal 1
 remote-address 10.0.2.2
ike peer cisco2 v1
 pre-shared-key simple heslo
 ike-proposal 1
 remote-address 10.0.1.2
```

```
#
ipsec policy map1 1 isakmp
 security acl 3001
 ike-peer cisco
 proposal tran1
 qos pre-classify
ipsec policy map1 2 isakmp
 security acl 3000
 ike-peer cisco2
 proposal tran1
 qos pre-classify
#
traffic classifier match_any operator or
 if-match any
traffic classifier match_DATA operator or
 if-match acl 3021
traffic classifier match_VOIP operator or
 if-match acl 3022
#
traffic behavior beh_VOIP
 remark dscp ef
 statistic enable
traffic behavior beh_DATA
 remark dscp af13
 statistic enable
#
traffic policy marking
 classifier match_DATA behavior beh_DATA
 classifier match_VOIP behavior beh_VOIP
#
aaa
 authentication-scheme default
 authorization-scheme default
 accounting-scheme default
 domain default
 domain default_admin
 local-user admin password cipher %$%$=i~>Xp&aY+*2cEVcS-A23Uwe%$%$
 local-user admin service-type http
#
firewall zone Local
 priority 128
#
```

```
interface GigabitEthernet0/0/0
 ip address 10.0.0.2 255.255.255.0
 traffic-policy marking outbound
 ipsec policy map1
 mtu 1445
#
interface GigabitEthernet0/0/1
 ip address 192.168.1.1 255.255.255.0
#
interface GigabitEthernet0/0/2
#
interface Cellular0/0/0
 link-protocol ppp
#
interface Cellular0/0/1
 link-protocol ppp
#
interface NULL0
#
ospf 1
 area 0.0.0.0
  network 10.0.0.0 0.0.0.255
  network 192.168.1.0 0.0.0.255
#
user-interface con 0
 authentication-mode password
 set authentication password cipher
%$%$Is3>#""U3)`w':%"+:z),2f\0gpi>q[R,&GX.xR7i%T%2f_,%$%$
user-interface vty 0 4
#
wlan ac
#
voice
 #
diagnose
```

Konfigurace RZ3 směrovače

```
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
```

```
!
hostname RZ3
!
boot-start-marker
boot-end-marker
!
no aaa new-model
bsd-client server url https://cloudsso.cisco.com/as/token.oauth2
!
ip cef
no ipv6 cef
!
multilink bundle-name authenticated
!
cts logging verbose
!
license udi pid CISCO2901/K9 sn FCZ1937C19F
!
redundancy
!
class-map match-all match_VOIP
 match access-group 110
class-map match-all match_SSH_response
 match protocol ssh
class-map match-all match_DATA
 match access-group 111
 match not dscp af42
class-map match-all match_SSH_response_DSCP
 match dscp af42
!
policy-map rz3_toLocal_input
 class match_SSH_response
  set dscp af42
policy-map marking
 class match_DATA
  set dscp af13
 class match_VOIP
  set dscp ef
!
crypto isakmp policy 10
 encr aes 256
 authentication pre-share
```

```
 group 5
crypto isakmp key heslo address 10.0.0.2
crypto isakmp key heslo address 10.0.1.2
!
crypto ipsec transform-set myset esp-aes 256 esp-sha-hmac
 mode tunnel
crypto ipsec transform-set myset1 esp-aes 256 esp-sha-hmac
 mode transport
!
crypto map ipsec_map 10 ipsec-isakmp
 set peer 10.0.0.2
 set transform-set myset
 match address 100
 qos pre-classify
crypto map ipsec_map 20 ipsec-isakmp
 set peer 10.0.1.2
 set transform-set myset1
 match address 101
!
interface Tunnel1
 ip address 1.1.1.2 255.255.255.252
 ip mtu 1440
 ip tcp adjust-mss 1360
 qos pre-classify
 tunnel source GigabitEthernet0/0
 tunnel destination 10.0.1.2
 crypto map ipsec_map
!
interface Embedded-Service-Engine0/0
 no ip address
 shutdown
!
interface GigabitEthernet0/0
 ip address 10.0.2.2 255.255.255.0
 duplex auto
 speed auto
 crypto map ipsec_map
 service-policy output marking
!
interface GigabitEthernet0/1
 ip address 192.168.2.1 255.255.255.0
 ip nbar protocol-discovery
```

```
 duplex auto
 speed auto
 service-policy input rz3_toLocal_input
!
interface Serial0/1/0
 no ip address
 shutdown
 clock rate 2000000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 2000000
!
router ospf 1
 network 10.0.2.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 10.0.2.1
ip route 192.168.0.0 255.255.255.0 1.1.1.1
!
access-list 100 permit ip 192.168.2.0 0.0.0.255 192.168.1.0 0.0.0.255
access-list 101 permit gre host 10.0.2.2 host 10.0.1.2
access-list 110 permit ip host 192.168.2.20 any
access-list 111 permit ip host 192.168.2.10 any
!
control-plane
!
line con 0
line aux 0
line 2
 no activation-character
 no exec
 transport preferred none
 transport output pad telnet rlogin lapb-ta mop udptn v120 ssh
 stopbits 1
line vty 0 4
```

```
 login
 transport input none
!
scheduler allocate 20000 1000
!
end
```

Příloha D

Řešení problémového značkování

Konfigurace RZ1 směrovače

```
hostname RZ1
!
boot-start-marker
boot-end-marker
!
no aaa new-model
!
resource policy
!
mmi polling-interval 60
no mmi auto-configure
no mmi pvc
mmi snmp-timeout 180
ip subnet-zero
ip cef
!
no ip dhcp use vrf connected
!
no ip ips deny-action ips-interface
!
crypto isakmp policy 10
 encr aes 256
 authentication pre-share
 group 5
crypto isakmp key heslo address 10.0.2.2
!
crypto ipsec transform-set myset esp-aes 256 esp-sha-hmac
!
crypto ipsec profile profile
 set transform-set myset
!
interface Tunnel0
 ip address 1.1.1.1 255.255.255.252
 tunnel source 10.0.1.2
 tunnel destination 10.0.2.2
 tunnel mode ipsec ipv4
```

```
 tunnel protection ipsec profile profile
 ip mtu 1445
!
interface FastEthernet0/0
 ip address 10.0.1.2 255.255.255.0
 duplex auto
 speed auto
!
interface FastEthernet0/1
 ip address 192.168.0.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/1/0
 no ip address
 shutdown
 clock rate 125000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 125000
!
router ospf 1
 log-adjacency-changes
 network 10.0.1.0 0.0.0.255 area 0
 network 192.168.0.0 0.0.0.255 area 0
!
ip classless
ip route 192.168.2.0 255.255.255.0 Tunnel0
!
control-plane
!
line con 0
line aux 0
line vty 0 4
 login
!
End
```

Konfigurace RZ3 směrovače

```
hostname RZ3
!
```

```
boot-start-marker
boot-end-marker
!
logging message-counter syslog
!
no aaa new-model
dot11 syslog
ip source-route
!
ip cef
no ipv6 cef
!
multilink bundle-name authenticated
!
voice-card 0
!
archive
 log config
  hidekeys
!
class-map match-all match_DATA_fromRZ3
 match access-group 101
!
policy-map marking
 class match_DATA_fromRZ3
  set dscp af13
!
crypto isakmp policy 10
 encr aes 256
 authentication pre-share
 group 5
crypto isakmp key heslo address 10.0.1.2
!
crypto ipsec transform-set myset esp-aes 256 esp-sha-hmac
!
crypto ipsec profile profile
 set transform-set myset
!
interface Tunnel0
 ip address 1.1.1.2 255.255.255.252
 tunnel source 10.0.2.2
 tunnel destination 10.0.1.2
```

```
 tunnel mode ipsec ipv4
 tunnel protection ipsec profile profile
 qos-preclassify
!
interface FastEthernet0/0
 ip address 10.0.2.2 255.255.255.0
 duplex auto
 speed auto
 service-policy output marking
!
interface FastEthernet0/1
 ip address 192.168.2.1 255.255.255.0
 duplex auto
 speed auto
!
interface Serial0/1/0
 no ip address
 shutdown
 clock rate 125000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 125000
!
router ospf 1
 log-adjacency-changes
 network 10.0.2.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
!
ip forward-protocol nd
ip route 192.168.0.0 255.255.255.0 Tunnel0
no ip http server
no ip http secure-server
!
access-list 101 permit ip host 192.168.2.10 any
!
control-plane
!
ccm-manager fax protocol cisco
!
mgcp fax t38 ecm
```

```
 !
 line con 0
 line aux 0
 line vty 0 4
  login
 !
 scheduler allocate 20000 1000
end
```

Konfigurace ISP směrovače

```
[V200R005C20SPC200]
#
 drop illegal-mac alarm
#
dhcp enable
#
pki realm default
 enrollment self-signed
#
aaa
 authentication-scheme default
 authorization-scheme default
 accounting-scheme default
 domain default
 domain default_admin
 local-user admin password irreversible-cipher
%@%@}*u6!8e,8+a4YR42e^>ULT'[`Kc!Iba$>AIT"~>;W{GCT'^L%@
 local-user admin service-type http
#
firewall zone Local
 priority 64
#
interface GigabitEthernet0/0/0
 ip address 10.0.1.1 255.255.255.0
#
interface GigabitEthernet0/0/1
 ip address 10.0.0.1 255.255.255.0
#
interface GigabitEthernet0/0/2
 ip address 10.0.2.1 255.255.255.0
#
interface Cellular0/0/0
#
interface Cellular0/0/1
#
interface NULL0
#
ospf 1
 area 0.0.0.0
  network 10.0.0.0 0.0.0.255
  network 10.0.1.0 0.0.0.255
```

```
 network 10.0.2.0 0.0.0.255
#
 snmp-agent local-engineid 800007DB030819A69B6D4D
#
user-interface con 0
 authentication-mode password
 set authentication password cipher %@%@!hfe<ML7M"j]f=BgO,S4,2S3kgyk-
+6)9~L6p24+EBO,2S6,%@%@
user-interface vty 0 4
#
wlan ac
#
voice
 #
 diagnose
#
return
```

Příloha E

Implementace nástroje omezování provozu

Konfigurace RZ1 směrovače

```
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RZ1
!
boot-start-marker
boot-end-marker
!
no aaa new-model
bsd-client server url https://cloudsso.cisco.com/as/token.oauth2
!
ip cef
no ipv6 cef
!
multilink bundle-name authenticated
!
cts logging verbose
!
license udi pid CISCO2901/K9 sn FCZ1937C19J
!
redundancy
!
class-map match-all match_VOIP
 match access-group 111
class-map match-all match_DATA
 match access-group 110
 match not dscp af43
class-map match-all match_SSH
 match protocol ssh
 match access-group 110
class-map match-all match_SSH_by_DSCP
 match dscp af43
!
policy-map rz1_toLocal_input
```

```
 class match_SSH
  set dscp af43
 class class-default
  police cir 80000000
   conform-action transmit
   exceed-action drop
policy-map marking
 class match_DATA
  set dscp af11
 class match_VOIP
  set dscp ef
!
crypto isakmp policy 10
 encr aes 256
 authentication pre-share
 group 5
crypto isakmp key heslo address 10.0.0.2
crypto isakmp key heslo address 10.0.2.2
!
!
crypto ipsec transform-set myset esp-aes 256 esp-sha-hmac
 mode tunnel
crypto ipsec transform-set myset1 esp-aes 256 esp-sha-hmac
 mode transport
!
crypto map ipsec_map 10 ipsec-isakmp
 set peer 10.0.2.2
 set transform-set myset1
 match address 100
crypto map ipsec_map 20 ipsec-isakmp
 set peer 10.0.0.2
 set transform-set myset
 match address 101
 qos pre-classify
!
interface Tunnel1
 ip address 1.1.1.1 255.255.255.252
 ip mtu 1440
 ip nbar protocol-discovery
 qos pre-classify
 tunnel source GigabitEthernet0/0
 tunnel destination 10.0.2.2
```

```
 crypto map ipsec_map
!
interface Embedded-Service-Engine0/0
 no ip address
 shutdown
!
interface GigabitEthernet0/0
 ip address 10.0.1.2 255.255.255.0
 duplex auto
 speed auto
 crypto map ipsec_map
 service-policy output marking
!
interface GigabitEthernet0/1
 ip address 192.168.0.1 255.255.255.0
 ip nbar protocol-discovery
 duplex auto
 speed auto
 service-policy input rz1_toLocal_input
!
interface Serial0/1/0
 no ip address
 shutdown
 clock rate 2000000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 2000000
!
router ospf 1
 network 10.0.1.0 0.0.0.255 area 0
 network 192.168.0.0 0.0.0.255 area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 10.0.1.1
ip route 192.168.2.0 255.255.255.0 1.1.1.2
!
```

```
 access-list 100 permit gre host 10.0.1.2 host 10.0.2.2
 access-list 101 permit ip 192.168.0.0 0.0.0.255 192.168.1.0 0.0.0.255
 access-list 110 permit ip host 192.168.0.10 any
 access-list 111 permit ip host 192.168.0.20 any
 !
 control-plane
 !
 line con 0
 line aux 0
 line 2
  no activation-character
  no exec
  transport preferred none
  transport output pad telnet rlogin lapb-ta mop udptn v120 ssh
  stopbits 1
 line vty 0 4
  login
  transport input none
 !
 scheduler allocate 20000 1000
 !
End
```

Konfigurace RZ2 směrovače

```
[V200R003C00SPC200]
#
 snmp-agent local-engineid 800007DB030819A69A8275
 snmp-agent
#
cwmp
 cwmp cpe connect retry 0
#
 http timeout 3
#
 drop illegal-mac alarm
#
dhcp enable
#
undo dhcp server bootp
#
acl number 3000
 rule 5 permit ip source 192.168.1.0 0.0.0.255 destination 192.168.0.0 0.0.0.255
acl number 3001
 rule 5 permit ip source 192.168.1.0 0.0.0.255 destination 192.168.2.0 0.0.0.255
acl number 3021
 rule 5 permit ip source 192.168.1.10 0
acl number 3022
 rule 5 permit ip source 192.168.1.20 0
#
ipsec proposal tran1
 esp authentication-algorithm sha1
 esp encryption-algorithm aes-256
#
ike proposal 1
 encryption-algorithm aes-cbc-256
 dh group5
#
ike peer cisco v1
 pre-shared-key simple heslo
 ike-proposal 1
 remote-address 10.0.2.2
ike peer cisco2 v1
 pre-shared-key simple heslo
 ike-proposal 1
 remote-address 10.0.1.2
```

```
#
ipsec policy map1 1 isakmp
 security acl 3001
 ike-peer cisco
 proposal tran1
 qos pre-classify
ipsec policy map1 2 isakmp
 security acl 3000
 ike-peer cisco2
 proposal tran1
 qos pre-classify
#
traffic classifier match_any operator or
 if-match any
traffic classifier match_DATA operator or
 if-match acl 3021
traffic classifier match_VOIP operator or
 if-match acl 3022
#
traffic behavior police
 car cir 80000 green pass yellow discard red discard
 statistic enable
traffic behavior beh_VOIP
 remark dscp ef
 statistic enable
traffic behavior beh_DATA
 remark dscp af13
 statistic enable
#
traffic policy police
 classifier match_any behavior police
traffic policy marking
 classifier match_DATA behavior beh_DATA
 classifier match_VOIP behavior beh_VOIP
#
aaa
 authentication-scheme default
 authorization-scheme default
 accounting-scheme default
 domain default
 domain default_admin
 local-user admin password cipher %$%$=i~>Xp&aY+*2cEVcS-A23Uwe%$%$
```

```
 local-user admin service-type http
#
firewall zone Local
 priority 128
#
interface GigabitEthernet0/0/0
 ip address 10.0.0.2 255.255.255.0
 traffic-policy marking outbound
 ipsec policy map1
 mtu 1445
#
interface GigabitEthernet0/0/1
 ip address 192.168.1.1 255.255.255.0
 traffic-policy police inbound
#
interface GigabitEthernet0/0/2
#
interface Cellular0/0/0
 link-protocol ppp
#
interface Cellular0/0/1
 link-protocol ppp
#
interface NULL0
#
ospf 1
 area 0.0.0.0
  network 10.0.0.0 0.0.0.255
  network 192.168.1.0 0.0.0.255
#
user-interface con 0
 authentication-mode password
 set authentication password cipher
%$%$Is3>#""U3)`w':%"+:z),2f\0gpi>q[R,&GX.xR7i%T%2f_,%$%$
user-interface vty 0 4
#
wlan ac
#
voice
 #
 diagnose
#
```

Return

Konfigurace RZ3 směrovače

```
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RZ3
!
boot-start-marker
boot-end-marker
!
no aaa new-model
bsd-client server url https://cloudsso.cisco.com/as/token.oauth2
!
ip cef
no ipv6 cef
!
multilink bundle-name authenticated
!
cts logging verbose
!
license udi pid CISCO2901/K9 sn FCZ1937C19F
!
redundancy
!
class-map match-all match_VOIP
 match access-group 110
class-map match-all match_SSH_response
 match protocol ssh
class-map match-all match_DATA
 match access-group 111
 match not dscp af42
class-map match-all match_SSH_response_DSCP
 match dscp af42
!
policy-map marking
 class match_DATA
  set dscp af13
 class match_VOIP
  set dscp ef
 class match_SSH_response
  set dscp af42
```

```
policy-map police
 class class-default
  police cir 80000000 bc 85000000 conform-action transmit  exceed-action drop
   service-policy marking
!
crypto isakmp policy 10
 encr aes 256
 authentication pre-share
 group 5
crypto isakmp key heslo address 10.0.0.2
crypto isakmp key heslo address 10.0.1.2
!
crypto ipsec transform-set myset esp-aes 256 esp-sha-hmac
 mode tunnel
crypto ipsec transform-set myset1 esp-aes 256 esp-sha-hmac
 mode transport
!
crypto map ipsec_map 10 ipsec-isakmp
 set peer 10.0.0.2
 set transform-set myset
 match address 100
 qos pre-classify
crypto map ipsec_map 20 ipsec-isakmp
 set peer 10.0.1.2
 set transform-set myset1
 match address 101
!
interface Tunnel1
 ip address 1.1.1.2 255.255.255.252
 ip mtu 1440
 qos pre-classify
 tunnel source GigabitEthernet0/0
 tunnel destination 10.0.1.2
 crypto map ipsec_map
!
interface Embedded-Service-Engine0/0
 no ip address
 shutdown
!
interface GigabitEthernet0/0
 ip address 10.0.2.2 255.255.255.0
 duplex auto
```

```
 speed auto
 crypto map ipsec_map
 service-policy output police
!
interface GigabitEthernet0/1
 ip address 192.168.2.1 255.255.255.0
 ip nbar protocol-discovery
 duplex auto
 speed auto
 service-policy input rz3_toLocal_input
!
interface Serial0/1/0
 no ip address
 shutdown
 clock rate 2000000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 2000000
!
router ospf 1
 network 10.0.2.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 10.0.2.1
ip route 192.168.0.0 255.255.255.0 1.1.1.1
!
access-list 100 permit ip 192.168.2.0 0.0.0.255 192.168.1.0 0.0.0.255
access-list 101 permit gre host 10.0.2.2 host 10.0.1.2
access-list 110 permit ip host 192.168.2.20 any
access-list 111 permit ip host 192.168.2.10 any
!
control-plane
!
line con 0
line aux 0
```

```
 line 2
  no activation-character
  no exec
  transport preferred none
  transport output pad telnet rlogin lapb-ta mop udptn v120 ssh
  stopbits 1
 line vty 0 4
  login
  transport input none
 !
 scheduler allocate 20000 1000
 !
 end
```

Příloha F

Implementace nástroje tvarování provozu

Konfigurace RZ1 směrovače

```
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RZ1
!
boot-start-marker
boot-end-marker
!
!
!
no aaa new-model
bsd-client server url https://cloudsso.cisco.com/as/token.oauth2
!
!
!
!
!
!
!
!
!
!
!
!
!
ip cef
no ipv6 cef
!
multilink bundle-name authenticated
!
!
cts logging verbose
!
!
```

```
license udi pid CISCO2901/K9 sn FCZ1937C19J
!
!
!
redundancy
!
!
!
!
!
!
class-map match-all match_VOIP
 match access-group 111
class-map match-all match_DATA
 match access-group 110
 match not dscp af43
class-map match-all match_SSH
 match protocol ssh
 match access-group 110
class-map match-all match_SSH_by_DSCP
 match dscp af43
!
policy-map rz1_toLocal_input
 class match_SSH
  set dscp af43
policy-map marking
 class match_DATA
  set dscp af11
 class match_VOIP
  set dscp ef
policy-map shapingv2
 class class-default
  shape average 80000000
   service-policy marking
!
crypto isakmp policy 10
 encr aes 256
 authentication pre-share
 group 5
crypto isakmp key heslo address 10.0.0.2
crypto isakmp key heslo address 10.0.2.2
!
```

```
crypto ipsec transform-set myset esp-aes 256 esp-sha-hmac
 mode tunnel
crypto ipsec transform-set myset1 esp-aes 256 esp-sha-hmac
 mode transport
!
crypto map ipsec_map 10 ipsec-isakmp
 set peer 10.0.2.2
 set transform-set myset1
 match address 100
crypto map ipsec_map 20 ipsec-isakmp
 set peer 10.0.0.2
 set transform-set myset
 match address 101
 qos pre-classify
!
interface Tunnel1
 ip address 1.1.1.1 255.255.255.252
 ip mtu 1440
 ip nbar protocol-discovery
 qos pre-classify
 tunnel source GigabitEthernet0/0
 tunnel destination 10.0.2.2
 crypto map ipsec_map
!
interface Embedded-Service-Engine0/0
 no ip address
 shutdown
!
interface GigabitEthernet0/0
 ip address 10.0.1.2 255.255.255.0
 duplex auto
 speed auto
 crypto map ipsec_map
 service-policy output shapingv2
!
interface GigabitEthernet0/1
 ip address 192.168.0.1 255.255.255.0
 ip nbar protocol-discovery
 duplex auto
 speed auto
 service-policy input rz1_toLocal_input
!
```

```
interface Serial0/1/0
 no ip address
 shutdown
 clock rate 2000000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 2000000
!
router ospf 1
 network 10.0.1.0 0.0.0.255 area 0
 network 192.168.0.0 0.0.0.255 area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 10.0.1.1
ip route 192.168.2.0 255.255.255.0 1.1.1.2
!
access-list 100 permit gre host 10.0.1.2 host 10.0.2.2
access-list 101 permit ip 192.168.0.0 0.0.0.255 192.168.1.0 0.0.0.255
access-list 110 permit ip host 192.168.0.10 any
access-list 111 permit ip host 192.168.0.20 any
!
control-plane
!
line con 0
line aux 0
line 2
 no activation-character
 no exec
 transport preferred none
 transport output pad telnet rlogin lapb-ta mop udptn v120 ssh
 stopbits 1
line vty 0 4
 login
 transport input none
!
scheduler allocate 20000 1000
```

```
    !
end
```

Konfigurace RZ2 směrovače

```
[V200R003C00SPC200]
#
 snmp-agent local-engineid 800007DB030819A69A8275
 snmp-agent
#
cwmp
 cwmp cpe connect retry 0
#
 http timeout 3
#
 drop illegal-mac alarm
#
dhcp enable
#
undo dhcp server bootp
#
acl number 3000
 rule 5 permit ip source 192.168.1.0 0.0.0.255 destination 192.168.0.0 0.0.0.255
acl number 3001
 rule 5 permit ip source 192.168.1.0 0.0.0.255 destination 192.168.2.0 0.0.0.255
acl number 3021
 rule 5 permit ip source 192.168.1.10 0
acl number 3022
 rule 5 permit ip source 192.168.1.20 0
#
ipsec proposal tran1
 esp authentication-algorithm sha1
 esp encryption-algorithm aes-256
#
ike proposal 1
 encryption-algorithm aes-cbc-256
 dh group5
#
ike peer cisco v1
 pre-shared-key simple heslo
 ike-proposal 1
 remote-address 10.0.2.2
ike peer cisco2 v1
 pre-shared-key simple heslo
 ike-proposal 1
```

```
 remote-address 10.0.1.2
#
ipsec policy map1 1 isakmp
 security acl 3001
 ike-peer cisco
 proposal tran1
 qos pre-classify
ipsec policy map1 2 isakmp
 security acl 3000
 ike-peer cisco2
 proposal tran1
 qos pre-classify
#
traffic classifier match_any operator or
 if-match any
traffic classifier match_DATA operator or
 if-match acl 3021
traffic classifier match_VOIP operator or
 if-match acl 3022
#
traffic behavior beh_VOIP
 remark dscp ef
 statistic enable
traffic behavior beh_DATA
 remark dscp af13
 statistic enable
#
traffic policy marking
 classifier match_DATA behavior beh_DATA
 classifier match_VOIP behavior beh_VOIP
#
aaa
 authentication-scheme default
 authorization-scheme default
 accounting-scheme default
 domain default
 domain default_admin
 local-user admin password cipher %$%$=i~>Xp&aY+*2cEVcS-A23Uwe%$%$
 local-user admin service-type http
#
firewall zone Local
 priority 128
```

```
#
interface GigabitEthernet0/0/0
 ip address 10.0.0.2 255.255.255.0
 traffic-policy marking outbound
 ipsec policy map1
 qos gts cir 80000
 mtu 1445
#
interface GigabitEthernet0/0/1
 ip address 192.168.1.1 255.255.255.0
#
interface GigabitEthernet0/0/2
#
interface Cellular0/0/0
 link-protocol ppp
#
interface Cellular0/0/1
 link-protocol ppp
#
interface NULL0
#
ospf 1
 area 0.0.0.0
  network 10.0.0.0 0.0.0.255
  network 192.168.1.0 0.0.0.255
#
user-interface con 0
 authentication-mode password
 set authentication password cipher
%$%$Is3>#""U3)`w':%"+:z),2f\0gpi>q[R,&GX.xR7i%T%2f_,%$%$
user-interface vty 0 4
#
wlan ac
#
Return
```

Konfigurace RZ3 směrovače

```
RZ3#show running-config
Building configuration...

Current configuration : 3596 bytes
!
! Last configuration change at 13:27:17 UTC Fri Apr 8 2022
!
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RZ3
!
boot-start-marker
boot-end-marker
!
!
!
no aaa new-model
bsd-client server url https://cloudsso.cisco.com/as/token.oauth2
!
!
!
!
!
!
!
!
!
!
!
!
!
ip cef
no ipv6 cef
!
multilink bundle-name authenticated
!
!
cts logging verbose
```

```
!
!
license udi pid CISCO2901/K9 sn FCZ1937C19F
!
!
!
redundancy
!
!
!
!
!
!
class-map match-all match_VOIP
 match access-group 110
class-map match-all match_SSH_response
 match protocol ssh
class-map match-all match_DATA
 match access-group 111
 match not dscp af42
class-map match-all match_SSH_response_DSCP
 match dscp af42
!
policy-map shaping_Child
 class match_DATA
  shape average 50000000
  set dscp af13
 class match_VOIP
  shape average 29000000
  set dscp ef
 class match_SSH_response_DSCP
  shape average 1000000
policy-map rz3_toLocal_input
 class match_SSH_response
  set dscp af42
policy-map shapingv2
 class class-default
  shape average 80000000
   service-policy shaping_Child
!
!
!
```

```
crypto isakmp policy 10
 encr aes 256
 authentication pre-share
 group 5
crypto isakmp key heslo address 10.0.0.2
crypto isakmp key heslo address 10.0.1.2
!
!
crypto ipsec transform-set myset esp-aes 256 esp-sha-hmac
 mode tunnel
crypto ipsec transform-set myset1 esp-aes 256 esp-sha-hmac
 mode transport
!
!
!
crypto map ipsec_map 10 ipsec-isakmp
 set peer 10.0.0.2
 set transform-set myset
 match address 100
 qos pre-classify
crypto map ipsec_map 20 ipsec-isakmp
 set peer 10.0.1.2
 set transform-set myset1
 match address 101
!
!
!
!
!
interface Tunnel1
 ip address 1.1.1.2 255.255.255.252
 qos pre-classify
 tunnel source GigabitEthernet0/0
 tunnel destination 10.0.1.2
 crypto map ipsec_map
!
interface Embedded-Service-Engine0/0
 no ip address
 shutdown
!
interface GigabitEthernet0/0
 ip address 10.0.2.2 255.255.255.0
```

```
 duplex auto
 speed auto
 crypto map ipsec_map
 service-policy output shapingv2
!
interface GigabitEthernet0/1
 ip address 192.168.2.1 255.255.255.0
 ip nbar protocol-discovery
 duplex auto
 speed auto
 service-policy input rz3_toLocal_input
!
interface Serial0/1/0
 no ip address
 shutdown
 clock rate 2000000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 2000000
!
router ospf 1
 network 10.0.2.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 10.0.2.1
ip route 192.168.0.0 255.255.255.0 1.1.1.1
!
!
!
access-list 100 permit ip 192.168.2.0 0.0.0.255 192.168.1.0 0.0.0.255
access-list 101 permit gre host 10.0.2.2 host 10.0.1.2
access-list 110 permit ip host 192.168.2.20 any
access-list 111 permit ip host 192.168.2.10 any
!
control-plane
```

```
 !
 !
 !
line con 0
line aux 0
line 2
 no activation-character
 no exec
 transport preferred none
 transport output pad telnet rlogin lapb-ta mop udptn v120 ssh
 stopbits 1
line vty 0 4
 login
 transport input none
!
scheduler allocate 20000 1000
!
end
```

Příloha G

Implementace omezování, tvarování a prioritizace provozu

Konfigurace RZ1 směrovače

```
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RZ1
!
boot-start-marker
boot-end-marker
!
no aaa new-model
bsd-client server url https://cloudsso.cisco.com/as/token.oauth2
!
ip cef
no ipv6 cef
!
multilink bundle-name authenticated
!
cts logging verbose
!
license udi pid CISCO2901/K9 sn FCZ1937C19J
!
redundancy
!
class-map match-all match_VOIP
 match access-group 111
class-map match-all match_DATA
 match access-group 110
 match not dscp af43
class-map match-all match_SSH
 match protocol ssh
 match access-group 110
class-map match-all match_SSH_by_DSCP
 match dscp af43
!
policy-map rz1_toLocal_input
```

```
 class match_SSH
  set dscp af43
policy-map CBWFQ_LLQ
 class match_DATA
  bandwidth 30000
  set dscp af11
 class match_VOIP
  priority 19000
  set dscp ef
 class match_SSH_by_DSCP
  bandwidth 1000
 class class-default
  fair-queue
policy-map shaping
 class class-default
  shape average 55000000
   service-policy CBWFQ_LLQ
!
crypto isakmp policy 10
 encr aes 256
 authentication pre-share
 group 5
crypto isakmp key heslo address 10.0.0.2
crypto isakmp key heslo address 10.0.2.2
!
!
crypto ipsec transform-set myset esp-aes 256 esp-sha-hmac
 mode tunnel
crypto ipsec transform-set myset1 esp-aes 256 esp-sha-hmac
 mode transport
!
!
!
crypto map ipsec_map 10 ipsec-isakmp
 set peer 10.0.2.2
 set transform-set myset1
 match address 100
crypto map ipsec_map 20 ipsec-isakmp
 set peer 10.0.0.2
 set transform-set myset
 match address 101
 qos pre-classify
```

```
!
!
!
!
!
interface Tunnel1
 ip address 1.1.1.1 255.255.255.252
 ip mtu 1440
 ip nbar protocol-discovery
 ip tcp adjust-mss 1360
 qos pre-classify
 tunnel source GigabitEthernet0/0
 tunnel destination 10.0.2.2
 crypto map ipsec_map
!
interface Embedded-Service-Engine0/0
 no ip address
 shutdown
!
interface GigabitEthernet0/0
 ip address 10.0.1.2 255.255.255.0
 ip tcp adjust-mss 1390
 duplex auto
 speed auto
 crypto map ipsec_map
 service-policy output shaping
!
interface GigabitEthernet0/1
 ip address 192.168.0.1 255.255.255.0
 ip nbar protocol-discovery
 duplex auto
 speed auto
 service-policy input rz1_toLocal_input
!
interface Serial0/1/0
 no ip address
 shutdown
 clock rate 2000000
!
interface Serial0/1/1
 no ip address
 shutdown
```

```
 clock rate 2000000
!
router ospf 1
 network 10.0.1.0 0.0.0.255 area 0
 network 192.168.0.0 0.0.0.255 area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 10.0.1.1
ip route 192.168.2.0 255.255.255.0 1.1.1.2
!
!
!
access-list 100 permit gre host 10.0.1.2 host 10.0.2.2
access-list 101 permit ip 192.168.0.0 0.0.0.255 192.168.1.0 0.0.0.255
access-list 110 permit ip host 192.168.0.10 any
access-list 111 permit ip host 192.168.0.20 any
!
control-plane
!
!
!
line con 0
line aux 0
line 2
 no activation-character
 no exec
 transport preferred none
 transport output pad telnet rlogin lapb-ta mop udptn v120 ssh
 stopbits 1
line vty 0 4
 login
 transport input none
!
scheduler allocate 20000 1000
!
End
```

Konfigurace RZ2 směrovače

```
[V200R003C00SPC200]
#
 snmp-agent local-engineid 800007DB030819A69A8275
 snmp-agent
#
cwmp
 cwmp cpe connect retry 0
#
 http timeout 3
#
 drop illegal-mac alarm
#
dhcp enable
#
undo dhcp server bootp
#
acl number 3000
 rule 5 permit ip source 192.168.1.0 0.0.0.255 destination 192.168.0.0 0.0.0.255
acl number 3001
 rule 5 permit ip source 192.168.1.0 0.0.0.255 destination 192.168.2.0 0.0.0.255
acl number 3021
 rule 5 permit ip source 192.168.1.10 0
acl number 3022
 rule 5 permit ip source 192.168.1.20 0
#
ipsec proposal tran1
 esp authentication-algorithm sha1
 esp encryption-algorithm aes-256
#
ike proposal 1
 encryption-algorithm aes-cbc-256
 dh group5
#
ike peer cisco v1
 pre-shared-key simple heslo
 ike-proposal 1
 remote-address 10.0.2.2
ike peer cisco2 v1
 pre-shared-key simple heslo
 ike-proposal 1
 remote-address 10.0.1.2
```

```
#
ipsec policy map1 1 isakmp
 security acl 3001
 ike-peer cisco
 proposal tran1
 qos pre-classify
ipsec policy map1 2 isakmp
 security acl 3000
 ike-peer cisco2
 proposal tran1
 qos pre-classify
#
traffic classifier match_any operator or
 if-match any
traffic classifier match_DATA operator or
 if-match acl 3021
traffic classifier match_VOIP operator or
 if-match acl 3022
#
traffic behavior DATA_CBWFQ
 queue af bandwidth 25000
 statistic enable
 remark dscp af13
traffic behavior VOIP_LLQ
 queue llq bandwidth 25000
 statistic enable
 remark dscp ef
#
traffic policy rz2_toNet_output
 classifier match_DATA behavior DATA_CBWFQ
 classifier match_VOIP behavior VOIP_LLQ
#
aaa
 authentication-scheme default
 authorization-scheme default
 accounting-scheme default
 domain default
 domain default_admin
 local-user admin password cipher %$%$=i~>Xp&aY+*2cEVcS-A23Uwe%$%$
 local-user admin service-type http
#
firewall zone Local
```

```
 priority 128
#
interface GigabitEthernet0/0/0
 ip address 10.0.0.2 255.255.255.0
 traffic-policy rz2_toNet_output outbound
 ipsec policy map1
 qos gts cir 50000
#
interface GigabitEthernet0/0/1
 ip address 192.168.1.1 255.255.255.0
#
interface GigabitEthernet0/0/2
#
interface Cellular0/0/0
 link-protocol ppp
#
interface Cellular0/0/1
 link-protocol ppp
#
interface NULL0
#
ospf 1
 area 0.0.0.0
  network 10.0.0.0 0.0.0.255
  network 192.168.1.0 0.0.0.255
#
user-interface con 0
 authentication-mode password
 set authentication password cipher
%$%$Is3>#""U3)`w':%"+:z),2f\0gpi>q[R,&GX.xR7i%T%2f_,%$%$
user-interface vty 0 4
#
wlan ac
#
voice
 #
 diagnose
#
Return
```

Konfigurace RZ3 směrovače

```
version 15.5
service timestamps debug datetime msec
service timestamps log datetime msec
no service password-encryption
!
hostname RZ3
!
boot-start-marker
boot-end-marker
!
no aaa new-model
bsd-client server url https://cloudsso.cisco.com/as/token.oauth2
!
ip cef
no ipv6 cef
!
multilink bundle-name authenticated
!
cts logging verbose
!
license udi pid CISCO2901/K9 sn FCZ1937C19F
!
redundancy
!
class-map match-all match_VOIP
 match access-group 110
class-map match-all match_SSH_response
 match protocol ssh
class-map match-all match_DATA
 match access-group 111
 match not dscp af42
class-map match-all match_SSH_response_DSCP
 match dscp af42
!
policy-map rz3_toLocal_input
 class match_SSH_response
  set dscp af42
policy-map CBWFQ_LLQ
 class match_DATA
  bandwidth 25000
```

```
 set dscp af13
 class match_VOIP
  priority 24000
  set dscp ef
 class match_SSH_response_DSCP
  bandwidth 1000
 class class-default
  fair-queue
policy-map shaping
 class class-default
  shape average 50000000
   service-policy CBWFQ_LLQ
!
crypto isakmp policy 10
 encr aes 256
 authentication pre-share
 group 5
crypto isakmp key heslo address 10.0.0.2
crypto isakmp key heslo address 10.0.1.2
!
crypto ipsec transform-set myset esp-aes 256 esp-sha-hmac
 mode tunnel
crypto ipsec transform-set myset1 esp-aes 256 esp-sha-hmac
 mode transport
!
crypto map ipsec_map 10 ipsec-isakmp
 set peer 10.0.0.2
 set transform-set myset
 match address 100
 qos pre-classify
crypto map ipsec_map 20 ipsec-isakmp
 set peer 10.0.1.2
 set transform-set myset1
 match address 101
!
interface Tunnel1
 ip address 1.1.1.2 255.255.255.252
 ip mtu 1440
 ip tcp adjust-mss 1360
 qos pre-classify
 tunnel source GigabitEthernet0/0
 tunnel destination 10.0.1.2
```

```
 crypto map ipsec_map
!
interface Embedded-Service-Engine0/0
 no ip address
 shutdown
!
interface GigabitEthernet0/0
 ip address 10.0.2.2 255.255.255.0
 duplex auto
 speed auto
 crypto map ipsec_map
 service-policy output shaping
!
interface GigabitEthernet0/1
 ip address 192.168.2.1 255.255.255.0
 ip nbar protocol-discovery
 duplex auto
 speed auto
 service-policy input rz3_toLocal_input
!
interface Serial0/1/0
 no ip address
 shutdown
 clock rate 2000000
!
interface Serial0/1/1
 no ip address
 shutdown
 clock rate 2000000
!
router ospf 1
 network 10.0.2.0 0.0.0.255 area 0
 network 192.168.2.0 0.0.0.255 area 0
!
ip forward-protocol nd
!
no ip http server
no ip http secure-server
!
ip route 0.0.0.0 0.0.0.0 10.0.2.1
ip route 192.168.0.0 255.255.255.0 1.1.1.1
!
```

```
 access-list 100 permit ip 192.168.2.0 0.0.0.255 192.168.1.0 0.0.0.255
 access-list 101 permit gre host 10.0.2.2 host 10.0.1.2
 access-list 110 permit ip host 192.168.2.20 any
 access-list 111 permit ip host 192.168.2.10 any
 !
 control-plane
 !
 line con 0
 line aux 0
 line 2
  no activation-character
  no exec
  transport preferred none
  transport output pad telnet rlogin lapb-ta mop udptn v120 ssh
  stopbits 1
 line vty 0 4
  login
  transport input none
 !
 scheduler allocate 20000 1000
 !
End
```

Konfigurace ISP směrovače

```
[V200R005C20SPC200]
#
 drop illegal-mac alarm
#
dhcp enable
#
pki realm default
 enrollment self-signed
#
traffic classifier match_any operator or
 if-match any
traffic classifier match_DATA operator or
 if-match dscp af11 af13
traffic classifier match_VOIP operator or
 if-match dscp ef
traffic classifier match_SSH operator or
 if-match dscp af42 af43
#
traffic behavior police
 car cir 50000 green pass yellow discard red discard
 statistic enable
traffic behavior DATA_CBWFQ
 queue af bandwidth 50000
 statistic enable
traffic behavior VOIP_LLQ
 queue llq bandwidth 50000
 statistic enable
traffic behavior SSH_CBWFQ
 queue af bandwidth 2000
 statistic enable
#
traffic policy police
 classifier match_any behavior police
traffic policy CBWFQ_LLQ
 classifier match_DATA behavior DATA_CBWFQ
 classifier match_VOIP behavior VOIP_LLQ
 classifier match_SSH behavior SSH_CBWFQ
#
aaa
 authentication-scheme default
```

```
 authorization-scheme default
 accounting-scheme default
 domain default
 domain default_admin
 local-user admin password irreversible-cipher
%@%@}*u6!8e,8+a4YR42e^>ULT'[`Kc!Iba$>AIT"~>;W{GCT'^L%@
 local-user admin service-type http
#
firewall zone Local
 priority 64
#
interface GigabitEthernet0/0/0
 ip address 10.0.1.1 255.255.255.0
 traffic-policy police inbound
 traffic-policy CBWFQ_LLQ outbound
#
interface GigabitEthernet0/0/1
 ip address 10.0.0.1 255.255.255.0
 traffic-policy police inbound
 traffic-policy CBWFQ_LLQ outbound
#
interface GigabitEthernet0/0/2
 ip address 10.0.2.1 255.255.255.0
 traffic-policy police inbound
 traffic-policy CBWFQ_LLQ outbound
#
interface Cellular0/0/0
#
interface Cellular0/0/1
#
interface NULL0
#
ospf 1
 area 0.0.0.0
  network 10.0.0.0 0.0.0.255
  network 10.0.1.0 0.0.0.255
  network 10.0.2.0 0.0.0.255
#
 snmp-agent local-engineid 800007DB030819A69B6D4D
#
user-interface con 0
 authentication-mode password
```

```
    set authentication password cipher %@%@!hfe<ML7M"j]f=BgO,S4,2S3kgyk-
+6)9~L6p24+EBO,2S6,%@%@
    user-interface vty 0 4
    #
    wlan ac
    #
    voice
     #
     diagnose
    #
return
```