VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION

ÚSTAV TELEKOMUNIKACÍ

DEPARTMENT OF TELECOMMUNICATIONS

SÍŤOVÁ INFRASTRUKTURA PRO VÝKONOVÉ TESTOVÁNÍ KVANTOVĚ ODOLNÝCH ŠIFRÁTORŮ

NETWORK INFRASTRUCTURE FOR PERFORMANCE TESTING OF QUANTUM-SAFE ENCRYPTORS

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE — Adam Fridrich

AUTHOR

VEDOUCÍ PRÁCE — prof. Ing. Jan Hajný, Ph.D.

SUPERVISOR

BRNO 2025

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Bakalářská práce

bakalářský studijní program **Informační bezpečnost**

Ústav telekomunikací

Student: Adam Fridrich ***ID:*** 247536

Ročník: 3 ***Akademický rok:*** 2024/25

NÁZEV TÉMATU:

Síťová infrastruktura pro výkonové testování kvantově odolných šifrátorů

POKYNY PRO VYPRACOVÁNÍ:

Téma je zaměřeno na sestavení síťové infrastruktury pro testování postkvantových šifrátorů na vysokých rychlostech až 100 Gbps. K dispozici budou vysokorychlostní šifrátory na platformě FPGA (nicméně není třeba znalost této platformy), komerční šifrátory a vysokorychlostní síťové prvky. V rámci práce student sestaví vhodnou síť pro testování a provede základní testy zaměřené zejména na přenosovou rychlost a odezvu. Součástí práce bude vhodná reprezentace výsledků a protokol z měření. Dále se student zaměří na integraci s dalšími typy zařízení (např. embedded prvky či aktivními síťovými prvky) a na pokročilejší testy s reálnými aplikacemi (web, VoIP, video, atp.).

DOPORUČENÁ LITERATURA:

[1] HAVLÍN, Jan. Šifrátor síťového provozu na platformě OS Linux. Bakalářská práce. Brno: VUT v Brně, 2024.

[2] CÍBIK, P.; DOBIÁŠ, P.; RICCI, S.; HAJNÝ, J.; MALINA, L.; JEDLIČKA, P.; SMÉKAL, D. Pushing AES-256-GCM to Limits: Design, Implementation and Real FPGA Tests. In Lecture Notes in Computer Science - Applied Cryptography and Network Security Workshops. Lecture Notes in Computer Science. 2024. s. 303-318. ISBN: 978-3-031-61486-6. ISSN: 0302-9743.

Termín zadání: 10.2.2025 ***Termín odevzdání:*** 3.6.2025

Vedoucí práce: prof. Ing. Jan Hajný, Ph.D.

prof. Ing. Jan Hajný, Ph.D.
předseda rady studijního programu

UPOZORNĚNÍ:

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

ABSTRAKT

Tato bakalářská práce je zaměřena na sestavení síťové infrastruktury pro testování kvantově odolných šifrátorů. V rámci práce byla sestavena síť pomocí virtualizovaného prostředí a dále byly realizovány výkonové testy pro rychlosti až 100 Gbit/s. Testy jsou zaměřeny na propustnost a latenci nešifrovaného a následně šifrovaného provozu. Pro šifrování síťového provozu byly využity šifrátory na platformě FPGA a firewally Juniper SRX4200. Schopnosti šifrátorů byly dále testovány a ověřovány v reálných aplikacích.

KLÍČOVÁ SLOVA

Šifrátor, FPGA, Juniper, výkonové testování, TCP, UDP, virtualizace, Linux

ABSTRACT

This bachelor thesis focuses on building a network infrastructure for testing quantum-safe encryptors. The thesis involved the construction of a network using a virtualized environment and furthermore, performance tests for speeds up to 100 Gbit/s. The tests focus on throughput and latency of unencrypted and then encrypted traffic. Encryption of network traffic was achieved with FPGA-based encryptors and Juniper SRX4200 firewalls. Capabilities of the encryptors were further tested and verified in real-world applications.

KEYWORDS

Encryptor, FPGA, Juniper, Performance testing, TCP, UDP, Virtualization, Linux

Vysázeno pomocí balíčku `thesis` verze 4.11; `https://latex.fekt.vut.cz/`

FRIDRICH, Adam. *Síťová infrastruktura pro výkonové testování kvantově odolných šifrátorů*. Bakalářská práce. Brno: Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací, 2024. Vedoucí práce: prof. Ing. Jan Hajný, PhD.

Prohlášení autora o původnosti díla

Jméno a příjmení autora:	Adam Fridrich
VUT ID autora:	247536
Typ práce:	Bakalářská práce
Akademický rok:	2024/25
Téma závěrečné práce:	Síťová infrastruktura pro výkonové testování kvantově odolných šifrátorů

Prohlašuji, že svou závěrečnou práci jsem vypracoval samostatně pod vedením vedoucí/ho závěrečné práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou všechny citovány v práci a uvedeny v seznamu literatury na konci práce.

Jako autor uvedené závěrečné práce dále prohlašuji, že v souvislosti s vytvořením této závěrečné práce jsem neporušil autorská práva třetích osob, zejména jsem nezasáhl nedovoleným způsobem do cizích autorských práv osobnostních a/nebo majetkových a jsem si plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon), ve znění pozdějších předpisů, včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č. 40/2009 Sb.

Brno ....

podpis autora*

*Autor podepisuje pouze v tištěné verzi.

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce prof. Ing. Janu Hajnému, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

The work was supported by the European Union's Horizon Europe project #101087529 CHESS.

Views and opinions expressed are however those of the author(s) only and do not necessarily reflect those of the European Union or European Research Executive Agency. Neither the European Union nor the granting authority can be held responsible for them.

Obsah

Úvod **11**

1 Zátěžové testování IP sítí **12**
1.1 Používané protokoly . . . 12
1.1.1 Transmission Control Protocol . . . 12
1.1.2 User Datagram Protocol . . . 12
1.1.3 Internet Protocol . . . 13
1.2 Problematika analýzy a testování IP sítí . . . 13
1.2.1 Analýza infrastruktury a síťového provozu . . . 14
1.2.2 Zátěžové testy . . . 14

2 Praktická implementace testbedu **15**
2.1 Praktické řešení sítě . . . 15
2.2 Metodika a nástroje testování . . . 15
2.3 Úvodní měření . . . 16
2.4 Optimalizace . . . 17
2.5 Analýza důsledků optimalizace . . . 19
2.6 Analýza síťového provozu . . . 20

3 Realizace zátěžových testů **22**
3.1 Provoz bez šifrování . . . 22
3.2 Provoz s šifrováním . . . 23
3.2.1 Platforma FPGA . . . 23
3.2.2 Firewally Juniper . . . 27

4 Reálné aplikace **34**
4.1 Webová komunikace . . . 34
4.1.1 Python HTTP . . . 34
4.1.2 NGINX . . . 36
4.2 Video Streaming . . . 37
4.3 Vysokorychlostní přepínač . . . 37

Závěr **39**

Literatura **41**

Seznam symbolů a zkratek **45**

A Protokol z měření **47**

Seznam obrázků

2.1 Graf porovnání hodnot RTT před a po optimalizaci. 19
2.2 Graf porovnání propustností TCP a UDP provozu před a po optimalizaci. 19
2.3 Ukázka testovacího TCP provozu s maximálním vytížením – MTU 1500 B. 20
2.4 Ukázka testovacího UDP provozu s maximálním vytížením – MTU 1500 B. 21
3.1 Vizualizace topologie pro testování s šifrátory na platformě FPGA. . 24
3.2 Vizualizace topologie pro testování s šifrátory Juniper. 27
3.3 Graf hodnot propustností pro různé parametry VPN – Juniper SRX4200. 33
4.1 Topologie pro testování s šifrátory na platformě FPGA + Switch. . . 37

Seznam tabulek

3.1 Výsledky testů propustnosti pro nešifrovaný TCP provoz. 22
3.2 Výsledky testů propustnosti pro nešifrovaný UDP provoz. 23
3.3 Výsledky testů propustnosti, TCP – FPGA UltraScale+ (Mango). . . 24
3.4 Výsledky testů propustnosti, UDP – FPGA UltraScale+ (Mango). . . 25
3.5 Výsledky testů propustnosti, TCP – FPGA Agilex (Intel). 25
3.6 Výsledky testů propustnosti, UDP – FPGA Agilex (Intel). 26
3.7 Výsledky testů propustnosti, TCP – FPGA Agilex (Intel), QKD. . . . 26
3.8 Výsledky testů propustnosti, UDP – FPGA Agilex (Intel), QKD. . . . 26
3.9 Výsledky dlouhodobě šifrovaného provozu – FPGA Agilex (Intel). . . 26
3.10 Výsledky testů propustnosti 10G linky pro nešifrovaný provoz. 27
3.11 Výsledky testů propustnosti VPN komunikace (Juniper SRX4200), Šifrovací algoritmus: DES-CBC, Hashovací algoritmus: HMAC-SHA-256-128. 29
3.12 Výsledky testů propustnosti VPN komunikace (Juniper SRX4200), Šifrovací algoritmus: 3DES-CBC, Hashovací algoritmus: HMAC-SHA-256-128. 29
3.13 Výsledky testů propustnosti VPN komunikace (Juniper SRX4200), Šifrovací algoritmus: AES-256-CBC, Hashovací algoritmus: HMAC-SHA-256-128. 30
3.14 Výsledky testů propustnosti VPN komunikace (Juniper SRX4200), Šifrovací algoritmus: AES-256-CBC, Hashovací algoritmus: SHA-512. 30
3.15 Výsledky testů propustnosti VPN komunikace (Juniper SRX4200), Šifrovací algoritmus: AES-256-CBC, Hashovací algoritmus: SHA-384. 31
3.16 Výsledky testů propustnosti VPN komunikace (Juniper SRX4200), Šifrovací algoritmus: AES-256-GCM. 31
3.17 Výsledky testů propustnosti VPN komunikace (Juniper SRX4200), Šifrovací algoritmus: AES-128-GCM. 32
4.1 Rychlosti stahování souborů z webserveru – Python HTTP. 36
4.2 Rychlosti stahování souborů z webserveru – NGINX. 36
4.3 Provoz přes switch, TCP – FPGA Agilex (Intel). 38
4.4 Provoz přes switch, UDP – FPGA Agilex (Intel). 38

Úvod

Bezpečnost současné kryptografie je založena na předpokladu, že určité matematické problémy nelze řešit v polynomiálním čase. Pro kvantové počítače je však výpočetní náročnost těchto problémů výrazně nižší. Přítomnost dostatečně výkonných kvantových počítačů by tak mohla bezpečnost zásadně narušit. Problematika testování výkonnosti představuje důležitou součást implementace různých řešení zahrnujících postkvantovou kryptografii. Tímto řešením mohou být například šifrátory.

Tato práce se zabývá problematikou výkonového testování kvantově odolných šifrátorů. Cílem bylo navrhnout a sestavit síťovou infrastrukturu a následně realizovat výkonové testy pro dva různé typy postkvantových šifrátorů. Text práce je rozdělen do celkem čtyř kapitol, přičemž první kapitola je zaměřena na teorii představující protokoly a pojmy využívané později v práci. Další tři kapitoly popisují vlastní řešení, tedy praktickou část této práce. Zde je představena realizace testbedu, průběh a výsledky samotného testování, a využití daných šifrátorů v reálných aplikacích.

Pro realizaci sítě bylo využito virtualizované prostředí, v němž byly vytvořeny dvě virtuální stanice. Tyto stanice představovaly koncové body, mezi nimiž probíhala síťová komunikace. Po sestavení sítě proběhly úvodní testy a následně byly podstoupeny optimalizační kroky, které umožnily dosažení lepších výsledků. Samotné testy byly zaměřeny zejména na propustnost a latenci, a byly realizovány pro rychlosti až 100 Gbit/s. Nejprve byly realizovány testy propustnosti TCP a UDP provozu bez šifrování. Hodnoty nešifrovaného provozu sloužily jako reference pro analýzu hodnot šifrovaného provozu. Testy šifrovaného provozu byly zhotoveny za užití šifrátorů na platformě FPGA a komerčních šifrátorů na prvcích Juniper. Tyto testy byly zhotoveny pro uměle generovaný síťový provoz, k čemuž byl využit nástroj iPerf.

Výkon šifrátorů byl také ověřen v rámci testů se skutečným síťovým provozem, tedy v reálných aplikacích. Tyto testy zahrnovaly webovou komunikaci – nalezení maximálních rychlostí pro stahování souborů a živý přenos videa, jako ověření provozu v reálném čase. V rámci testování s dalšími síťovými prvky byl využit vysokorychlostní přepínač Mikrotik, který byl přidán do 100 Gbit/s linky. Testy provozu přes přepínač byly opět v podobě testů propustnosti, kdy bylo zkoumáno, zda bude provoz limitován samotným přepínačem.

Výstupem práce je sestavená síťová infrastruktura a výsledné hodnoty a závěry plynoucí z jednotlivých testů. Součástí práce je také protokol z měření v podobě přílohy. Protokol shrnuje výsledky získané v práci a blíže popisuje průběh testování.

1 Zátěžové testování IP sítí

Tato kapitola je zaměřena na teorii popisující problematiku testování počítačových sítí a generování síťového provozu. Představuje protokoly TCP, UDP a IP pomocí kterých probíhalo samotné testování. Poté jsou představeny oblasti analýzy infrastruktury při její implementaci, analýza síťového provozu a zátěžové testování. Cílem kapitoly je představit oblasti a objasnit pojmy vyskytující se později v práci.

1.1 Používané protokoly

1.1.1 Transmission Control Protocol

Transmission Control Protocol (TCP) je protokol transportní vrstvy využívaný pro spolehlivý přenos dat po síti. Je nejpoužívanějším protokolem této vrstvy. Spolehlivost TCP je zajištěna tím, že protokol dokáže rozpoznat a opravit chyby, které vznikly při přenosu. Jedná se o tzv. stavový protokol – komunikující strany spolu navazují a ukončují spojení na základě vzájemné domluvy. TCP také disponuje mechanismy pro řízení provozu, kontrolu přetížení a již zmíněnou detekci chybovosti[1]. V rámci TCP jsou přenášená data ve formě tzv. segmentů, které jsou sekvenčně číslovány. Doručení segmentu příjemci je zakončeno odesláním potvrzovací ACK zprávy zpět odesílateli. Frekvenci zasílaných ACK zpráv lze regulovat pomocí velikosti TCP okna, které udává velikost přenesených dat, než je požadována ACK zpráva[2]. Informace o velikosti TCP okna je součástí záhlaví a základní velikost je 16 kB. Nastavená velikost TCP okna může mít značný vliv na maximální možnou propustnost[3]. Celková velikost záhlaví TCP segmentu je v rozmezí 20–60 B. Protokol je optimalizován pro přenosy velkých objemů dat a důraz byl kladen také na výkon[4]. Protokol TCP tak představuje častou volbu pro testování sítí[5].

1.1.2 User Datagram Protocol

User Datagram Protocol (UDP) je protokol transportní vrstvy využívaný pro přenos dat v reálném čase. Jedná se o bezstavový protokol, jež je uzpůsoben pro komunikaci s nízkou latencí. Na spolehlivost přenosu zde není kladen důraz. Oproti TCP nedisponuje mechanismy pro detekci chyb nebo znovuzaslání ztracených datagramů. Protokol tedy odesílá data bez záruky doručení[6]. Díky nižší úrovni režie komunikace je přenos pomocí UDP rychlejší a vhodnější pro komunikace jako jsou audiohovory, videohovory nebo DNS lookup[7]. V záhlaví UDP datagramu lze nalézt informace o zdrojovém a cílovém portu, kontrolním součtu a délce datagramu. Celková velikost záhlaví je 8 B, je tak značně menší v porovnání s TCP[6]. Realizace zátěžových

testů s využitím UDP provozu může přinést užitečné informace o maximálních možných rychlostech bezstavové komunikace, kterou lze na síti uskutečnit. Zároveň se těmito testy může identifikovat míra ztrátovosti, protože UDP ztracená data znovu nezasílá. Opakované zasílání dat mohou zajišťovat vyšší vrstvy síťového modelu. Při testech je také možné zkoumat chování sítě při stavu přetížení, jelikož sám protokol není schopen tuto skutečnost nijak detekovat. Zkrátka výsledky zátěžových testů UDP provozu vypovídají o výkonnosti infrastruktury, kdy nedochází k interferenci transportním protokolem.

1.1.3 Internet Protocol

Internet Protocol (IP) je protokol síťové vrstvy patřící do skupiny TCP/IP. Zajišťuje adresaci, směrování a dělení či skládání přenášených dat do jednotlivých segmentů – paketů. Protokol tedy zajišťuje konektivitu síťových prvků na 3. vrstvě na základě IP adres. Implementace protokolu je prerekvizitou pro uskutečnění komunikace na vyšších vrstvách[8]. Na úrovni protokolu může také docházet k fragmentaci – dělení jednotlivých paketů na menší celky. K fragmentaci může docházet např. pokud zařízení nedokáže zpracovat paket o dané velikosti nebo nastavená velikost paketu MTU (Maximum Transmission Unit) je na koncových stanicích příliš velká. Informace o fragmentaci, užívané pro tvoření menších celků a následné skládání do původní podoby, jsou přenášené v záhlaví paketu[8]. V záhlaví lze také nalézt zdrojovou a cílovou IP adresu ad. Celková velikost záhlaví je 20–60 B[9]. Pomocí protokolu lze také testovat např. latenci přenosového média. Latence může být měřena ve formě Round-Trip Time (RTT) pomocí příkazů `ping` nebo `traceroute`.

1.2 Problematika analýzy a testování IP sítí

S rostoucím objemem dat přenášených po sítích, se nároky na výkon neustále zvyšují. Síťové infrastruktury by měly být navrženy tak, aby fungovaly spolehlivě nejen při běžné zátěži, ale také za podmínek maximálního vytížení. V praxi to znamená, že by mělo být známo, jaký typ provozu bude po síti přenášen, objemy samotného provozu v různých situacích nebo způsob, jakým budou reagovat jednotlivé síťové prvky na vrcholové zatížení. Analýza infrastruktury a realizace zátěžových testů tvoří velmi důležitou součást procesu praktického řešení sítě. Díky těmto procesům lze identifikovat předpokládané stavy, tzn. správně identifikovat a interpretovat výsledky nebo případné anomálie, a předurčit tak infrastrukturu správnému fungování[10].

1.2.1 Analýza infrastruktury a síťového provozu

Úvodní analýza spojená s návrhem a implementací síťové infastruktury by měla určit, jakým způsobem by měla být síť realizována. Pomocí analýzy lze identifikovat, jaké fyzické komponenty by měly být součástí sítě – jestli je vhodné využít virtualizaci, zda budou implementovány redundantní prvky, využití balancování zátěže apod. V potaz by měly být brány aspekty, jako jsou výkon, kapacita, škálovatelnost, bezpečnost, odolnost atd. Síť by měla být realizována tak, aby splňovala požadavky návrhu a jednotlivé prvky pracovaly efektivně[11]. Po implementaci infrastruktury v praxi by mělo ideálně následovat monitorování výkonu nebo realizace zátěžových testů. Součástí těchto procesů je i analýza síťového provozu.

Na základě analýzy příchozího a odchozího síťového provozu, který prochází přes zkoumaný prvek, lze zjistit dosažený výkon sítě. Analyzují se například hodnoty propustnosti – tj. dosažené přenosové rychlosti, latence – časového rozdílu mezi odesláním a přijetím zprávy, nebo ztrátovosti paketů[12]. Hodnoty síťových parametrů získané z výsledků testování, by měly odpovídat rozmezí, které bylo určeno při analýze. Lze například stanovit prahovou hodnotu ztráty paketů na 5%. Pokud výsledná naměřená hodnota překročí tento práh, nasvědčuje tak, že v síti nastal poruchový nebo mimořádný stav, jako je např. přetížení[5].

1.2.2 Zátěžové testy

Zátěžové testy slouží k ověření výkonnosti, odolnosti a správného chování síťové infrastruktury v různých situacích. Výsledky testů mohou také odhalit nevhodná nastavení nebo existující limitující faktory, což vyžaduje následnou optimalizaci. Jedná se o běžně používanou metodu testování kritických infrastruktur, cloudových a streamovacích služeb, ale také menších sítí. Součástí testování by měla být již zmíněná analýza, která mimo jiné pomůže určit, jakým způsobem budou testy probíhat a jaké nástroje k tomu využít. Jednodušší způsob může zahrnovat např. nástroj iPerf[13]. U velkých sítí, o rychlostech přesahujících i Tb/s, je vhodné využít sofistikovanější nástroje a postupy testování[12]. Zátěžové testy lze realizovat pomocí uměle generovaného síťového provozu. Tento provoz by měl ideálně emulovat skutečný provoz, aby byly výsledné hodnoty srovnatelné s hodnotami při reálném nasazení sítě[12]. Postupy a způsoby realizace jednotlivých testů se liší v závislosti na stanoveném cíli. Zátěžové testování lze dále rozčlenit na tzv. „Performance testing“, „Stress testing“, „Capacity testing“ nebo „Peak testing“. Rozdíly mezi těmito kategoriemi jsou v různých časových délkách trvání testů a objemu či charakteru testovacího síťového provozu[14].

2 Praktická implementace testbedu

V této kapitole je prezentována praktická implementace síťové infrastruktury. Poté je představena metodika a průběh úvodního měření propustnosti pro TCP a UDP provoz a následné kroky optimalizace, které měly za cíl dosažení vyšší propustnosti. Součástí jsou i výsledky testování latence fyzické linky mezi virtuálními stroji.

2.1 Praktické řešení sítě

Pro praktické řešení byla využita virtualizace. Virtualizované prostředí bylo v podobě open-source hypervizora XCP-ng ve verzi 8.2.1 a grafické rozhraní Xen Orchestra pro správu virtuálních stanic. Pro samotné virtuální stanice byl nainstalován OS Linux v distribuci Debian 12.7. Stanice byly celkem dvě, plnící role klient-server. Každé stanici bylo přiděleno 10 jader CPU (Intel Xeon 6136 @ 3 GHz), 32 GB RAM a jeden interface 100G síťové karty Mellanox ConnectX-5 MT27800 jako virtuální rozhraní. Síťová karta má celkem dvě 100G rozhraní, která byla propojena do smyčky, takže přidělením rozhraní virtuálním stanicím byla vytvořena fyzická linka mezi virtuálními stroji o teoretické rychlosti 100 Gbit/s. V průběhu testování byl XCP-ng 8.2.1 aktualizován na verzi 8.3, což umožnilo virtuálním stanicím přímo pracovat s fyzickým rozhraním síťové karty namísto virtuálních rozhraní[15].

2.2 Metodika a nástroje testování

Testování propustnosti bylo realizováno za pomoci nástroje iPerf. Nástroj iPerf nabízí různé verze, díky kterým lze testovat jednovláknový výkon (iPerf3) nebo vícevláknový výkon (iPerf2)[16]. Při realizaci výkonových testů byl iPerf používán v obou verzích. Doba trvání jednotlivých testů byla 30 sekund. Tato délka byla zvolena kvůli dostatečné vypovídající hodnotě o aktuální dosažitelné propustnosti. To bylo ověřováno opakováním jednotlivých testů. Každý test byl opakován minimálně třikrát, čímž byla ověřena jednoznačnost hodnot. Hodnoty v tabulkách v další části uvádí průměrnou hodnotu vypočtenou z výsledných hodnot všech provedených testů. Samotné testy byly realizovány s využitím různého počtu CPU jader pro každou stanici a různé velikosti MTU. Přidělení konkrétního počtu jader procesu `iperf` při testech, které to vyžadovaly, bylo realizováno pomocí nástroje `taskset`. Ověření, zda jsou vytěžována jen přidělená jádra, bylo provedeno analýzou grafu „CPU usage“ v prostředí Xen Orchestra a také nástrojem `htop`. Pro šifrovaný provoz byly testované kategorie počtu užitých CPU jader a velikostí MTU přizpůsobovány schopnostem šifrátorů. Testbed byl umístěn v laboratoři FEKT a veškeré testování a správa virtuálních stanic probíhaly vzdáleně přes VPN a grafické rozhraní hypervizora.

Testování latence probíhalo pomocí příkazů `ping` a `traceroute`. Tyto příkazy byly využity pro měření hodnoty RTT (Round-Trip Time) na úrovni síťové vrstvy[17].

2.3 Úvodní měření

V úvodním stavu, tj. po instalaci hypervizora, virtuálních stanic klient-server a přidělení virtuálních rozhraní, byly na stanicích nastaveny IP adresy 10.10.0.1/24, 10.10.0.2/24 pro virtuální rozhraní fyzické linky. Následně byla ověřena konektivita.

V této fázi byl využit iPerf3 pro testování TCP provozu. Dvěma paralelními proudy, parametr `-P 2`, testovacího provozu dosahovala průměrná hodnota propustnosti 20 Gbit/s. Bylo zřejmé, že limitujícím faktorem byl procesor odesílatele, což šlo poznat z grafu „CPU usage“ v prostředí Xen Orchestra nebo při použití parametru `-V` (verbose) v příkazu `iperf3`. Jedno jádro CPU bylo vytíženo na maximum. Dalším postupem tedy bylo vytvoření více instancí iPerf serverů naslouchajících na různých portech a na straně klienta poté použít více příkazů pro připojení najednou. Dvěma paralelními testy se vytížení CPU (2 jádra na každé stanici) rozložilo a celková propustnost dosahovala rychlostí až 30 Gbit/s. Rychlost tuto hodnotu však nepřekročila, ani při více paralelních testech současně. Bylo pozorováno pouze další rozložení vytížení mezi jednotlivými CPU jádry. Po zjištění, že jednovláknový iPerf3 nedosáhne vyšší rychlosti, byl nainstalován vícevláknový iPerf2. Ten byl však v tomto stavu také limitován a rychlost byla maximálně 30 Gbit/s. Výsledné rychlosti testů propustnosti UDP provozu, provedeny pomocí iPerf3, dosahovaly maximálních hodnot 1,5–1,6 Gbit/s – jednojádrový výkon. Důležitým parametrem při testování UDP provozu pomocí iPerf je specifikace šířky pásma `-b 100G`. Bez specifikace šířky pásma byly výsledné hodnoty v rámci Mb/s. Bylo provedeno až pět testů současně (5 instancí serverů, 5 paralelních testů pro každý z portů), kdy iPerf3 dosahoval pořád 1,5 Gbit/s na jádro, takže výsledná rychlost byla pětinásobná, tedy 8 Gbit/s. Hodnota odpovídala i v grafu “Network throughput” v Xen Orchestra pro dané virtuální rozhraní. Testování s iPerf2 nepřineslo žádné nové výsledky ani pro UDP provoz. Změna velikosti paketu (MTU) na velikost 9000 B z původní hodnoty 1500 B pro virtuální rozhraní na rychlost neměla v tomto stavu vliv.

Součástí testování 100G linky bylo testování její latence. Průměrná hodnota RTT pro 13 dotazů příkazem `ping` byla 0,677 ms. Čas vykonání příkazu `traceroute` byl 1,536 ms. Jednalo se pouze o jeden skok, kterým byla druhá virtuální stanice.

Tyto dosažené hodnoty nedosahovaly žádaných výsledků. Výsledné rychlosti testů propustnosti se zdaleka nepřibližovaly 100 Gbit/s a hodnota RTT byla, vzhledem k charakteru zapojení sítě, tedy přímo připojené stanice, pro danou rychlost vysoká. Bylo tedy zřejmé, že je nutné podstoupit kroky pro dosažení optimálních hodnot.

2.4 Optimalizace

Prvním krokem optimalizace bylo přidání následujících parametrů do systémového konfiguračního souboru `/etc/sysctl.conf` na virtuálních stanicích:

Výpis 2.1: Parametry přidané do systémové konfigurace.

```
# Maximální velikost TCP bufferu (až 2 GB)
net.core.rmem_max = 2147483647
net.core.wmem_max = 2147483647

# Auto. ladění TCP bufferů: min. 4 kB, vý. 85 kB, max. 128 MB
net.ipv4.tcp_rmem = 4096 87380 134217728
net.ipv4.tcp_wmem = 4096 65536 134217728

# Povolení automatického zjišťování optimální MTU
net.ipv4.tcp_mtu_probing = 1

# Výchozí qdisc na 'fg' (fair queueing) - řazení paketů
net.core.default_qdisc = fg

# Maximální paměť pro optimalizaci socketů (1 MB)
net.core.optmem_max = 1048576
```

To mělo za následek možnost měnit velikost TCP okna při iPerf testování, v tomto stavu byly však následky minimální. Pozorovaná změna byla ve vytížení procesoru, kdy před změnou bylo CPU odesílatele maximálně vytížené a CPU příjemce o něco méně, zatímco po změně naopak. Tato změna byla následkem přidání parametrům „fair queueing“ a max. paměti socketů. Díky FQ se fronty paketů pro zpracování rovnoměrně rozdělí a nedochází k nerovnoměrnému zatížení CPU[18].

Dalším krokem bylo rozhodnutí, že by virtuální stanice pracovaly přímo s fyzickým rozhraním síťové karty namísto virtuálních. Zde přišla na řadu aktualizace XCP-ng na verzi 8.3 umožňující povolení PCI passthrough[15]. Povolení této funkcionality a přidělení fyzických rozhraní virtuálním stanicím se provedlo v Xen Orchestra. Jednotlivé porty fyzické síťové karty tak bylo možné přidělit virtuální stanici na základě vygenerovaného ID. Pro umožnění této funkcionality bylo nutné mít v BIOS systému nejprve povoleny parametry VT-d a IOMMU[19]. Pro ověření nastavení a přítomnosti firmware na síťové kartě byly využity příkazy `lspci`, `dmesg` a `ethtool`. V této fázi tedy virtuální stroje mohly pracovat přímo se síťovou kartou.

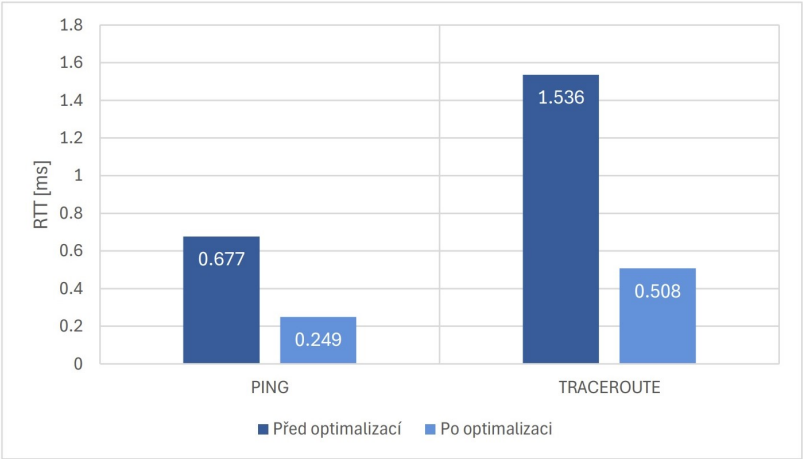
Tento krok měl zásadní dopad na výkon sítě. Průměrná hodnota RTT pro 13 dotazů `ping` byla 0,249 ms, čas `traceroute` 0,508 ms. RTT se tedy zmenšil na třetinu. Rychlost TCP provozu pro jednojádrový výkon přesáhla 30 Gbit/s oproti předešlým

20 Gbit/s. Rychlost UDP provozu byla 2,8 Gbit/s oproti předchozí 1,5 Gbit/s. Testováno opět pomocí iPerf3. Maximální dosažené hodnoty s více jádry se blížily ke 100 Gbit/s a jsou prezentovány v další kapitole.

Rozdílnost rychlostí TCP a UDP provozu závisí na více faktorech. Možnost, kdy by způsob virtualizace limitoval výkon sítě, byla z většiny eliminována povolením PCI passthrough. Systémová nastavení virtuálních stanic, včetně nastavení síťové karty, by mělo být adekvátní pro rychlost až 100 Gbit/s. Ověření správnosti nastavení, tj. že NIC (Network Interface Card) datagramy o nastavené velikosti dále nefragmentuje a komunikace probíhala dle interpretace iPerf, bylo ověřeno pomocí programu Wireshark. Důvodem, proč UDP provoz dosahoval nižších rychlostí, přestože disponuje nižší úrovní režie komunikace, bylo kvůli samotnému programu iPerf. Program iPerf nejdříve generuje co největší provoz, který se následně snaží po lince přenést. Způsob generování a následného monitorování provozu přidává novou úroveň procesního zatížení. Pro TCP provoz tento aspekt nepředstavuje výrazný problém, protože samotné TCP disponuje mechanismy pro řízení provozu a kontrolu přetížení a je optimalizováno i pro vysoké rychlosti. Protokol UDP podobné mechanismy nemá. Pozorovaný jev – maximální vytížení přidělených CPU jader pro téměř všechny testy, a přesto dosažení menších rychlostí oproti TCP, je s největší pravděpodobností způsobeno samotným protokolem UDP a funkcionalitou iPerf. UDP nedokáže kontrolovat přetížení a iPerf nedokáže efektivně generovat UDP provoz ve vysokých rychlostech. Při testování tedy docházelo k tomu, že byl procesor odesílatele „zahlcen“ a nedokázal efektivně zpracovávat a posílat generované datagramy[22]. Těchto výsledků bylo dosaženo při povolených mechanismech GSO (Generic Segmentation Offload) a GRO (Generic Recieve Offload), které by měly pomoci snížit zatížení CPU i při UDP provozu. Pro kontrolu povolených typů „Offload“ mechanismů byl využit nástroj `ethtool`. Rychlost testovaného UDP provozu předčila TCP až při MTU na 8000 B s využití všech jader, to kvůli nižší frekvenci zpracovávání jednotlivých datagramů. Pro generování UDP provozu a testy propustnosti byl využit také nástroj `nuttcp` jako alternativa k iPerf s cílem dosažení vyšších rychlostí. Užitím tohoto nástroje však byla dosažena stejná maximální propustnost (UDP – 25 Gbit/s při MTU 1500 B) opět v důsledku limitace procesorem. Lze tedy očekávat, že rychlosti UDP provozu by byly vyšší, pokud by byl k dispozici výkonnější hardware. Neefektivní způsob zpracovávání by ale pořád zůstal stejný. Pro testy s rychlostmi, které by dosahovaly teoretického maxima by mohl být využit např. samostatný externí generátor provozu.

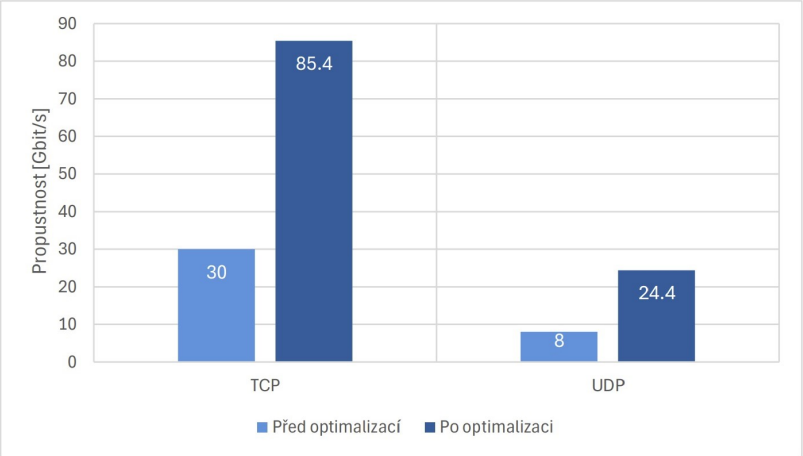
2.5 Analýza důsledků optimalizace

Optimalizační kroky, které jsou popsané v předchozí podkapitole, měly zásadní dopad na výkonnost sítě. Zásadní rozdíl byl pozorován po povolení PCI passthrough pro 100G rozhraní síťové karty. Srovnání hodnot latence je zobrazeno na obr. č. 2.1.



Obr. 2.1: Graf porovnání hodnot RTT před a po optimalizaci.

Hodnota RTT byla se zmenšila téměř na třetinu. Důsledky optimalizace byly pozorovány zejména na hodnotách propustnosti. Srovnání nejvyšších dosažených rychlostí TCP a UDP provozu pro velikost MTU 1500 B je zobrazeno na obr. č. 2.3.



Obr. 2.2: Graf porovnání propustností TCP a UDP provozu před a po optimalizaci.

S největší pravděpodobností je takto zásadní rozdíl způsoben kumulativními změnami, které se projevili až při povolení PCI passthrough. Ve virtualizovaném prostředí pracují stanice s virtuálními prvky (virtuální síťové rozhraní) a díky změnám jako IOMMU[19] a následným povolením PCI passthrough, se umožní, aby stanice pracovaly přímo s fyzickými prvky, což může mít zásadní dopad na výkon sítě.

2.6 Analýza síťového provozu

Pro analýzu testovacího síťového provozu byl využit nástroj Wireshark. Následující ukázky popisují TCP a UDP provoz realizovaný s vytížením všech dostupných CPU jader, a to pro MTU 1500 B.

No.	Time	Source	Destination	Protocol	Length	Info
94192	5.998937558	10.10.0.2	10.10.0.1	TCP	65226	44012 → 5001 [PSH, ACK] Se
94193	5.999094542	10.10.0.2	10.10.0.1	TCP	65226	[TCP Previous segment not
94194	5.999125295	10.10.0.2	10.10.0.1	TCP	65226	44012 → 5001 [PSH, ACK] Se
94195	5.999128804	10.10.0.1	10.10.0.2	TCP	66	[TCP ACKed unseen segment]
94196	5.999163565	10.10.0.2	10.10.0.1	TCP	1626	44012 → 5001 [PSH, ACK] Se
94197	5.999164526	10.10.0.2	10.10.0.1	TCP	65226	44012 → 5001 [PSH, ACK] Se
94198	5.999168348	10.10.0.1	10.10.0.2	TCP	66	5001 → 43990 [ACK] Seq=29
94199	5.999286629	10.10.0.2	10.10.0.1	TCP	65226	[TCP Previous segment not
94200	5.999293590	10.10.0.2	10.10.0.1	TCP	65226	[TCP Previous segment not
94201	5.999313205	10.10.0.1	10.10.0.2	TCP	66	[TCP ACKed unseen segment]
94202	5.999330068	10.10.0.2	10.10.0.1	TCP	65226	44012 → 5001 [PSH, ACK] Se
94203	5.999348311	10.10.0.2	10.10.0.1	TCP	65226	44012 → 5001 [PSH, ACK] Se
94204	5.999368659	10.10.0.1	10.10.0.2	TCP	66	5001 → 44012 [ACK] Seq=29
94205	5.999546385	10.10.0.1	10.10.0.2	TCP	66	5001 → 44012 [ACK] Seq=29
94206	5.999548982	10.10.0.2	10.10.0.1	TCP	818	43990 → 5001 [PSH, ACK] Se
94207	5.999549515	10.10.0.2	10.10.0.1	TCP	63778	[TCP Previous segment not
94208	5.999567990	10.10.0.2	10.10.0.1	TCP	65226	43990 → 5001 [PSH, ACK] Se
94209	5.999574206	10.10.0.2	10.10.0.1	TCP	65226	44012 → 5001 [PSH, ACK] Se

Obr. 2.3: Ukázka testovacího TCP provozu s maximálním vytížením – MTU 1500 B.

Při analýze TCP provozu, přesahujícího rychlosti 80 Gbit/s, bylo zjevné, že strana příjemce nezachycovala všechen zasílaný provoz. To se projevovalo zprávami typu `TCP Segment not captured` a `TCP ACKed unseen segment`. Vzhledem k tomu, že se jednalo o maximální dosažitelný provoz, byl výskyt zpráv způsoben ztrátovostí, což vyžadovalo opakované zasílání dat, a přetížením, kdy protokol TCP na tuto skutečnost reagoval. Při zachycení provozu byl také blíže pozorován vliv optimalizace na způsob zpracování přenášených dat. Zasílané segmenty byly typicky o délce 65226 B, což by odpovídalo výchozí hodnotě pro automatické ladění bufferů `net.ipv4.tcp_wmem`, která byla nastavena v systémové konfiguraci stanic, viz podkapitola „Optimalizace“ 2.4. Vlivem povolení offload mechanismů TSO a GRO tak dělení segmentů do velikosti jednotlivých paketů a skládání zpět do segmentů probíhalo na síťové kartě. Wireshark tedy zachytil celé TCP segmenty před zpracováním na NIC v případě, že byl spuštěn na straně odesílatele, nebo po zpětném složení na straně příjemce. Jelikož se jednalo o hraniční vytížení hardware, při aktivním zachycování provozu programem Wireshark bylo pozorováno snížení propustnosti na hodnotu ∼70 Gbit/s. Rozdílné zdrojové porty reprezentují jednotlivé proudy síťového provozu, jejichž počet byl nastaven v příkazu `iperf`. Cílový port `5001` je implicitní číslo portu pro iperf příjemce (server). Změny zahrnující řazení paketů a offload mechanismy měly za následek zvýšení využití prostředků příjemce a naopak snížení u vysílací strany. Procesor příjemce byl tedy maximálně vytížen, zatímco vytížení procesoru zdroje bylo 80–90%. Tento jev byl pozorován především u TCP provozu.

Při analýze provozu o rychlosti ∼20 Gbit/s byla frekvence opakovaného zasílaní dat velmi podobná. Zachycený UDP provoz v programu Wireshark zobrazuje obr. č. 2.4.

No.	Time	Source	Destination	Protocol	Length	Info
8291…	5.184717338	10.10.0.2	10.10.0.1	UDP	1500	56138 → 5001 Len=1458
8291…	5.184716779	10.10.0.2	10.10.0.1	UDP	1500	32794 → 5001 Len=1458
8291…	5.184722203	10.10.0.2	10.10.0.1	UDP	1500	34922 → 5001 Len=1458
8291…	5.184722333	10.10.0.2	10.10.0.1	UDP	1500	56138 → 5001 Len=1458
8291…	5.184723093	10.10.0.2	10.10.0.1	UDP	1500	33463 → 5001 Len=1458
8291…	5.184725513	10.10.0.2	10.10.0.1	UDP	1500	42123 → 5001 Len=1458
8291…	5.184726296	10.10.0.2	10.10.0.1	UDP	1500	57036 → 5001 Len=1458
8291…	5.184726296	10.10.0.2	10.10.0.1	UDP	1500	56138 → 5001 Len=1458
8291…	5.184738762	10.10.0.2	10.10.0.1	UDP	1500	56138 → 5001 Len=1458
8291…	5.184727495	10.10.0.2	10.10.0.1	UDP	1500	32794 → 5001 Len=1458
8291…	5.184739856	10.10.0.2	10.10.0.1	UDP	1500	56138 → 5001 Len=1458
8291…	5.184741654	10.10.0.2	10.10.0.1	UDP	1500	42123 → 5001 Len=1458
8291…	5.184743590	10.10.0.2	10.10.0.1	UDP	1500	33463 → 5001 Len=1458
8291…	5.184743046	10.10.0.2	10.10.0.1	UDP	1500	56138 → 5001 Len=1458
8291…	5.184745032	10.10.0.2	10.10.0.1	UDP	1500	56138 → 5001 Len=1458
8291…	5.184745188	10.10.0.2	10.10.0.1	UDP	1500	57036 → 5001 Len=1458
8291…	5.184745375	10.10.0.2	10.10.0.1	UDP	1500	56138 → 5001 Len=1458
8291…	5.184746843	10.10.0.2	10.10.0.1	UDP	1500	42123 → 5001 Len=1458

Obr. 2.4: Ukázka testovacího UDP provozu s maximálním vytížením – MTU 1500 B.

UDP provoz dosahoval propustnosti ∼24 Gbit/s a aktivní zachycování provozu opět způsobilo nepatrné snížení této hodnoty. Pro testovací UDP provoz, který byl generovaný iPerf2, bylo nutné upravit velikost datagramu tak, aby odpovídala aktuálně nastavené velikosti MTU. To bylo provedeno pomocí parametru `-l 1458` (velikost bufferu) pro testy o velikosti MTU 1500 B. Nastavená velikost datagramu tak brala v potaz záhlaví i na nižších vrstvách a celková přenášená velikost jednoho rámce činila 1500 B. Program iPerf2 implicitně používá délku UDP datagramu 1470 B a nebere v potaz nastavení MTU na síťových rozhraních. Parametr `-l` byl proto využit ve všech testech UDP provozu pro zajištění příslušné velikosti datagramu, např. `-l 8k` pro testy s MTU 8000 B. Procentuální hodnoty ztrátovosti poskytl iPerf po ukončení jednotlivých testů. Průměrná hodnota ztrátovosti celkem tří provedených testů pro MTU 1500 B, vypočtena ze všech paralelních proudů testovacího provozu, byla 3,38 %.

Výsledné hodnoty testů latence, jež jsou uvedeny v předchozí části, byly ověřovány opakovanými měřeními. Čas vykonání příkazu `ping` byl výsledkem časového intervalu výměny dvou zpráv mezi stanicemi, tedy `ICMP Echo Request` a `ICMP Echo Reply`. Výsledný čas pro příkaz `traceroute` zahrnoval mimo ICMP (Internet Control Message Protocol) zprávy i přenos UDP datagramů a ARP dotazy/odpovědi – proto byl celkový čas výrazně vyšší. Čas pro příkaz `traceroute`, pokud proveden v dostatečných časových odstupech, byl konzistentní před optimalizací (∼1,536 ms) i po ní (∼0,508 ms) a přestože se jednalo pouze o přímo připojené stanice, plnil `traceroute` funkci ověření hodnot latence a vliv optimalizace na tyto hodnoty.

[0]Na 100G síťové kartě (Mellanox ConnectX-5) byl firmware verze 16.30.1004 a komunikace probíhala přes Ethernet, dodatečné ovladače umožňující komunikaci Infiniband aj. využity nebyly.

3 Realizace zátěžových testů

V této kapitole jsou prezentovány dosažené výsledky testů propustnosti provozu bez šifrování a poté s šifrováním. Počáteční testy šifrovaného provozu byly zhotoveny za pomoci šifrátorů na platformě FPGA (Field Programmable Gate Array). Testy pro šifrovaný provoz byly také realizovány s využitím komerčně dostupných šifrátorů, jež byly součástí síťových prvků Juniper. Zátěžové testy byly provedeny pro čtyři různé velikosti MTU a s různým počtem CPU jader. Velikosti MTU byly zvoleny následovně: 576 B – minimální velikost IPv4 paketu, může sloužit jako referenční hodnota, pokud by docházelo k fragmentaci, 1300 B – referenční velikost pro minimální velikost IPv6 paketů (1280 B) nebo šifrovaný provoz, kdy se mohou enkapsulací přidávat nová záhlaví apod., 1500 B – běžná hodnota pro Ethernet/Ethernet II (1518 B) a 9000 B – používané ve vysokorychlostních infrastrukturách.

3.1 Provoz bez šifrování

Testy pro vícejádrový výkon, které jsou zobrazeny v tabulkách, byly realizovány pomocí iPerf2. Rychlostí blížících se 100 Gbit/s bylo dosaženo s využitím 10–16 paralelních proudů a velikostí TCP okna 256kB[3] nebo 2MB. Analogicky pro testy dosahující kolem 60 Gbit/s, byl počet proudů 4–8. Užití více či méně paralelních proudů, než bylo pro daný test optimální, vedlo k nižším propustnostem. Optimální počet proudů spočíval v nalezení hodnoty, při které docházelo k maximálnímu vytížení přidělených CPU jader tak, aby nedocházelo k přetížení. Suboptimální počet proudů nevytěžoval přidělená CPU jádra dostatečně, tím nebyla jádra vytížena na maximum, a rychlost tak nedosahovala maxima. Výsledné propustnosti dosahovaly vyšších hodnot při větších jednotkách MTU. Pro dosažení obdobných rychlostí s většími MTU oproti menších MTU, byl počet užitých proudů menší. Výsledné hodnoty testů propustnosti pro nešifrovaný TCP provoz jsou zobrazeny v tabulce č. 3.1.

Tab. 3.1: Výsledky testů propustnosti pro nešifrovaný TCP provoz.

Počet CPU jader	MTU [B]			
	576	1300	1500	9000
1	14,0 Gbit/s	26,0 Gbit/s	30,6 Gbit/s	32,4 Gbit/s
2	24,4 Gbit/s	39,4 Gbit/s	46,5 Gbit/s	55,1 Gbit/s
4	32,1 Gbit/s	63,6 Gbit/s	64,7 Gbit/s	69,1 Gbit/s
6	34,0 Gbit/s	72,8 Gbit/s	76,0 Gbit/s	92,1 Gbit/s
8	36,1 Gbit/s	78,4 Gbit/s	82,4 Gbit/s	96,1 Gbit/s
10	38,9 Gbit/s	81,7 Gbit/s	85,4 Gbit/s	96,4 Gbit/s

Pro běžnou velikost MTU bylo dosaženo 85 Gbit/s s využitím všech jader, která byla k dispozici. S dalšími optimalizačními kroky nebo více jádry CPU k dispozici, lze očekávat další navýšení hodnot[23]. Poté byl testován UDP provoz. Výsledky testů propustnosti pro nešifrovaný UDP provoz jsou zobrazeny v tabulce č. 3.2.

Tab. 3.2: Výsledky testů propustnosti pro nešifrovaný UDP provoz.

Počet CPU jader	MTU [B]			
	576	1300	1500	8000
1	1,3 Gbit/s	3,1 Gbit/s	3,6 Gbit/s	15,7 Gbit/s
2	2,3 Gbit/s	5,9 Gbit/s	5,8 Gbit/s	28,9 Gbit/s
4	4,5 Gbit/s	10,6 Gbit/s	11,5 Gbit/s	52,6 Gbit/s
6	6,3 Gbit/s	15,1 Gbit/s	16,9 Gbit/s	75,2 Gbit/s
8	7,8 Gbit/s	18,6 Gbit/s	22,7 Gbit/s	95,2 Gbit/s
10	9,4 Gbit/s	21,4 Gbit/s	24,4 Gbit/s	∼100,0 Gbit/s

Výsledné rychlosti testů propustnosti UDP provozu byly nižší v porovnání s TCP. Příčina nižších rychlostí je diskutována v posledním odstavci podkapitoly 2.4. V situaci skutečného provozu lze tak očekávat i vyšší propustnost. Dosažitelné rychlosti však budou záviset na vytížení CPU, způsobu generování provozu apod.

3.2 Provoz s šifrováním

Testy šifrovaného provozu byly realizovány za užití dvou různých typů šifrátorů, na platformě FPGA – Xilinx UltraScale a Agilex hardware, a firewallů Juniper SRX4200. Testy byly provedeny zejména se zaměřením na nejvyšší výkon, tzn. využití všech jader CPU, které byly pro virtuální stanice k dispozici. Pro referenci byly rovněž provedeny testy jednojádrového výkonu, jejichž výsledky jsou uvedeny v tabulkách společně výsledky maximálního výkonu. Samotné testy byly pro každou hodnotu provedeny min. třikrát a v tabulkách jsou uvedeny ve formě průměrné hodnoty s odchylkou. Testované velikosti MTU byly uzpůsobeny schopnostem jednotlivých šifrátorů.

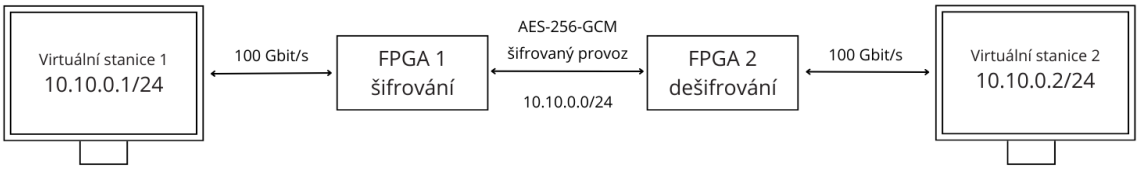
3.2.1 Platforma FPGA

Platforma FPGA nabízí flexibilní a efektivní řešení pro různé výpočetní úlohy, jelikož ji lze programovat přímo pro danou činnost. Efektivita platformy spočívá v akceleraci a paralelním zpracování s využitím hardwarových obvodů[20]. Testované šifrátory FPGA byly vyvíjeny na VUT FEKT a pro šifrovací algoritmus byl využit AES (Advanced Encryption Standard) v módu GCM (Galois/Counter Mode) a délkou klíče 256 bitů. Šifra AES-256-GCM nabízí autentizované šifrování dat, je vhodná

pro vysoké propustnosti a zároveň je také kvantově odolná[21]. Předpokladem pro výsledky šifrovaného provozu pomocí FPGA byla žádná či minimální limitace oproti hodnotám propustnosti provozu bez šifrování.

Platforma FPGA – Silicom Denmark fb4CGg3@VU9P (Mango)

Jako první byly na linku 100 Gbit/s přidány šifrátory FPGA (Silicom Denmark fb4CGg3@VU9P) v podobě SFF (Small Form Factor) počítačů. Šifrátory disponovaly 100G rozhraním, takže teoretická propustnost linky zůstala nezměněná. FPGA jsou transparentní – pro realizaci testů byly na virtuálních stanicích stejné IP adresy, jako v zapojení bez šifrátorů. Topologie po zapojení šifrátorů je zobrazena na obr. č. 3.1.



Obr. 3.1: Vizualizace topologie pro testování s šifrátory na platformě FPGA.

Doba trvání samotných testů byla opět 30 sekund a interval mezi jednotlivými testy byl různý. Tím bylo dosaženo mírně rozdílných výsledků, které byly způsobeny změnami v pracovní frekvenci využívaných jader procesoru. Hodnoty propustnosti zobrazené v tabulkách představují průměrnou rychlost vypočtenou na základě všech provedených testů. Odchylka značí rozdíl mezi průměrnou rychlostí a výsledkem testu, jehož hodnota byla nejvíce vzdálená od průměrné. Testy byly provedeny pouze pro kategorie MTU 576 B, 1300 B a 1500 B. Šifrátory FPGA mají velikost MTU implicitně nastavenou na hodnotu 1526 B a větší MTU jsou považovány za neplatné. Výsledky šifrovaného TCP provozu jsou prezentovány v tabulce č. 3.3.

Tab. 3.3: Výsledky testů propustnosti, TCP – FPGA UltraScale+ (Mango).

Počet CPU jader	MTU [B]		
	576	1300	1500
1	14,0 ± 0,9 Gbit/s	23,5 ± 1,4 Gbit/s	25,7 ± 2,0 Gbit/s
10	42,0 ± 5,4 Gbit/s	75,2 ± 2,6 Gbit/s	79,6 ± 3,6 Gbit/s

Tabulka s výsledky testů UDP provozu zobrazuje propustnost na straně příjemce, tedy odeslaná data odečtená o ztracená data. Tyto hodnoty poskytl iPerf po dokončení jednotlivých testů. Objem vysílaného provozu, regulován počtem paralelních proudů v iPerf příkazu, byl zvolen tak, aby docházelo k maximálnímu využití prostředků a zároveň nedocházelo k přebytečné ztrátovosti. Ve velmi omezených případech byla pozorována ztrátovost u jednoho z proudů 10–12%, ztrátovost

všech ostatních proudů byla <<5%. Procentuální hodnoty ztrátovosti byly proměnné a proto byly testy UDP provozu 5–10-krát pro danou kategorii. Výsledné hodnoty šifrovaného UDP provozu jsou prezentovány v tabulce č. 3.4.

Tab. 3.4: Výsledky testů propustnosti, UDP – FPGA UltraScale+ (Mango).

Počet CPU jader	MTU [B]		
	576	1300	1500
1	1,3 ± 0,2 Gbit/s	3,1 ± 0,4 Gbit/s	3,4 ± 0,2 Gbit/s
10	10,4 ± 0,4 Gbit/s	22,5 ± 2,4 Gbit/s	26,5 ± 2,1 Gbit/s

Rychlosti dosažené při testech propustnosti byly srovnatelné s hodnotami nešifrovaného provozu. Největší pozorovaný rozdíl byl u TCP provozu s velikostí MTU 1500 B – 30,6 Gbit/s vs 25,7 Gbit/s pro jednojádrový výkon a 85,4 Gbit/s vs 79,2 Gbit/s pro desetijádrový výkon. Hodnota latence, testovaná pomocí příkazu `ping`, se pro 13 dotazů a opakovaná měření, pohybovala v rozmezí 0,231–0,288 ms. Po optimalizaci a pro přímo připojené virtuální stanice byla hodnota latence 0,249 ms. Hodnoty získané po zapojení šifrátorů FPGA do 100G linky tedy nebyly výrazně rozdílné.

Po dobu testování způsoboval testovací provoz desynchronizaci klíčů na šifrátorech. Chyby na kartách vznikaly v případě, kdy byl určitý paket vyhodnocen jako neplatný, což mělo za následek desynchronizaci a klíče poté musely být opět synchronizovány manuálně. V rámci testování tedy probíhala komunikace s vývojáři FPGA šifrátorů a průběžné aktualizace firmware šifrátorů.

Platforma FPGA – Silicom SmartNIC N6010 (Intel)

V další fázi byly realizovány testy s šifrátory FPGA a Silicom SmartNIC N6010 hardware v podobě serverových šifrátorů. Topologie zapojení a způsob testování byly shodné jako u předchozích FPGA. Výsledné hodnoty šifrovaného TCP provozu zobrazuje tab. č. 3.5.

Tab. 3.5: Výsledky testů propustnosti, TCP – FPGA Agilex (Intel).

Počet CPU jader	MTU [B]		
	576	1300	1500
1	15,2 ± 0,5 Gbit/s	23,5 ± 1,0 Gbit/s	25,3 ± 1,5 Gbit/s
10	43,3 ± 2,0 Gbit/s	81,3 ± 1,0 Gbit/s	85,8 ± 1,4 Gbit/s

Propustnosti TCP provozu byly opět téměř shodné s hodnotami nešifrovaného provozu a také dříve testovaných šifrátorů. Rovněž tomu tak bylo i pro výsledky UDP provozu, které jsou zobrazeny v tabulce č. 3.6.

Tab. 3.6: Výsledky testů propustnosti, UDP – FPGA Agilex (Intel).

Počet CPU jader	MTU [B]		
	576	1300	1500
1	1,3 ± 0,1 Gbit/s	2,9 ± 0,5 Gbit/s	3,5 ± 0,1 Gbit/s
10	10,5 ± 0,1 Gbit/s	23,2 ± 1,6 Gbit/s	26,5 ± 2,3 Gbit/s

Po zhotovení testů šifrovaného provozu byly realizovány testy s QKD (Quantum Key Distribution). Hodnoty šifrovaného provozu s QKD uvádí tab. č. 3.7 a tab. č. 3.8.

Tab. 3.7: Výsledky testů propustnosti, TCP – FPGA Agilex (Intel), QKD.

Počet CPU jader	MTU [B]		
	576	1300	1500
1	14,9 ± 1,1 Gbit/s	23,3 ± 2,5 Gbit/s	24,6 ± 1,8 Gbit/s
10	43,0 ± 0,2 Gbit/s	81,7 ± 2,7 Gbit/s	84,0 ± 2,8 Gbit/s

Tab. 3.8: Výsledky testů propustnosti, UDP – FPGA Agilex (Intel), QKD.

Počet CPU jader	MTU [B]		
	576	1300	1500
1	1,4 ± 0,1 Gbit/s	3,0 ± 0,2 Gbit/s	3,5 ± 0,1 Gbit/s
10	10,8 ± 0,4 Gbit/s	24,4 ± 1,5 Gbit/s	26,4 ± 0,8 Gbit/s

Režim QKD neměl na výsledné propustnosti vliv. Hodnoty pro TCP i UDP provoz byly v rámci odchylek téměř shodné s předchozími testy. Funkčnost QKD opět vyžadovala několik verzí firmware. Pro ověření funkčnosti šifrátorů v dlouhodobém časovém intervalu byly realizovány tři testy v rozmezí 2, 6 a 20 hodin. Jednalo se o test s vytížením jednoho jádra CPU na každé stanici, kdy bylo cílem úspěšně realizovat provoz po celý časový interval. Tyto testy byly provedeny pro provoz v režimech s i bez QKD a proběhly úspěšně. Průměrné rychlosti uvádí tab. č. 3.9.

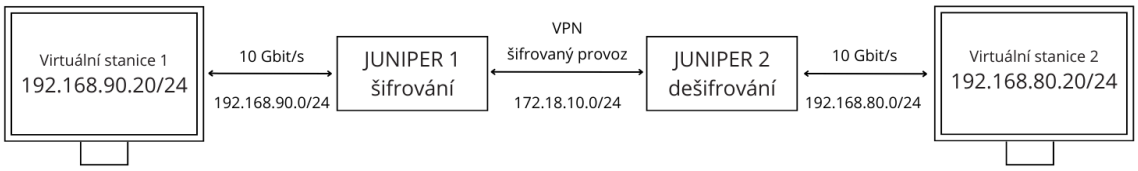
Tab. 3.9: Výsledky dlouhodobě šifrovaného provozu – FPGA Agilex (Intel).

Režim provozu	Délka testu		
	2 hodiny	6 hodin	20 hodin
bez QKD	12,8 Gbit/s	12,8 Gbit/s	12,7 Gbit/s
s QKD	11,2 Gbit/s	14,9 Gbit/s	11,6 Gbit/s

Při testech propustnosti s šifrátory FPGA byl naplněn předpoklad, že vliv samotných šifrátorů na výsledné hodnoty nebyl velký. Limitace provozu byla minimální.

3.2.2 Firewally Juniper

Firewally Juniper SRX4200 jsou komerční síťové prvky[24]. Podporují standard RFC 8784[25], což umožňuje realizovat kvantově bezpečné spojení pomocí IPsec (Internet Protocol Security) a IKEv2 (Internet Key Exchange Protocol Version 2). Firewally SRX4200 disponují 10G rozhraním, testování tedy probíhalo na samostatné lince o teoretické rychlosti 10 Gbit/s. V případě testeru se jednalo o rozhraní 10G síťové karty Broadcom BCM57412. Pro zhotovení testů byly využity dva firewally, mezi nimiž byl ustanoven kvantově odolný VPN spoj. Firewally byly připojeny k testeru obdobně jako FPGA. Topologii zapojení spolu s adresací stanic zobrazuje obr. č. 3.2.



Obr. 3.2: Vizualizace topologie pro testování s šifrátory Juniper.

Jelikož se jednalo o samostatnou linku, byly v průběhu testování provedeny stejné optimalizační kroky, jako pro 100G linku, uvedené v podkapitole „Optimalizace“ 2.4. Všechny výsledky, uvedené v této části, jsou prezentovány po provedení optimalizace 10G linky. Pro tuto linku byly testy pro nešifrovaný provoz zhotoveny samostatně a jsou zobrazeny v tabulce 3.10. Tyto testy byly realizovány pro přímo propojené stanice před přidáním firewallů.

Tab. 3.10: Výsledky testů propustnosti 10G linky pro nešifrovaný provoz.

TCP				
Počet CPU jader	MTU [B]			
	576	1300	1500	9000
1	8,5 Gbit/s	9,3 Gbit/s	9,4 Gbit/s	9,9 Gbit/s
10	8,5 Gbit/s	9,3 Gbit/s	9,4 Gbit/s	9,9 Gbit/s
UDP				
Počet CPU jader	MTU [B]			
	576	1300	1500	9000
1	1,3 Gbit/s	3,0 Gbit/s	3,5 Gbit/s	8,0 ± 1 Gbit/s
10	2 Gbit/s	7,0 Gbit/s	8,0 ± 1 Gbit/s	8,5 ± 1 Gbit/s

Rychlosti TCP provozu byly stejné pro jednojádrový i desetijádrový výkon. Provoz byl limitován kapacitou linky, ne vytížením CPU. Provoz UDP byl opět výrazně

limitován a proměnné rychlosti jsou uvedeny s odchylkou. Průměrné hodnoty latence se pohybovaly v rozmezí 0,600 ms až 0,750 ms před optimalizací a 0,300 ms až 0,405 ms po optimalizaci.

Pro šifrování provozu umožňují firewally volit a kombinovat vybrané parametry VPN komunikace. Lze volit různé autentizační mechanismy (ověřující přenášená data) a autentizační metodu (autentizující zdroj zpráv IKE, jako např. certifikáty, předsdílené klíče apod.). Pro parametry IPsec lze volit různé autentizační algoritmy (typ hashovacího algoritmu – ověřující data v paketech) a samotné šifrovací algoritmy. Cílem následujících testů tedy bylo nalézt nejvýkonnější a zároveň bezpečnou kombinaci těchto parametrů, která by byla vhodná pro kvantově odolný kanál. Pro daný model SRX4200 je výrobcem uvedena maximální propustnost IPsec VPN (Virtual Private Network) komunikace 35 Gbit/s [24]. Tento prvek však disponuje 8x10G Ethernet rozhraními, takže se jedná o celkovou propustnost při komunikaci uskutečněnou s více rozhraními. Následující testy byly zhotoveny pro jedno 10G rozhraní na každém z firewallů. Zároveň výrobce u hodnoty propustnosti VPN uvádí jednotku MTU 1400 B, proto byly následující testy provedeny i pro tuto velikost. Pro velikosti MTU >1400 B (na koncových bodech) byla pozorována výrazná limitace provozu způsobená „přetékáním“ dat v důsledku přidávání IPsec záhlaví, kdy docházelo k fragmentaci. Propustnost pro všechna větší MTU a různé parametry VPN byla maximálně 1,6 Gbit/s. V průběhu testování však byla objevena možnost změnit hodnotu MTU pro IPsec na firewallech[26]. Na firewallech tedy bylo nastaveno maximální MTU 9192 B a při testování, včetně MTU 9000 B na virtuálních stanicích, tak nedocházelo k fragmentaci. Hodnoty následujících testů jsou uvedeny po změně maximální MTU na firewallech.

Pro autentizační metodu a autentizační mechanismus VPN komunikace byla zvolena kombinace předsdílených klíčů (PSK) a SHA-256. Tyto parametry zůstaly po celou dobu testování stejné, jelikož nebyl předpokládán výrazný dopad na propustnost v důsledku změn těchto parametrů. Prvotní testy proběhly pro šifrovací algoritmy DES (Data Encryption Standard) a 3DES (Triple DES) s hashovacím algoritmem HMAC-SHA-256-128. Algoritmus DES je v současnosti považován za nebezpečný kvůli krátké délce klíče – 56 bitů. Tento šifrovací algoritmus byl jako první zvolen proto, že byla očekávána relativně vysoká propustnost kvůli výpočetní nenáročnosti šifrování a hodnoty získané z těchto testů by poté mohly být porovnány s dalšími výsledky. Pro všechny testy s síťovými prvky SRX4200 byla pozorována limitace samotnými firewally, výkon tedy nezávisel na výpočetním výkonu hardware testeru, ale právě firewallů. Výsledné hodnoty testů tedy byly poměrně jednoznačné a proto jsou hodnoty v následujících tabulkách bez uvedeny odchylek. Průběh následujícího testování byl totožný s testy na platformě FPGA, vyjma rozdílných kategorií MTU. Výsledky prvotních testů s šifrovacím algoritmem DES prezentuje tab. č. 3.11

Tab. 3.11: Výsledky testů propustnosti VPN komunikace (Juniper SRX4200), Šifrovací algoritmus: DES-CBC, Hashovací algoritmus: HMAC-SHA-256-128.

TCP					
Počet CPU jader	MTU [B]				
	576	1300	1400	1500	9000
1	1,4 Gbit/s	3,3 Gbit/s	3,5 Gbit/s	3,8 Gbit/s	9,8 Gbit/s
10	1,4 Gbit/s	3,3 Gbit/s	3,5 Gbit/s	3,8 Gbit/s	9,8 Gbit/s
UDP					
Počet CPU jader	MTU [B]				
	576	1300	1400	1500	9000
1	1,2 Gbit/s	2,4 Gbit/s	3,0 Gbit/s	3,0 Gbit/s	9,8 Gbit/s
10	1,4 Gbit/s	2,5 Gbit/s	3,6 Gbit/s	3,8 Gbit/s	9,8 Gbit/s

Nejvyšší propustnost o rychlosti 9,8 Gbit/s byla dosažena pro jednotku MTU 9000 B. Maximální propustnost tedy dosahovala teoretického maxima 10G linky. Konečná velikost přenášených paketů byla vyšší než 9000 B v důsledku zapouzdřovacích dat IPsec – AH (Authentication Header) a ESP (Encapsulating Security Payload). Velikost zapouzdřovacích dat byla nižší, než rozdíl 9192-9000 B a nedocházelo tak k „přetékání“ ani pro tuto velikost. V dalším kroku byl šifrovací algoritmus změněn na 3DES a výsledky těchto testů jsou zobrazeny v tabulce č. 3.12.

Tab. 3.12: Výsledky testů propustnosti VPN komunikace (Juniper SRX4200), Šifrovací algoritmus: 3DES-CBC, Hashovací algoritmus: HMAC-SHA-256-128.

TCP					
Počet CPU jader	MTU [B]				
	576	1300	1400	1500	9000
1	1,4 Gbit/s	3,3 Gbit/s	3,5 Gbit/s	3,8 Gbit/s	9,8 Gbit/s
10	1,4 Gbit/s	3,3 Gbit/s	3,5 Gbit/s	3,8 Gbit/s	9,8 Gbit/s
UDP					
Počet CPU jader	MTU [B]				
	576	1300	1400	1500	9000
1	1,2 Gbit/s	2,5 Gbit/s	3,0 Gbit/s	3,0 Gbit/s	9,8 Gbit/s
10	1,4 Gbit/s	3,3 Gbit/s	3,6 Gbit/s	3,8 Gbit/s	9,8 Gbit/s

Výsledné hodnoty propustnosti byly téměř totožné s předchozími. Maximální dosažená rychlost byla 9,8 Gbit/s. V další fázi byl šifrovací algoritmus nastaven na AES-256-CBC. Výsledky po provedení změny prezentuje tab. č. 3.13.

Tab. 3.13: Výsledky testů propustnosti VPN komunikace (Juniper SRX4200), Šifrovací algoritmus: AES-256-CBC, Hashovací algoritmus: HMAC-SHA-256-128.

TCP					
Počet CPU jader	MTU [B]				
	576	1300	1400	1500	9000
1	2,0 Gbit/s	3,0 Gbit/s	3,2 Gbit/s	3,3 Gbit/s	4,8 Gbit/s
10	2,0 Gbit/s	3,0 Gbit/s	3,2 Gbit/s	3,3 Gbit/s	4,8 Gbit/s
UDP					
Počet CPU jader	MTU [B]				
	576	1300	1400	1500	9000
1	1,3 Gbit/s	2,4 Gbit/s	2,6 Gbit/s	2,7 Gbit/s	4,8 Gbit/s
10	2,1 Gbit/s	3,2 Gbit/s	3,3 Gbit/s	3,3 Gbit/s	4,8 Gbit/s

Při užití šifrovacího algoritmu AES-256-CBC a hashovacího algoritmu HMAC-SHA-256-128 klesla hodnota maximální propustnosti na 4,8 Gbit/s. Pro testy s MTU 1500 B a 9000 B byla pozorována výrazně vyšší limitace rychlosti. Tato limitace byla s největší pravděpodobností způsobena nedostatečným výkonem hashe SHA-256[27]. Vliv hashovacího algoritmu na dosažitelnou propustnost byl ověřen v následujících dvou testech. Při dalším testu zůstaly všechny parametry stejné a byl změněn pouze typ hashe, který byl nastaven na SHA-512. Výsledky testu po změně hashovacího algoritmu jsou zobrazeny v tab. č. 3.14.

Tab. 3.14: Výsledky testů propustnosti VPN komunikace (Juniper SRX4200), Šifrovací algoritmus: AES-256-CBC, Hashovací algoritmus: SHA-512.

TCP					
Počet CPU jader	MTU [B]				
	576	1300	1400	1500	9000
1	1,4 Gbit/s	3,3 Gbit/s	3,5 Gbit/s	3,8 Gbit/s	9,8 Gbit/s
10	1,4 Gbit/s	3,3 Gbit/s	3,5 Gbit/s	3,8 Gbit/s	9,8 Gbit/s
UDP					
Počet CPU jader	MTU [B]				
	576	1300	1400	1500	9000
1	1,3 Gbit/s	2,4 Gbit/s	2,4 Gbit/s	3,0 Gbit/s	9,8 Gbit/s
10	1,4 Gbit/s	3,3 Gbit/s	3,7 Gbit/s	3,8 Gbit/s	9,8 Gbit/s

Propustnost při velikosti MTU 9000 B opět dosahovala maxima linky. Algoritmus SHA-512 tedy nabízel výrazně vyšší výkon v porovnání s HMAC-SHA-256-128.

Typ hashovacího algoritmu však nebyl hlavním faktorem, který by limitoval síťový provoz. To bylo ověřeno realizací dalších testů po změně hashovacího algoritmu na SHA-384. Tyto hodnoty jsou uvedeny v tabulce č. 3.15.

Tab. 3.15: Výsledky testů propustnosti VPN komunikace (Juniper SRX4200), Šifrovací algoritmus: AES-256-CBC, Hashovací algoritmus: SHA-384.

TCP					
Počet CPU jader	MTU [B]				
	576	1300	1400	1500	9000
1	1,4 Gbit/s	3,3 Gbit/s	3,5 Gbit/s	3,8 Gbit/s	9,8 Gbit/s
10	1,4 Gbit/s	3,3 Gbit/s	3,5 Gbit/s	3,8 Gbit/s	9,8 Gbit/s
UDP					
Počet CPU jader	MTU [B]				
	576	1300	1400	1500	9000
1	1,2 Gbit/s	2,4 Gbit/s	2,7 Gbit/s	2,7 Gbit/s	9,8 Gbit/s
10	1,4 Gbit/s	3,3 Gbit/s	3,7 Gbit/s	3,8 Gbit/s	9,8 Gbit/s

Výsledné hodnoty kombinace AES-256-CBC + SHA384 byly velmi podobné s předchozími výsledky. Pro běžné velikosti MTU však doposud dosažené propustnosti byly vždy <4 Gbit/s. Dalším krokem tedy byla změna šifrovacího algoritmu na AES-256-CGM. Tato šifra je schopna samostatně zajistit také integritu šifrovaných dat. Pro dané firewally nebyla možnost kombinovat šifru a různými typy hashů. Síťový provoz šifrovaný AES a módem GCM tedy nevyužíval žádný další hashovací algoritmus pro zajištění integrity. Výsledky testů s AES-256-GCM uvádí tab. č. 3.16.

Tab. 3.16: Výsledky testů propustnosti VPN komunikace (Juniper SRX4200), Šifrovací algoritmus: AES-256-GCM.

TCP					
Počet CPU jader	MTU [B]				
	576	1300	1400	1500	9000
1	3,1 Gbit/s	6,0 Gbit/s	6,3 Gbit/s	6,7 Gbit/s	9,8 Gbit/s
10	3,1 Gbit/s	6,0 Gbit/s	6,3 Gbit/s	6,7 Gbit/s	9,8 Gbit/s
UDP					
Počet CPU jader	MTU [B]				
	576	1300	1400	1500	9000
1	1,2 Gbit/s	2,7 Gbit/s	2,7 Gbit/s	2,7 Gbit/s	9,8 Gbit/s
10	2,1 Gbit/s	6,1 Gbit/s	6,3 Gbit/s	6,7 Gbit/s	9,8 Gbit/s

Při užití AES v módu GCM bylo dosaženo propustnosti 6,7 Gbit/s pro běžnou velikost MTU. To je téměř dvojnásobek oproti předchozím testům. Jednalo se tak o doposud nejvýkonnější testovaný šifrovací algoritmus. Poslední test proběhl po změně na šifru AES-128-GCM. Výsledné hodnoty jsou prezentovány v tab. č. 3.17.

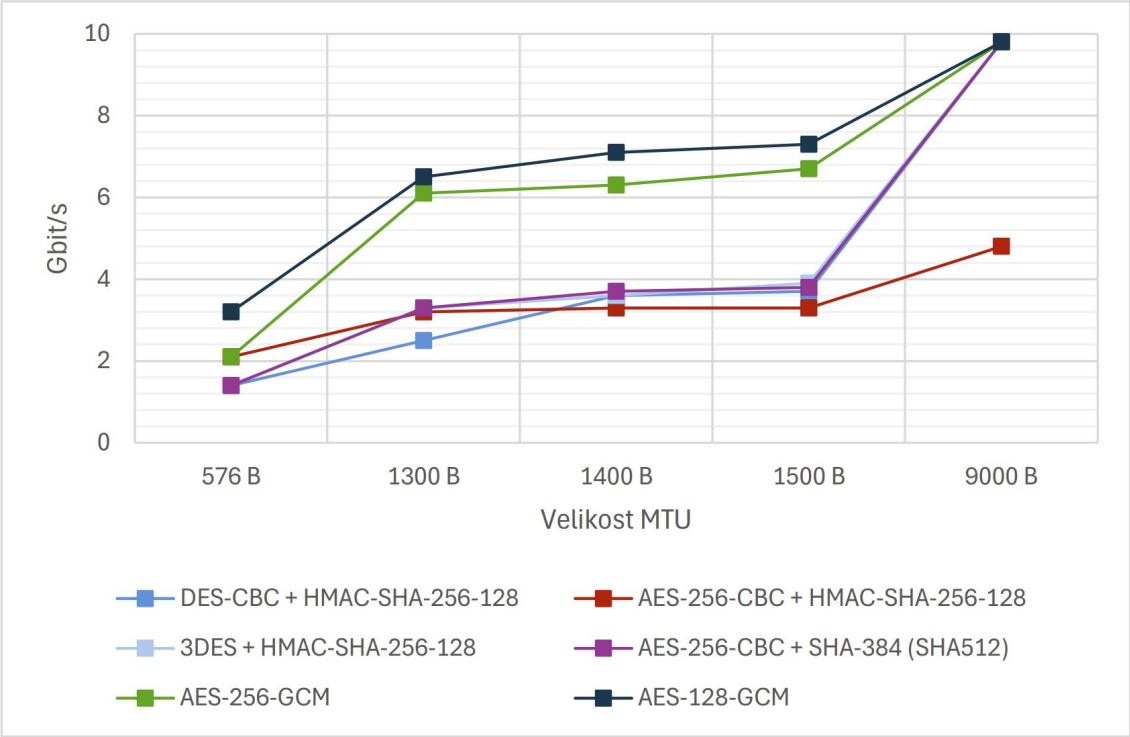
Tab. 3.17: Výsledky testů propustnosti VPN komunikace (Juniper SRX4200), Šifrovací algoritmus: AES-128-GCM.

TCP					
Počet CPU jader	MTU [B]				
	576	1300	1400	1500	9000
1	3,2 Gbit/s	6,4 Gbit/s	6,8 Gbit/s	7,2 Gbit/s	9,8 Gbit/s
10	3,2 Gbit/s	6,4 Gbit/s	6,8 Gbit/s	7,3 Gbit/s	9,8 Gbit/s
UDP					
Počet CPU jader	MTU [B]				
	576	1300	1400	1500	9000
1	1,3 Gbit/s	2,7 Gbit/s	2,7 Gbit/s	2,7 Gbit/s	9,8 Gbit/s
10	3,2 Gbit/s	6,5 Gbit/s	7,1 Gbit/s	7,3 Gbit/s	9,8 Gbit/s

Pro MTU 1500 B byla maximální propustnost 7,3 Gbit/s. Délka šifrovacího klíče měla také určitý vliv na celkový výkon.

Výše testované kombinace zahrnují téměř všechny parametry, které lze na firewallech nastavit. Parametry, které testovány nebyly, zahrnují hash typu MD5 nebo různé délky šifrovacích klíčů algoritmu AES-CBC. Při užití těchto parametrů nebyly předpokládány výrazné odchylky od získaných hodnot. Šifry DES, 3DES a AES-CBC dosahovaly přibližně stejného výkonu. Určitý vliv na výkon měl také typ autentizačního algoritmu – hashe, který byl zvolen pro ověřování dat. Jednoznačně nejvýkonnějším šifrovacím algoritmem byl AES v módu GCM. Při užití AES-GCM bylo dosaženo propustností ∼7 Gbit/s pro běžné velikosti MTU pro TCP i UDP provoz. Propustnost při velikosti MTU 9000 B byla limitována schopnostmi 10G linky. Tím bylo dosaženo maximální teoretické propustnosti pro jednu fyzickou linku. Při realizaci komunikace s více fyzickými linkami lze očekávat celkovou propustnost firewallů vyšší než 10 Gbit/s.

Nejvyšší dosažené propustnosti pro každou z testovaných kombinací zobrazuje graf na obr. č. 3.3. Hodnoty propustností jsou členěny dle testovaných velikostí MTU.



Obr. 3.3: Graf hodnot propustností pro různé parametry VPN – Juniper SRX4200.

Z grafu je zjevné, že výkon AES-GCM byl výrazně vyšší, než u ostatních testovaných šifer. Hodnoty propustností pro šifrovací algoritmy DES a AES-CBC jsou srovnatelné pro všechny testované velikosti MTU. Výjimkou je limitace HMAC-SHA-256-128 při užití AES-256-CBC a MTU 9000 B. Mimo tuto kombinaci bylo pro MTU velikosti 9000 B dosaženo vždy teoretického maxima 10G linky v podobě propustnosti 9,8 Gbit/s. Pro běžně používanou velikost MTU 1500 B se rychlosti pohybovaly v rozmezích 3,3 Gbit/s až 3,8 Gbit/s pro šifry AES a DES a 6,3 Gbit/s až 7,3 Gbit/s pro šifry AES-GCM.

Kombinace parametrů, která splňovala požadavky pro kvantově odolný kanál, byla zvolena následovně: Pro VPN kombinace PSK + SHA256 a pro IPsec šifrovací algoritmus AES-256-GCM. Zvolený šifrovací algoritmus byl stejný i pro šifrování na platformě FPGA.

4 Reálné aplikace

Tato kapitola popisuje využití výše uvedených šifrátorů v reálných aplikacích. Nejprve byla testována výkonnost síťové infrastruktury pro webovou komunikaci, a to v podobě měření rychlosti stahování souborů. Pro testování byly implementovány dvě varianty webového serveru zahrnující Python HTTP server a NGINX, kdy testy probíhaly na každém z serverů samostatně. V další fázi proběhly testy zaměřené na provoz v reálném čase za pomoci video streamu. K tomu byl využit nástroj VLC. Poslední část popisuje testování provozu přes vysokorychlostní přepínač, který byl přidán do 100 Gbit/s linky. Zde byl zkoumán potenciální vliv dalších síťových prvků na samotný provoz. Cílem těchto testů bylo ověřit schopnosti nejen šifrátorů, ale také samotné infrastruktury pro reálné situace a objevení případných dalších limitací.

4.1 Webová komunikace

Pro uskutečnění webové komunikace byly na jedné virtuální stanici zprovozněny dva různé webové servery – Python HTTP a NGINX. Dvě varianty webových serverů byly zvoleny kvůli jejich odlišnosti komplexnosti, způsobu využití hardware a dosažitelnému výkonu. Na stejnou virtuální stanici byly umístěny tři soubory o velikostech 100 MB, 1 G, a 10 GB. Adresář s těmito soubory byl přiřazen webovým serverům, které tak umožňovaly sdílení statického obsahu. Druhá virtuální stanice plnila roli klienta a na tuto stanici byl daný obsah sdílen, přičemž byly zkoumány rychlosti stahování. Aby komunikace probíhala po požadované lince, tedy přes šifrátory FPGA nebo firewally Juniper, bylo webovému serveru vždy přiděleno příslušné rozhraní.

4.1.1 Python HTTP

Python HTTP server je příkladem jednoduchého webového serveru. Implicitně se jedná o jednovláknový proces. Je již součástí balíčku Python a v distribuci Debian jej lze spustit jako skript příkazem `python3 -m http.server` bez další konfigurace. Umožňuje tak sdílení obsahu adresáře, ve které byl skript spuštěn[28]. Spuštění webového serveru na virtuální stanici dále zahrnovalo přiřazení portu `8080` a využití příslušného síťového rozhraní, tedy `--bind 192.168.90.20` pro firewally Juniper a `--bind 10.10.0.1` pro FPGA. Poté mohl být daný soubor stažen na straně klienta příkazem `wget http://<IP_adresa>:<port>/<nazev_souboru>`.

V úvodní fázi dosahovaly rychlosti stahování maximálních hodnot ∼550 MB/s, tedy v přepočtu 4,4 Gbit/s. Vzhledem k schopnostem šifrátorů se nejednalo o příliš vysoký provoz. Rychlost stahování a tedy i konečná doba stažení byla téměř konstantní pro všechny testované velikosti souborů. Nebyla pozorována žádná limitace

způsobená samotnými šifrátory např. v podobě proměnlivé ztrátovosti nebo vlivů, které by měly dopad na rychlost stahování. Provoz byl limitován rychlostí zápisu na disk klienta a proto v průběhu testování proběhly kroky optimalizace pro maximalizaci provozu.

Optimalizační kroky zahrnovaly především užití RAM disku pro ukládání souborů a testování různých způsobů samotného procesu stahování. Jelikož stanice klienta měla k dispozici 32 GB RAM a při testování byla využívána zhruba polovina, 12 GB RAM bylo přiděleno jako další disk s cílem maximalizace rychlosti zápisu[31]. Na tento disk byly dané soubory ukládány. Maximální rychlost zápisu na RAM disk byla zjištěna za pomoci nástroje `dd`, umožňující manipulaci s disky na OS Linux. Užitý příkaz[32] a získanou rychlost zápisu 2,1 GB/s uvádí výpis č. 4.1.

Výpis 4.1: Test rychlosti zápisu na RAM disk.

```
dd if=/dev/zero of=/tmp/ramdisk/test bs=1048576 count=4096
oflag=nocache conv=fsync

4096+0 records in
4096+0 records out
4294967296 bytes (4.3 GB, 4.0 GiB) copied,
2.012612 s, 2.1 GB/s
```

Rychlost zápisu 2,1 GB/s byla tedy nejvyšší možnou rychlostí pro aktuální hardware. Způsob, jakým byly soubory stahovány, však měl značný dopad na dosažitelnou rychlost. Užitím nástroje `wget` po přidělení RAM disku rychlost stahování dosahovala 650 MB/s. Pro `wget` byla tato hodnota maximum. Stahování pomocí prohlížeče bylo výrazně pomalejší, s rychlostmi dosahující ∼200 MB/s. Užitím utility `aria2` s paralelními připojeními byla maximální rychlost 750MB/s. Výrazné zvýšení výkonu a přiblížení se zjištěnému maximu zápisu na disk bylo pozorováno až při užití nástrojů `wget2` a `uGet`. S nástrojem `uGet` bylo dosaženo průběžné rychlosti 1647 MB/s. Jednalo se o maximální rychlost stahování jednoho souboru s využitím více paralelních spojení se serverem pomocí parametru `max_connections=16`. Maximální rychlosti zápisu bylo dosaženo až při užití `wget2` pro dvě paralelní stahování současně. Jednalo se o dva příkazy spuštěné současně v různých terminálech pro soubory o velikostech 10 GB a 1 GB. Tato kombinace byla vhodná, protože RAM disk měl velikost pouze 12 GB. Rychlosti obou stahování tedy přesahovaly 1 GB/s a byly konstantní po celou dobu stahování, přestože 1 GB soubor byl stažen výrazně rychleji. V případě šifrování FPGA tedy bylo dosaženo maximální rychlosti zapisování pro aktuální hardware 2,1 GB/s. Šifrátory FPGA tento provoz, v přepočtu 16,8 Gbit/s, dokázaly spolehlivě šifrovat s konstantní rychlostí. Pro firewally Juniper byla rychlost stahování ∼1,2 GB/s pro MTU 9000 B, což odpovídalo maximální

rychlosti linky ∼9,6 Gbit/s, viz obr. 3.3. V rámci pokusu o dosažení vyšších rychlostí přenosu byl otestován i přenos souborů pomocí FTP (File Transfer Protocol) v podobě `vsftpd`. Rychlosti přenosu FTP však byly v rozmezí 400 MB/s až 500 MB/s. Nebylo tak dosaženo vyšší rychlosti přenosu.

Následující tab. č. 4.1 zobrazuje dosažené rychlosti stahování pro Python HTTP. Jsou zde uvedeny hodnoty získané před a po provedení výše uvedených kroků, přičemž hodnota 2100 MB/s je dána součtem ze zmíněných dvou paralelních stahování.

Tab. 4.1: Rychlosti stahování souborů z webserveru – Python HTTP.

Linka	Před optimalizací	Po optimalizaci
Juniper SRX4200	550 MB/s	1200 MB/s
šifrátory FPGA	550 MB/s	2100 MB/s

Pro webový server Python HTTP bylo dosaženo maximální rychlosti stahování souborů. Zásadní vliv na rychlost měl způsob stahování a limitace způsobená rychlostí zápisu na disk. Při vyšší rychlosti zapisování na disk by dosažitelná rychlost stahování byla také vyšší. Šifrátory ani webový server provoz nelimitovaly.

4.1.2 NGINX

NGINX je příkladem komplexnějšího webového serveru. Jedná se o open-source server podporující kromě HTTP/HTTPS i vyrovnávání zátěže, caching, mail proxy a další[29]. Tento webový server lze také optimalizovat pro danou činnost. V rámci práce byl server NGINX částečně optimalizován změnami parametrů, např. `sendfile _max_chunk 512k`, konfiguračního souboru `/etc/nginx/nginx.conf`[30]. Vliv těchto změn na výkon byl však pro komunikaci jednoho klienta se serverem zanedbatelný. Dosažené rychlosti stahování pro webový server NGINX prezentuje tabulka č. 4.2.

Tab. 4.2: Rychlosti stahování souborů z webserveru – NGINX.

Linka	Před optimalizací	Po optimalizaci
Juniper SRX4200	550 MB/s	1200 MB/s
šifrátory FPGA	550 MB/s	2100 MB/s

Výsledky byly stejné jako pro Python HTTP. Shodné výsledky byly způsobeny limitující zapisovací rychlostí na disk v obou případech. Limitujícím faktorem byl tedy aktuální hardware, nikoli vytížení RAM či CPU, schopnosti webových serverů nebo schopnosti šifrátorů. Pro vícero současných komunikací mezi více klienty a serverem NGINX by se začal projevovat vliv optimalizace a v této situaci by byl pravděpodobně výkonnější, než Python HTTP.

4.2 Video Streaming

Pro ověření, zda hodnota ztrátovosti při provozu přes šifrátory je přijatelná i při skutečném provozu, byl otestován přenos videa v reálném čase. Tento typ provozu není příliš náchylný na latenci, jelikož video je přehráváno s drobným zpožděním a je využívána vyrovnávací paměť. Například proměnlivé zpoždění, tj. jitter, je tak téměř eliminováno. Jednalo se tak o ověření především hodnot ztrátovosti.

Přenos videa byl realizován pomocí přehrávače VLC. Tento přehrávač umožňuje jednoduše vysílat v rámci interní sítě. Na jedné stanici tedy bylo nakonfigurováno vysílání na adresu `http://<IP_adresa>:8080`, a zadáním dané adresy do přehrávače na druhé stanici bylo vysílané video přehráváno. Video bylo vysílané z úložiště na virtuální stanici a mělo datový tok o rychlosti 4280 kbit/s. Pro využití větší šířky pásma byly dále realizovány dva přenosy současně. Všechna videa zobrazené na stanici klienta byla bez viditelného zkreslení obrazu. Rozsáhlejší testy by vyžadovaly infrastrukturu s dedikovanými grafickými kartami. Pro tyto testy byla využívána iGPU (Integrated Graphics Processing Unit), která byla součástí procesoru.

Po domluvě s vedoucím práce další testy provozu v reálném čase, zahrnující VoIP apod., neproběhly. Důvodem byla nemožnost práce se zvukem na virtuálních stanicích kvůli absenci fyzické zvukové karty v testeru. Snaha o zprovoznění zvuku zahrnovala implementaci různých virtuálních zvukových karet nebo využití vzdálené plochy. Tento problém se v daném časovém horizontu nepodařilo vyřešit a testy VoIP také nebyly prioritou zadání. Proto byla pozornost věnována dalším oblastím práce.

4.3 Vysokorychlostní přepínač

Poslední testy ověřující funkcionalitu šifrátorů v reálných situacích zahrnovaly přidání vysokorychlostního přepínače do 100 Gbit/s linky. Tyto testy měly za cíl zjistit, zda bude nějakým způsobem ovlivněn síťový provoz, pokud se nebude jednat o přímé propojení testovaných prvků. Zapojení dalších síťových prvků do linky představuje reálnější situaci, kdy může docházet k působení dalších vlivů například způsobující limitaci provozu. Přidaný 100G přepínač byl Mikrotik CRS518-16XS-2XQ-RM. Testování tedy probíhalo s FPGA. Topologii zapojení prezentuje obr. č. 4.1.

Obr. 4.1: Topologie pro testování s šifrátory na platformě FPGA + Switch.

Provoz šifrovaný šifrátory Juniper byl výrazně nižší a nevytěžoval by přepínač dostatečně, zároveň nebyl předpokládán výrazný vliv přepínače. Testy propustnosti v této části tak firewally Juniper SRX4200 nezahrnovaly. Pro šifrátory FPGA však přidání přepínače do linky představovalo výzvu, protože provoz procházející přes přepínač opět způsoboval desynchronizaci klíčů na kartách. To bylo vyřešeno dalšími průběžnými opravami a aktualizacemi firmware.

Postup při realizaci testů propustnosti byl totožný s předchozím testováním, viz podkapitola „Platforma FPGA“ 3.2.1. Výsledky TCP provozu zobrazuje tab. č. 4.3.

Tab. 4.3: Provoz přes switch, TCP – FPGA Agilex (Intel).

Počet CPU jader	MTU [B]		
	576	1300	1500
1	13,7 ± 0,6 Gbit/s	21,7 ± 2,1 Gbit/s	25,4 ± 1,1 Gbit/s
10	42,2 ± 1,7 Gbit/s	72,6 ± 2,0 Gbit/s	80,0 ± 0,1 Gbit/s

Nejvyšší dosažená propustnost byla 80 Gbit/s. V případě propustnosti bez přepínače se jednalo o 85 Gbit/s. Přepínač tedy mírně limitoval provoz. Maximální pozorovaný rozdíl hodnot bez a s přepínačem byl 8,7 Gbit/s pro desetijádrový výkon a hodnotu MTU 1300 B. V ostatních případech byla pozorována limitace ve výši <5 Gbit/s. Výsledky testů propustnosti UDP provozu zobrazuje tab. č. 4.4.

Tab. 4.4: Provoz přes switch, UDP – FPGA Agilex (Intel).

Počet CPU jader	MTU [B]		
	576	1300	1500
1	1,1 ± 0,1 Gbit/s	2,4 ± 0,1 Gbit/s	3,2 ± 0,2 Gbit/s
10	9,1 ± 0,3 Gbit/s	20,1 ± 1,0 Gbit/s	22,9 ± 0,9 Gbit/s

Pro UDP provoz byla maximální limitace způsobená přepínačem 3,6 Gbit/s. Snížení propustnosti tak bylo podobné, jako pro TCP provoz. Nebyl pozorován rozdíl na hodnotě ztrátovosti při provozu přes přepínač v porovnání s přímým propojením.

Přidání přepínače mělo za následek mírnou limitaci provozu v řádu jednotek Gbit/s. Další vlivy v podobě vyšších hodnot ztrátovosti apod. pozorovány nebyly. Daný přepínač byl schopen pracovat také na síťové vrstvě, což by umožňovalo testování se směrováním provozu. Tato možnost však otestována nebyla kvůli rychlosti samotného směrování. Výsledky by byly zkreslené schopnostmi přepínače, protože přepínač nebyl pro směrování provozu o daných rychlostech uzpůsoben.

Závěr

Cílem bakalářské práce bylo navrhnout a sestavit síťovou infrastrukturu a následně realizovat výkonové testy pro postkvantové šifrátory. Síť byla realizována za pomoci virtualizace v podobě hypervizora XCP-ng a prostředí Xen Orchestra. Zde byly vytvořeny dvě virtuální stanice, které představovaly koncové body komunikace a mezi kterými byly utvářeny linky pro síťovou komunikaci. Do těchto linek byly postupně zapojovány testované prvky. Testy byly realizovány pro rychlosti až 100 Gbit/s.

Úvodní testování bylo zaměřené na dosažení maximální propustnosti 100G linky. V této fázi byly provedeny kroky optimalizace, které umožnily dosažení výrazně vyšších hodnot. Nejdůležitějšími kroky optimalizace byla změna síťových parametrů v systémové konfiguraci stanic, povolení PCI passthrough pro síťové karty a zapnutí offloading mechanismů. Po optimalizaci byla průměrná hodnota maximální propustnosti 96,4 Gbit/s (TCP) při velikosti MTU 9000 B. Maximální propustnost před optimalizací byla pouze ∼30 Gbit/s. Jednalo se o provoz generovaný pomocí iPerf2. Vliv optimalizace byl pozorován také na hodnotě latence. Latence byla testovaná pomocí nástrojů `ping` a `traceroute` a její průměrná hodnota klesla na třetinu.

Další postup zahrnoval testy propustnosti nešifrovaného a šifrovaného TCP a UDP provozu, které byly provedeny pro různý počet jader CPU. Testy byly realizovány pro 4 různé velikosti MTU – 576 B, 1300 B, 1500 B a 9000 B. Různé kategorie testů pro daná MTU měly sloužit jako referenční hodnoty pro různé situace, pokud např. zařízení nebudou podporovat větší velikosti (9000 B) nebo dojde ke snížení velikosti vlivem enkapsulace (1300 B) apod. Pro běžnou velikost MTU – 1500 B byla průměrná propustnost 85,4 Gbit/s. Jednalo se o maximální výkon s využitím všech CPU jader. V této části také proběhla analýza síťového provozu pomocí nástroje Wireshark, kdy bylo ověřeno, že komunikace skutečně probíhala dle interpretace iPerf. Zároveň byl blíže pozorován vliv optimalizace, kdy data byla zasílána ve velikosti 65226 B segmentů (dle parametru v konf. souboru) a dělení do velikostí jednotlivých paketů probíhalo až na síťové kartě.

Testování šifrovaného provozu bylo provedeno za pomoci šifrátorů na platformě FPGA a komerčně dostupných šifrátorů, jež byly součástí firewallů Juniper SRX4200. V rámci práce byly testovány dva různé šifrátory FPGA zahrnující Xilinx UltraScale+ a Agilex hardware. Zde byly testované kategorie MTU přizpůsobeny schopnostem šifrátorů. Tyto testy byly zaměřené především na nejvyšší výkon. Při velikosti MTU 1500 B přesahovala propustnost 80 Gbit/s pro oba typy hardware. Limitace provozu samotnými šifrátory tak byla minimální. Pro FPGA Agilex (Intel) byl dále testován i režim QKD, který žádnou další limitaci nezpůsoboval a propustnost byla stejná, jako v předchozím případě. Dále byly realizovány dlouhodobé testy o době trvání 2, 6 a 20 hodin pro ověření funkčnosti šifrátorů v delších časových intervalech.

Tyto testy proběhly úspěšně pro režimy provozu s i bez QKD. Firewally Juniper disponovaly pouze 10G síťovým rozhraním, proto byly testy nešifrovaného provozu realizovány pro tuto linku samostatně. Pro tuto linku nešifrovaný provoz dosahoval maxima v podobě 9,9 Gbit/s. Testování šifrovaného provozu zahrnovalo testy propustnosti pro různé kombinace parametrů IPsec/VPN a šifrovacích algoritmů, které prvky nabízely. Tyto testy měly za cíl nalézt vhodnou kombinaci pro kvantově odolný kanál, která by zároveň nabízela vysoký výkon. To splňovala kombinace IPsec s šifrou AES-256-GCM a IKEv2 s hashovacím algoritmem SHA-256 a PSK. Při této kombinaci bylo dosaženo maximální propustnosti pro danou linku, tedy 9,8 Gbit/s pro MTU 9000 B. Vybraný šifrovací algoritmus byl tedy stejný pro FPGA i Junipery. Testované MTU byly opět přizpůsobeny schopnostem šifrátorů a i při těchto testech byl opět využit nástroj iPerf pro generování síťového provozu.

Protože všechny dosavadní testy byly realizovány za pomoci uměle generovaného síťového provozu, byla v další části testována funkčnost šifrátorů se skutečným provozem. Jednalo se o webovou komunikaci v podobě stahování souborů, živý přenos videa a provoz přes další síťový prvek, kterým byl vysokorychlostní přepínač. Pro webovou komunikaci byly na jedné stanici implementovány dva webové servery – Python HTTP a NGINX. Dva různé webové servery byly zvoleny kvůli různé komplexnosti a funkčnosti, vliv nebo limitace samotnými servery však při daných testech pozorovány nebyly. Rychlost stahování souborů byla limitována rychlostí zápisu na disk klienta a proto byla využita část RAM jako další úložiště. Maximální rychlost zápisu na RAM disk byla 2,1 GB/s a jednalo se tak o maximum pro daný hardware. Při stahování souborů bylo maximální rychlosti dosaženo pro oba webové servery při paralelním stahování dvou souborů pomocí nástroje `wget2`. Rychlost stahování 2,1 GB/s platila pro FPGA, pro linku s šifrátory Juniper byla rychlost limitována schopnostmi linky, rychlost tedy dosahovala ∼1,2 GB/s neboli ∼9,6 Gbit/s. Živý přenos videa sloužil pro ověření schopností šifrátorů při provozu v reálném čase. Přenos videa byl realizován pomocí VLC z jedné stanice na druhou po požadované lince. Dále byly spuštěny dva přenosy současně pro využití větší šířky pásma. Pro všechna zobrazená videa nebylo pozorováno žádné zkreslení obrazu způsobeno přebytečnou ztrátovostí aj. vlivy. Testování s dalším síťovým prvkem zahrnovalo zapojení přepínače do 100G linky. Tyto testy byly realizovány pouze pro FPGA kvůli vyšším dosažitelným rychlostem a byly opět v podobě testů propustnosti, přičemž byly zkoumány limitující vlivy způsobené přepínačem. Pro velikost MTU 1500 B byla průměrná propustnost 80,0 Gbit/s, tedy o jednotky Gbit/s méně v porovnání provozu bez přepínače. Další vlivy způsobené přepínačem pozorovány nebyly.

Podnětem pro další práci s šifrátory by mohlo být rozsáhlejší testování, např. využití přepínače pro zapojení více linek současně, nebo využití šifrátorů ve reálné síti.

Literatura

[1] FARMER, J.; LANE, B.; BOURG, K.; WANG, W. *Transmission Control Protocol; FTTx Networks: Chapter 12 - IPTV Architecture*. 2016. ISBN: 978-0-12-420137-8. Online. Dostupné z: https://www.sciencedirect.com/topics/computer-science/transmission-control-protocol. [cit. 2024-12-01].

[2] EDDY, W. *RFC 9293: Transmission Control Protocol*. ISSN: 2070-1721. IETF Datatracker. Online. Dostupné z: https://datatracker.ietf.org/doc/html/rfc9293#name-managing-the-window. [cit. 2024-12-01].

[3] PATEL, D. *How to Get Maximum Throughput with a 100 Gb Ethernet Adapter in a Non-Virtualized Environment*. IBM. 2020. Online. Dostupné z: https://www.ibm.com/support/pages/how-get-maximum-throughput-100-gb-ethernet-adapter-non-virtualized-environment. [cit. 2024-12-01].

[4] JACOBSON, V.; BRADEN, R.; BORMAN, D. *RFC 1323: TCP Extensions for High Performance*. Online. IETF Datatracker. Dostupné z: https://datatracker.ietf.org/doc/html/rfc1323. [cit. 2024-12-01].

[5] ČESKÝ TELEKOMUNIKAČNÍ ÚŘAD. *Měření datových parametrů sítí pomocí TCP protokolu*. 2014. Online. Dostupné z: https://ctu.gov.cz/sites/default/files/obsah/stranky/937/soubory/merenidatovychparametrusitipomocitcpprotokolu.pdf. [cit. 2024-12-01].

[6] POSTEL J. *RFC 768: User Datagram Protocol*. Online. IETF Datatracker 1980, Dostupné z: https://datatracker.ietf.org/doc/html/rfc768. [cit. 2024-12-01].

[7] STAPKO T. *User Datagram Protocol; How TCP/IP and Ethernet Work. Practical Embedded Security: CHAPTER 2 - Network Communications Protocols and Built-in Security*. Online. ScienceDirect. Dostupné z: https://www.sciencedirect.com/topics/computer-science/user-datagram-protocol. [cit. 2024-12-01].

[8] SHINDER, D.; SHINDER, T.; TODD, C.; HUNTER, L. *Internet Protocol; MCSA/MCSE (Exam 70-291) Study Guide: Chapter 1 - MCSA/MCSE 70-291: Reviewing TCP/IP Basics*. ISBN: 978-1-931836-92-0. 2003. Online. Dostupné z: https://www.sciencedirect.com/book/9781931836920/mcsa-mcse-exam-70-291-study-guide. [cit. 2024-12-01].

[9] IETF DATATRACKER. *RFC 791: Internet Protocol; Section 3.1 - Internet Header Format*. 1981. Online. Dostupné z: https://datatracker.ietf.org/doc/html/rfc791#section-3.1. [cit. 2024-12-01].

[10] LAKHINA A.; PAPAGIANNAKI K.; CROVELLA M.; DIOT C.; KOLACZYK E.; TAFT N. *Structural Analysis of Network Traffic Flows*. Online. Dostupné z: https://dl.acm.org/doi/abs/10.1145/1005686.1005697. [cit. 2024-12-01].

[11] VELAYUDHAN, D. *Network infrastructure design: planning and implementation*. Connectwise. 2023. Online. Dostupné z: https://www.connectwise.com/blog/business-continuity/network-infrastructure-design-best-practices. [cit. 2024-12-01].

[12] CHEN, Y.; TIAN, B.; TIAN, C.; DAI, L.; ZHOU, Y.; MA, M.; TANG, M.; ZHENG, H.; YANG, Z.; CHEN, G.; CAI, D.; ZHAI, E. *Norma: Towards Practical Network Load Testing*. USENIX. 2023. Online. Dostupné z: https://www.usenix.org/system/files/nsdi23-chen-yanqing.pdf. [cit. 2024-12-01].

[13] IPERF. *iPerf Documentation*. Online. Dostupné z: https://iperf.fr/iperf-doc.php. [cit. 2024-12-01].

[14] LAMBERTI, A. *Network Load Testing and Network Load Balancing*. Obkio. 2023. Online. Dostupné z: https://obkio.com/blog/network-load-testing-and-network-load-balancing/. [cit. 2024-12-01].

[15] XCP-NG. *XCP-ng 8.3 Release Notes: PCI passthrough support in the API*. 2024. Online. Dostupné z: https://docs.xcp-ng.org/releases/release-8-3/#pci-passthrough-support-in-the-api. [cit. 2024-12-01].

[16] FASTERDATA. *Performance Expectations for a 100G Host*. 2024. Online. Dostupné z: https://fasterdata.es.net/performance-testing/performance-expectations-for-a-100g-host/. [cit. 2024-12-01].

[17] CONSTANTINE, B.; FORGET, G.; GEIB, R. SCHRAGE, R. *RFC 6349: Framework for TCP Throughput Testing: Chapter 3 - Methodology, Section 3.2.1 - Measuring RTT*. IETF Datatracker. 2011. Online. Dostupné z: https://datatracker.ietf.org/doc/html/rfc6349#section-3.2. [cit. 2024-12-01].

[18] FASTERDATA. *Linux Tuning*. Online. Dostupné z: https://fasterdata.es.net/host-tuning/linux/. [cit. 2024-12-01].

[19] FASTERDATA. *IOMMU*. Online. Dostupné z: https://fasterdata.es.net/host-tuning/linux/100g-tuning/iommu/. [cit. 2024-12-01].

[20] P. DHARANISH, S. KOKILA, M. MYTHILY, A. BALACHANDRAN. *FPGA implementation of post quantum cryptography for high performance*. Online. Dostupné z: https://www.researchgate.net/publication/385963847_FPGA_implementation_of_post_quantum_cryptography_for_high_performance. [cit. 2025-04-20].

[21] CÍBIK, P.; DOBIÁŠ, P.; RICCI, S.; HAJNÝ, J.; MALINA, L.; JEDLIČKA, P.; SMÉKAL, D. *Pushing AES-256- GCM to Limits: Design, Implementation and Real FPGA Tests. In Lecture Notes in Computer Science - Applied Cryptography and Network Security Workshops. Lecture Notes in Computer Science*. 2024. s. 303-318. ISBN: 978-3-031-61486-6. ISSN: 0302-9743 [cit. 2024-04-20].

[22] SERVERFAULT. *Why is UDP Slower than TCP on Ubuntu Server?*. 2012. Online. Dostupné z: https://serverfault.com/questions/432101/why-is-udp-slower-than-tcp-on-ubuntu-server. [cit. 2024-12-01].

[23] HOCK, M.; VEIT, M.; NEUMEISTER, F.; BLESS, R.; ZITTERVART, R. *TCP at 100 Gbit/s – Tuning, Limitations, Congestion Control*. IEEE Xplore. 2020. Online. Dostupné z: https://ieeexplore.ieee.org/document/8990842. [cit. 2024-12-01].

[24] JUNIPER NETWORKS. *SRX4100 AND SRX4200 FIREWALLS DATASHEET*. Juniper Networks, Inc. Online. Dostupné z: https://www.juniper.net/content/dam/www/assets/datasheets/us/en/security/srx4100-srx4200-firewall-datasheet.pdf. [cit. 2025-02-22].

[25] FLUHRER, S.; KAMPANAKIS, P; MCGREW, D.; SMYSLOV, V. *RFC 8784: Mixing Preshared Keys in the Internet Key Exchange Protocol Version 2 (IKEv2) for Post-quantum Security*. IETF Datatracker. 2020. Online. Dostupné z: https://datatracker.ietf.org/doc/html/rfc8784. [cit. 2025-02-22].

[26] JUNIPER NETWORKS. *Documentation - Junos OS: Junos CLI Reference - tunnel-mtu (Services IPsec VPN)*. Juniper Networks, Inc. Online. Dostupné z: https://www.juniper.net/documentation/us/en/software/junos/cli-reference/topics/ref/statement/tunnel-mtu-edit-services.html. [cit. 2025-05-18].

[27] GUERON S.; JOHNSON S.; WALKER J. *SHA-512/256: Conclusion*. s. 6. Online. Dostupné z: https://eprint.iacr.org/2010/548.pdf. [cit. 2025-05-18].

[28] ZACZYŃSKI, B. *How to Launch an HTTP Server in One Line of Python Code*. RealPython. Online. Dostupné z: https://realpython.com/python-http-server/. [cit. 2025-03-17].

[29] SOLARWINDS. *What Is NGINX?*. Online. Dostupné z: https://www.papertrail.com/solution/guides/nginx/. [cit. 2025-03-21].

[30] AMD. *NGINX Tuning Guide for AMD EPYC 7002 Series Processors: Chapter 5 - NGINX Tuning Methodology and Recommendations, Section 5.5 - NGINX Parameters*. 2020. Online. Dostupné z: https://www.amd.com/content/dam/amd/en/documents/epyc-technical-docs/tuning-guides/amd-epyc-7002-tg-nginx.pdf. [cit. 2025-03-21].

[31] XIAO, G. *How to Easily Create RAM Disk on Debian, Ubuntu, Linux Mint, CentOS*. 2024. LinuxBabe. Online. Dostupné z: https://www.linuxbabe.com/command-line/create-ramdisk-linux. [cit. 2025-03-21].

[32] ALEXANDER, Z. *How to measure I/O performance of ramdisk (/dev/ram0)*. Serverfault. 2015. Online. Dostupné z: https://serverfault.com/a/706075. [cit. 2025-03-21].

Seznam symbolů a zkratek

ACK	Acknowledgement
AES	Advanced Encryption Standard
AH	Authentication Header
ARP	Address Resolution Protocol
BIOS	Basic Input-Output System
CPU	Central Processing Unit
DNS	Domain Name System
ESP	Encapsulating Security Payload
FPGA	Field-Programmable Gate Array
FTP	File Transfer Protocol
FQ	Fair Queueing
GCM	Galois/Counter Mode
GRO	Generic Recieve Offload
ICMP	Internet Control Message Protocol
ID	Identifier
iGPU	Integrated Graphics Processing Unit
IKEv2	Internet Key Exchange Protocol Version 2
IOMMU	Input-Output Memory Management Unit
IP	Internet Procotol
IPsec	Internet Protocol Security
MTU	Maximum Transmission Unit
NIC	Network Interface Card
OS	Operating System
PCI	Peripheral Component Interconnect

PSK	Pre-Shared Keys
QKD	Quantum Key Distribution
RAM	Random Access Memory
RTT	Round-Trip Time
SFF	Small Form Factor
TCP	Transmission Control Procotol
TSO	TCP Segmentation Offload
UDP	User Datagram Procotol
VPN	Virtual Private Network
VT-d	Virtualization Technology for Direct Input/Output

A Protokol z měření

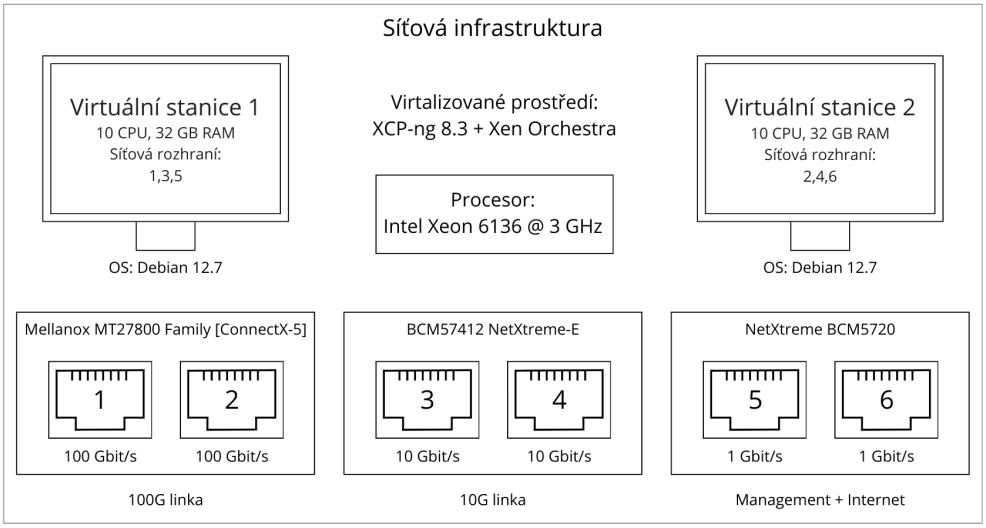
1 Implementace testbedu, testované prvky a metodika měření

1.1 Úvod

Protokol shrnuje postup a výsledky získané při výkonových testech postkvantových šifrátorů. Testované prvky zahrnovaly šifrátory založené na platformě FPGA, které byly vyvíjeny na VUT FEKT, a srovnání s komerčně dostupnými firewally Juniper SRX4200. Testování na platformě FPGA zahrnovalo dva různé typy šifrátorů – Xilinx UltraScale+ a Agilex hardware. Výkonové testy byly realizovány především v podobě testů propustnosti, přičemž teoretické maximum bylo 100 Gbit/s pro FPGA a 10 Gbit/s pro Junipery. Schopnosti šifrátorů byly dále ověřeny v dlouhodobých testech a v reálných aplikacích.

1.2 Praktická implementace testbedu

Pro praktické řešení byla sítě využita virtualizace. Virtualizované prostředí zahrnovalo hypervizora XCP-ng 8.3 s grafickým rozhraním Xen Orchestra. Zde byly vytvořeny dvě virtuální stanice, které představovaly koncové body komunikace a mezi kterými byly utvářeny linky pro síťovou komunikaci. Do těchto linek byly postupně zapojovány testované prvky. Každé stanici bylo přiděleno 10 jader CPU (Intel Xeon 6136), 32 GB RAM a příslušná síťová rozhraní. Tester obsahoval více síťových karet, každou disponující dvěma rozhraními, což umožňovalo realizaci spoje mezi virtuálními stanicemi o daných rychlostech. Realizaci testeru blíže prezentuje obr. č. 1.1.

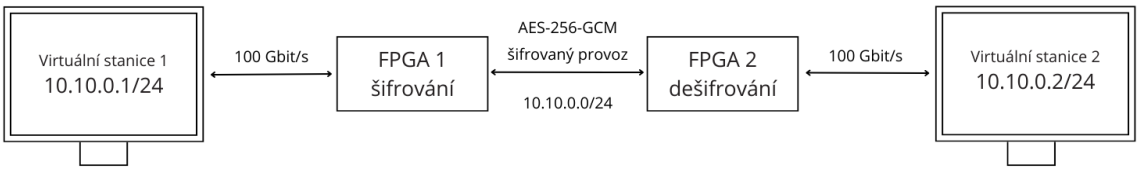


Obr. 1.1: Vizualizace topologie pro testování s šifrátory na platformě FPGA.

1.3 Testované prvky

1.3.1 Platforma FPGA

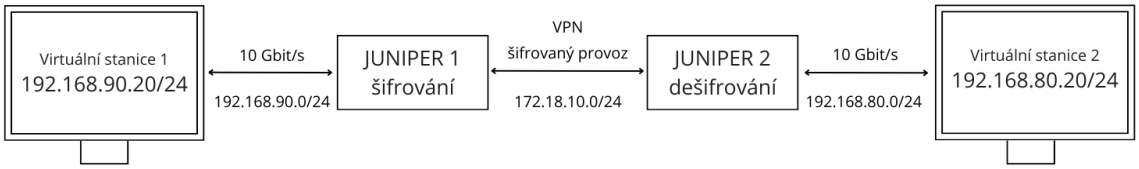
Platforma FPGA nabízí flexibilní a efektivní řešení pro různé výpočetní úlohy, jelikož ji lze programovat přímo pro danou činnost. Efektivita platformy spočívá v akceleraci a paralelním zpracování s využitím hardwarových obvodů. Testované šifrátory FPGA byly vyvíjeny na VUT FEKT a pro šifrovací algoritmus byl využit AES v módu GCM a délkou klíče 256 bitů. Šifra AES-256-GCM nabízí autentizované šifrování dat, je vhodná pro vysoké propustnosti a zároveň je také kvantově odolná. V rámci práce byly testovány dva různé typy šifrátorů zahrnující Xilinx UltraScale+ a Agilex harware. Realizace 100G linky proběhla s využitím síťové karty Mellanox MT27800 ConnectX-5. Předpokladem pro výsledky šifrovaného provozu pomocí FPGA byla žádná či minimální limitace oproti hodnotám propustnosti provozu bez šifrování. Topologii pro testování propustností šifrátorů FPGA zobrazuje obr. č. 1.2.



Obr. 1.2: Vizualizace topologie pro testování s šifrátory na platformě FPGA.

1.3.2 Firewally Juniper SRX4200

Firewally Juniper SRX4200 jsou komerční síťové prvky. Podporují standard RFC 8784, což umožňuje realizovat kvantově bezpečné spojení pomocí IPsec a IKEv2. Firewally SRX4200 disponují 10G rozhraním, testování tedy probíhalo na samostatné lince o teoretické rychlosti 10 Gbit/s. V případě testeru se jednalo o rozhraní 10G síťové karty Broadcom BCM57412. Pro zhotovení testů byly využity dva firewally, mezi nimiž byl ustanoven kvantově odolný VPN spoj. Topologii pro testování propustností s šifrátory Juniper prezentuje obr. č. 1.3.



Obr. 1.3: Vizualizace topologie pro testování s šifrátory Juniper.

1.4 Metodika testování

Testování propustnosti bylo realizováno za pomoci nástroje iPerf. Nástroj iPerf nabízí různé verze, díky kterým lze testovat jednovláknový výkon (iPerf3) nebo vícevláknový výkon (iPerf2). Při realizaci výkonových testů byl iPerf používán v obou verzích. Přidělení konkrétního počtu jader procesu `iperf` při testech, které to vyžadovaly, bylo realizováno pomocí nástroje `taskset`. Úvodní testy nešifrovaného provozu zahrnovaly testy pro různé velikosti MTU a počty užitých CPU jader. Velikosti MTU byly zvoleny následovně:

- 576 B – minimální velikost IPv4 paketu, může sloužit jako referenční hodnota, pokud by docházelo k fragmentaci,
- 1300 B – referenční velikost pro minimální velikost IPv6 paketů (1280 B) nebo šifrovaný provoz, kdy se mohou enkapsulací přidávat nová záhlaví apod.,
- 1500 B – běžná hodnota pro Ethernet/Ethernet II (1518 B),
- 9000 B – používané ve vysokorychlostních infrastrukturách.

Doba trvání jednotlivých testů byla 30 sekund. Tato délka byla zvolena kvůli dostatečné vypovídající hodnotě o aktuální dosažitelné propustnosti. Každý test byl opakován minimálně třikrát, čímž byla ověřena jednoznačnost hodnot. Pro každou kombinaci MTU, počtu jader CPU a typu provozu byla nejprve nalezena strukura příkazu, kterou bylo dosaženo maximálního vytížení CPU jader tak, aby zároveň nedocházelo k přetížení. Pro každou linku byly nejprve realizovány testy nešifrovaného provozu – přímé propojení virtuálních stanic. Testy šifrovaného provozu zahrnovaly postupné zapojování testovaných prvků do dané linky. Tyto testy byly zaměřeny především na nejvyšší výkon (využití všech CPU jader). Pro referenci byly testy šifrovaného provozu provedeny i pro jednojádrový výkon a výsledné hodnoty těchto testů jsou uvedeny v tabulkách spolu s hodnotami maximálního výkonu. Hodnoty šifrovaného provozu jsou zároveň v podobě průměrné hodnoty, vypočtené ze všech provedených testů, s odchylkou, která značí získanou hodnotu, která byla nejvíce vzdálená a průměrné. Testované kategorie MTU byly v průběhu testování přizpůsobovány schopnostem šifrátorů. Změny MTU byly prováděny na virtuálních stanicích příkazem `sudo ip link set dev <interface> <velikost>`.

Hodnota latence byla testována pomocí nástrojů `ping` a `traceroute`. Analýza provozu probíhala zachycením provozu v programu Wireshark v prostředí virtuálních stanic.

2 Realizace zátěžových testů

2.1 Platforma FPGA

V úvodní fázi byly zhotoveny testy propustnosti nešifrovaného provozu 100G linky. Testy zahrnují hodnoty rychlostí pro různá CPU jádra a velikosti MTU. Následující dvě tabulky prezentují výsledky pro TCP a UDP provoz po provedení optimalizace.

Tab. 2.1: Výsledky testů propustnosti pro nešifrovaný TCP provoz.

Počet CPU jader	MTU [B]			
	576	1300	1500	9000
1	14,0 Gbit/s	26,0 Gbit/s	30,6 Gbit/s	32,4 Gbit/s
2	24,4 Gbit/s	39,4 Gbit/s	46,5 Gbit/s	55,1 Gbit/s
4	32,1 Gbit/s	63,6 Gbit/s	64,7 Gbit/s	69,1 Gbit/s
6	34,0 Gbit/s	72,8 Gbit/s	76,0 Gbit/s	92,1 Gbit/s
8	36,1 Gbit/s	78,4 Gbit/s	82,4 Gbit/s	96,1 Gbit/s
10	38,9 Gbit/s	81,7 Gbit/s	85,4 Gbit/s	96,4 Gbit/s

Pro běžnou velikost MTU bylo dosaženo 85 Gbit/s s využitím všech jader, která byla k dispozici. Pro velikost MTU 9000 B bylo dosaženo 96,4 Gbit/s – téměř teoretického maxima. Dále byly realizovány testy UDP provozu.

Tab. 2.2: Výsledky testů propustnosti pro nešifrovaný UDP provoz.

Počet CPU jader	MTU [B]			
	576	1300	1500	8000
1	1,3 Gbit/s	3,1 Gbit/s	3,6 Gbit/s	15,7 Gbit/s
2	2,3 Gbit/s	5,9 Gbit/s	5,8 Gbit/s	28,9 Gbit/s
4	4,5 Gbit/s	10,6 Gbit/s	11,5 Gbit/s	52,6 Gbit/s
6	6,3 Gbit/s	15,1 Gbit/s	16,9 Gbit/s	75,2 Gbit/s
8	7,8 Gbit/s	18,6 Gbit/s	22,7 Gbit/s	95,2 Gbit/s
10	9,4 Gbit/s	21,4 Gbit/s	24,4 Gbit/s	∼100,0 Gbit/s

Propustnost UDP provozu byla tedy výrazně nižší v porovnání s TCP provozem. Pro UDP provoz byla největší testovaná velikost MTU 8000 B, protože při užití všech deseti jader procesoru již rychlost přesahovala teoretické maximum. Při testech s většími MTU bylo zjevné zvýšení propustnosti díky nižší frekvenci generování datagramů. Nedocházelo tak k zahlcení procesoru, takže byly pro 8 a 10 jader, při MTU 8000 B, rychlosti srovnatelné s hodnotami TCP provozu.

Průběh testování TCP provozu spolu s užitým příkazem pro dosažení maximální propustnosti osmijádrového výkonu a velikost MTU 9000 B zobrazuje obr. č. 2.1.

```
fridrich@debian2: ~
fridrich@debian2: ~        fridrich@debian2: ~
[  6] 29.0000-30.0000 sec   887 MBytes  7.44 Gbits/sec
[  9] 29.0000-30.0000 sec   768 MBytes  6.44 Gbits/sec
[  4] 29.0000-30.0000 sec   916 MBytes  7.68 Gbits/sec
[  1] 29.0000-30.0000 sec  1.02 GBytes  8.78 Gbits/sec
[ 12] 29.0000-30.0000 sec  1.13 GBytes  9.68 Gbits/sec
[ 10] 29.0000-30.0000 sec   905 MBytes  7.59 Gbits/sec
[  7] 29.0000-30.0000 sec   894 MBytes  7.50 Gbits/sec
[ 11] 29.0000-30.0000 sec   932 MBytes  7.82 Gbits/sec
[  2] 29.0000-30.0000 sec   924 MBytes  7.75 Gbits/sec
[SUM] 29.0000-30.0000 sec  11.2 GBytes  96.0 Gbits/sec
[  3] 0.0000-30.0076 sec  28.1 GBytes  8.05 Gbits/sec
[  8] 0.0000-30.0068 sec  26.3 GBytes  7.52 Gbits/sec
[  6] 0.0000-30.0066 sec  25.1 GBytes  7.18 Gbits/sec
[  4] 0.0000-30.0062 sec  27.5 GBytes  7.87 Gbits/sec
[ 12] 0.0000-30.0054 sec  34.6 GBytes  9.90 Gbits/sec
[  7] 0.0000-30.0046 sec  26.8 GBytes  7.67 Gbits/sec
[  2] 0.0000-30.0075 sec  28.3 GBytes  8.10 Gbits/sec
[  5] 0.0000-30.0068 sec  30.7 GBytes  8.79 Gbits/sec
[  1] 0.0000-30.0059 sec  28.1 GBytes  8.04 Gbits/sec
[ 11] 0.0000-30.0039 sec  28.0 GBytes  8.01 Gbits/sec
[  9] 0.0000-30.0064 sec  24.2 GBytes  6.93 Gbits/sec
[ 10] 0.0000-30.0052 sec  28.0 GBytes  8.03 Gbits/sec
[SUM] 0.0000-30.0025 sec   336 GBytes  96.1 Gbits/sec
fridrich@debian2:~$ taskset -c 0-7 iperf -c 10.10.0.1 -i 1 -P 12 -t 30 -w 256k
```

Obr. 2.1: Struktura příkazu iPerf a průběžné hodnoty propustnosti TCP – 8 CPU.

Důležitými parametry pro dosažení těchto propustností byly optimální počet proudů `-P` a velikost TCP okna `-w`. V rámci testování proběhla také analýza provozu programem Wireshark za účelem ověření, že komunikace probíhala dle interpretace iPerf. Provoz zachycený programem Wireshark prezentuje obr. č. 2.2

No.	Time	Source	Destination	Protocol	Length	Info
94192	5.998937558	10.10.0.2	10.10.0.1	TCP	65226	44012 → 5001 [PSH, ACK] Se
94193	5.999094542	10.10.0.2	10.10.0.1	TCP	65226	[TCP Previous segment not
94194	5.999125295	10.10.0.2	10.10.0.1	TCP	65226	44012 → 5001 [PSH, ACK] Se
94195	5.999128804	10.10.0.1	10.10.0.2	TCP	66	[TCP ACKed unseen segment]
94196	5.999163565	10.10.0.2	10.10.0.1	TCP	1626	44012 → 5001 [PSH, ACK] Se
94197	5.999164526	10.10.0.2	10.10.0.1	TCP	65226	44012 → 5001 [PSH, ACK] Se
94198	5.999168348	10.10.0.1	10.10.0.2	TCP	66	5001 → 43990 [ACK] Seq=29

Obr. 2.2: Ukázka testovacího TCP provozu s maximálním vytížením.

Celková délka zasílaných segmentů byla 65226 B. Jednalo se o vliv offload mechanismů, kdy dělení segmentů do velikosti jednotlivých paketů probíhalo až na úrovni síťové karty. Rozdílné porty identifikovaly různé paralelní proudy, jejichž počet byl specifikován v `iperf` příkazu.

Testování UDP provozu probíhalo obdobně. Při testech UDP provozu bylo nutné v příkazu určit typ provozu `-u`, šířku pásma `-b 100G`, optimální počet proudů `-P` a správnou velikost datagramu `-l` pro dosažení maximální propustnosti. Hodnota

latence 100G linky, testovaná pomocí příkazu `ping`, se pro 10+ dotazů a opakovaná měření, pohybovala v rozmezí 0,231–0,288 ms před i po přidání šifrátorů do linky.

Po zhotovení testů nešifrovaného provozu 100G linky byly dále realizovány testy šifrovaného provozu. Výsledky testů propustnosti jsou prezentovány v následujících čtyřech tabulkách. Testy byly zaměřeny na nejvyšší výkon – užití všech 10 CPU jader. Pro referenci byly realizovány také testy jednojádrového výkonu. Nejvyšší testovaná velikost MTU je 1500 B, šifrátory mají velikost MTU implicitně nastavenou na hodnotu 1526 B a větší jednotky jsou považovány za neplatné.

Tab. 2.3: Výsledky testů propustnosti, TCP – FPGA UltraScale+ (Mango).

Počet CPU jader	MTU [B]		
	576	1300	1500
1	14,0 ± 0,9 Gbit/s	23,5 ± 1,4 Gbit/s	25,7 ± 2,0 Gbit/s
10	42,0 ± 5,4 Gbit/s	75,2 ± 2,6 Gbit/s	79,6 ± 3,6 Gbit/s

Tab. 2.4: Výsledky testů propustnosti, UDP – FPGA UltraScale+ (Mango).

Počet CPU jader	MTU [B]		
	576	1300	1500
1	1,3 ± 0,2 Gbit/s	3,1 ± 0,4 Gbit/s	3,4 ± 0,2 Gbit/s
10	10,4 ± 0,4 Gbit/s	22,5 ± 2,4 Gbit/s	26,5 ± 2,1 Gbit/s

Tab. 2.5: Výsledky testů propustnosti, TCP – FPGA Agilex (Intel).

Počet CPU jader	MTU [B]		
	576	1300	1500
1	15,2 ± 0,5 Gbit/s	23,5 ± 1,0 Gbit/s	25,3 ± 1,5 Gbit/s
10	43,3 ± 2,0 Gbit/s	81,3 ± 1,0 Gbit/s	85,8 ± 1,4 Gbit/s

Tab. 2.6: Výsledky testů propustnosti, UDP – FPGA Agilex (Intel).

Počet CPU jader	MTU [B]		
	576	1300	1500
1	1,3 ± 0,1 Gbit/s	2,9 ± 0,5 Gbit/s	3,5 ± 0,1 Gbit/s
10	10,5 ± 0,1 Gbit/s	23,2 ± 1,6 Gbit/s	26,5 ± 2,3 Gbit/s

Rychlosti dosažené při testech propustnosti byly srovnatelné s hodnotami nešifrovaného provozu. Limitace provozu šifrátory byla maximálně v řádu jednotek Gbit/s.

Pro šifrátory FPGA Intel byl dále testován režim QKD a funkčnost šifrátorů v dlouhodobém intervalu, protože v průběhu testování docházelo k desynchronizaci klíčů. To bylo vyřešeno komunikací s vývojáři FPGA a průběžnými aktualizacemi firmware. Výsledné hodnoty provozu při QKD prezentují následující dvě tabulky.

Tab. 2.7: Výsledky testů propustnosti, TCP – FPGA Agilex (Intel), QKD.

Počet CPU jader	MTU [B]		
	576	1300	1500
1	14,9 ± 1,1 Gbit/s	23,3 ± 2,5 Gbit/s	24,6 ± 1,8 Gbit/s
10	43,0 ± 0,2 Gbit/s	81,7 ± 2,7 Gbit/s	84,0 ± 2,8 Gbit/s

Tab. 2.8: Výsledky testů propustnosti, UDP – FPGA Agilex (Intel), QKD.

Počet CPU jader	MTU [B]		
	576	1300	1500
1	14,9 ± 1,1 Gbit/s	23,3 ± 2,5 Gbit/s	24,6 ± 1,8 Gbit/s
10	43,0 ± 0,2 Gbit/s	81,7 ± 2,7 Gbit/s	84,0 ± 2,8 Gbit/s

Režim QKD neměl na výsledné propustnosti vliv. Hodnoty pro TCP i UDP provoz byly v rámci odchylek téměř shodné s předchozími testy.

Dlouhodobé testy byly realizovány pro časové intervaly 2, 6 a 20 hodin. Jednalo se o testy s vytížením jednoho jádra CPU na každé stanici, kdy bylo cílem úspěšně realizovat provoz po celý časový interval. Tyto testy byly provedeny pro provoz v režimech s i bez QKD a proběhly úspěšně. Průměrné rychlosti uvádí tab. č. 2.9.

Tab. 2.9: Výsledky dlouhodobě šifrovaného provozu – FPGA Intel.

Režim provozu	Délka testu		
	2 hodiny	6 hodin	20 hodin
bez QKD	12,8 Gbit/s	12,8 Gbit/s	12,7 Gbit/s
s QKD	11,2 Gbit/s	14,9 Gbit/s	11,6 Gbit/s

Výsledky testů propustnosti TCP provozu 100G linky a MTU 1500 B shrnuje tabulka č. 2.10. Tabulka uvádí u každé hodnoty procentuální hodnotu odchylky, která byla odvozena od odchylky v Gbit/s z výše prezentovaných tabulek. Hodnoty nešifrovaného provozu jsou pouze ve formě průměrné hodnoty. Při testech nešifrovaného provozu v úvodní části byla pozornost věnována dosažení maximálních propustností a odchylky vypočítávány nebyly. Velikosti odchylek nešifrovaného provozu by však byly srovnatelné s šifrovaným provozem.

Tab. 2.10: Shrnutí výsledků TCP provozu 100G linky – MTU 1500 B.

Nešifrovaný provoz			
Počet CPU jader	MTU [B]		
	576	1300	1500
1	14,0 Gbit/s	26 Gbit/s	30,6 Gbit/s
10	38,9 Gbit/s	81,7 Gbit/s	85,4 Gbit/s
UltraScale+ (Mango)			
Počet CPU jader	MTU [B]		
	576	1300	1500
1	14,0 Gbit/s ± 6,4 %	23,5 Gbit/s ± 6,0 %	25,7 Gbit/s ± 7,8 %
10	42,0 Gbit/s ± 12,9 %	75,2 Gbit/s ± 3,5%	79,6 Gbit/s ± 4,5 %
Agilex (Intel)			
Počet CPU jader	MTU [B]		
	576	1300	1500
1	15,2 Gbit/s ± 3,3 %	23,5 Gbit/s ± 4,3 %	25,3 Gbit/s ± 5,9 %
10	43,3 Gbit/s ± 4,6 %	81,3 Gbit/s ± 1,2 %	85,8 Gbit/s ± 1,6 %
Agilex (Intel), QKD			
Počet CPU jader	MTU [B]		
	576	1300	1500
1	14,9 Gbit/s ± 7,4 %	23,3 Gbit/s ± 10,7 %	24,6 Gbit/s ± 7,3 %
10	43,0 Gbit/s ± 0,5 %	81,7 Gbit/s ± 3,3 %	84,0 Gbit/s ± 3,3 %

Pro šifrátory FPGA byl naplněn předpoklad minimální limitace provozu.

2.2 Firewally Juniper

Protože firewally Juniper disponovaly pouze 10G rozhraním, byla pro následující testování využit samostatná linka. Pro tuto linku byly testy nešifrovaného provozu zhotoveny samostatně a jsou zobrazeny v tabulce 2.11. Tyto testy byly realizovány pro přímo propojené stanice před přidáním firewallů.

Tab. 2.11: Výsledky testů propustnosti 10G linky pro nešifrovaný provoz.

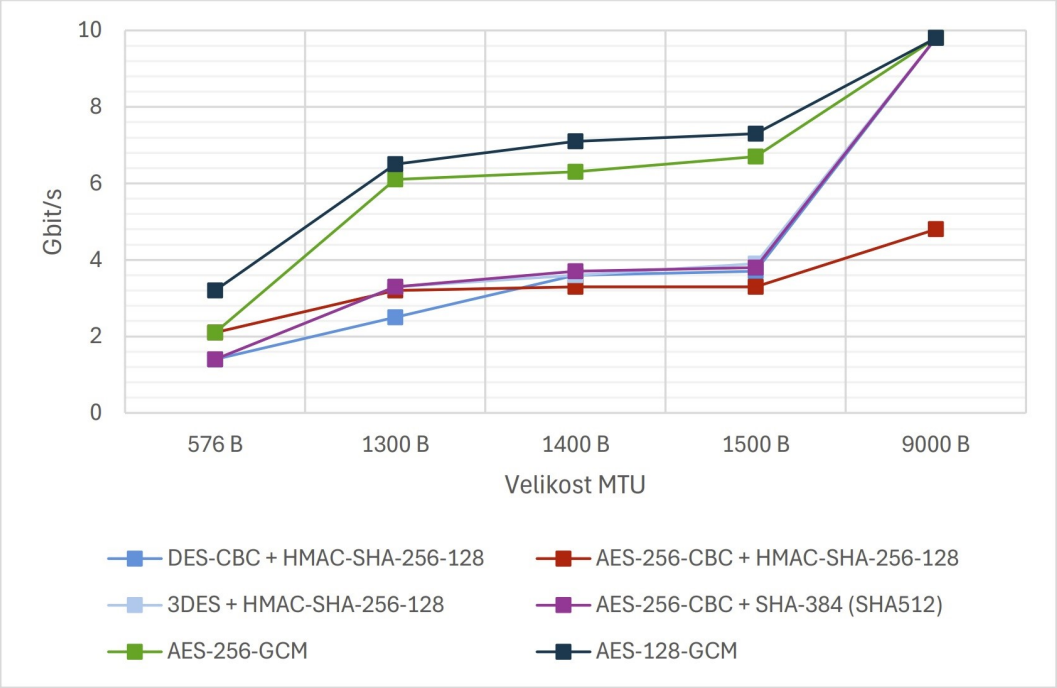
TCP				
Počet CPU jader	MTU [B]			
	576	1300	1500	9000
1	8,5 Gbit/s	9,3 Gbit/s	9,4 Gbit/s	9,9 Gbit/s
10	8,5 Gbit/s	9,3 Gbit/s	9,4 Gbit/s	9,9 Gbit/s
UDP				
Počet CPU jader	MTU [B]			
	576	1300	1500	9000
1	1,2 Gbit/s	1,9 Gbit/s	8,0 ± 1 Gbit/s	8,0 ± 1 Gbit/s
10	2 Gbit/s	7,0 Gbit/s	8,0 ± 1 Gbit/s	8,5 ± 1 Gbit/s

Rychlosti TCP provozu byly stejné pro jednojádrový i desetijádrový výkon. Provoz byl limitován kapacitou linky, ne vytížením CPU. Provoz UDP byl opět výrazně limitován a proměnné rychlosti jsou uvedeny s odchylkou. Průměrné hodnoty latence se pohybovaly v rozmezí 0,300 ms až 0,405 ms.

Pro šifrování provozu umožňují firewally volit a kombinovat vybrané parametry VPN komunikace. Lze volit různé autentizační mechanismy (ověřující přenášená data) a autentizační metodu (autentizující zdroj zpráv IKE, jako např. certifikáty, předsdílené klíče apod.). Pro parametry IPsec lze volit různé autentizační algoritmy (typ hashovacího algoritmu – ověřující data v paketech) a samotné šifrovací algoritmy. Cílem následujících testů tedy bylo nalézt nejvýkonnější a zároveň bezpečnou kombinaci těchto parametrů, která by byla vhodná pro kvantově odolný kanál. Pro daný model SRX4200 je výrobcem uvedena maximální propustnost IPsec VPN (Virtual Private Network) komunikace 35 Gbit/s. Tento prvek však disponuje 8x10G Ethernet rozhraními, takže se jedná o celkovou propustnost při komunikaci uskutečněnou s více rozhraními. Následující testy byly zhotoveny pro jedno 10G rozhraní na každém z firewallů. Zároveň výrobce u hodnoty propustnosti VPN uvádí jednotku MTU 1400 B, proto byly následující testy provedeny i pro tuto velikost. Maximální MTU pro IPsec bylo na Juniperech nastaveno na hodnotu 9192 B. To umožnilo testovat propustnost pro MTU 9000 B na koncových stanicích, aniž by docházelo

k „přetékání“ vlivem zapouzdřovacích IPsec dat.

Pro autentizační metodu a autentizační mechanismus VPN komunikace byla zvolena kombinace předsdíledných klíčů (PSK) a SHA-256. Tyto parametry zůstaly po celou dobu testování stejné, jelikož nebyl předpokládán výrazný dopad na propustnost v důsledku změn těchto parametrů. Ostatní parametry byly průběžně měněny a pro dané kombinace byly realizovány testy propustnosti. Shrnutí maximálních propustností pro každou kombinaci a velikost MTU zobrazuje graf na obr. č. 2.3.



Obr. 2.3: Graf hodnot propustností pro různé parametry VPN – Juniper SRX4200.

Výše testované kombinace zahrnují téměř všechny parametry, které lze na firewallech nastavit. Parametry, které testovány nebyly, zahrnují hash typu MD5 nebo různé délky šifrovacích klíčů algoritmu AES-CBC. Při užití těchto parametrů nebyly předpokládány výrazné odchylky od získaných hodnot. Šifry DES, 3DES a AES-CBC dosahovaly přibližně stejného výkonu. Určitý vliv na výkon měl také typ autentizačního algoritmu – hashe, který byl zvolen pro ověřování dat. Jednoznačně nejvýkonnějším šifrovacím algoritmem byl AES v módu GCM. Při užití AES-GCM bylo dosaženo propustností ∼7 Gbit/s pro běžné velikosti MTU pro TCP i UDP provoz. Propustnost při velikosti MTU 9000 B byla limitována schopnostmi 10G linky. Tím bylo dosaženo maximální teoretické propustnosti pro jednu fyzickou linku. Kombinace parametrů, která splňovala požadavky pro kvantově odolný kanál, byla zvolena následovně: Pro VPN kombinace PSK + SHA256 a pro IPsec šifrovací algoritmus AES-256-GCM. Zvolený šifrovací algoritmus byl stejný i pro šifrování na platformě FPGA.

2.3 Reálné aplikace

Na závěr byly testovány schopnosti šifrátorů v reálných aplikacích. Nejprve byla testována výkonnost síťové infrastruktury pro webovou komunikaci, a to v podobě měření rychlosti stahování souborů. K tomu byly využity webové servery Python HTTP a NGINX. Po zprovoznění serverů byla maximální rychlost stahování 550 MB/s, limitující faktor byla rychlost zápisu na disk klienta. Pro odstranění limitace byla část RAM využita jako další úložiště. Maximální rychlost zápisu na RAM disk byla 2,1 GB/s a této rychlost bylo dosaženo při stahování dvou souborů současně pomocí nástroje `wget2`. V další fázi proběhly testy zaměřené na provoz v reálném čase za pomoci video streamu. Přenos videa byl realizován pomocí přehrávače VLC. Video bylo vysílané z úložiště na virtuální stanici. Pro využití větší šířky pásma byly dále realizovány dva přenosy současně. Všechna videa zobrazené na stanici klienta byla bez viditelného zkreslení obrazu. V poslední části proběhlo testování provozu přes vysokorychlostní přepínač (Mikrotik CRS518-16XS-2XQ-RM), který byl přidán do 100 Gbit/s linky. Tyto testy měly za cíl zjistit, zda bude nějakým způsobem ovlivněn síťový provoz, pokud se nebude jednat o přímé propojení testovaných prvků. Testy propustnosti v této části firewally Juniper SRX4200 nezahrnovaly kvůli nižším dosažitelným rychlostem. Postup při realizaci testů propustnosti byl totožný s předchozím testováním. Výsledky zobrazují tab. č. 2.12 a tab. č.2.13.

Tab. 2.12: Provoz přes switch, TCP – FPGA Agilex (Intel).

Počet CPU jader	MTU [B]		
	576	1300	1500
1	13,7 ± 0,6 Gbit/s	21,7 ± 2,1 Gbit/s	25,4 ± 1,1 Gbit/s
10	42,2 ± 1,7 Gbit/s	72,6 ± 2,0 Gbit/s	80,0 ± 0,1 Gbit/s

Tab. 2.13: Provoz přes switch, UDP – FPGA Agilex (Intel).

Počet CPU jader	MTU [B]		
	576	1300	1500
1	1,1 ± 0,1 Gbit/s	2,4 ± 0,1 Gbit/s	3,2 ± 0,2 Gbit/s
10	9,1 ± 0,3 Gbit/s	20,1 ± 1,0 Gbit/s	22,9 ± 0,9 Gbit/s

Nejvyšší dosažená propustnost byla 80 Gbit/s. V případě propustnosti bez přepínače se jednalo o 85 Gbit/s. Přepínač tedy mírně limitoval provoz.

Závěr

Obsahem protokolu z měření je popis využité testovací infrastruktury a prezentace dosažených výsledků postkvantových šifrátorů. Síť byla realizována za pomoci virtualizace v podobě hypervizora XCP-ng a prostředí Xen Orchestra. Zde byly vytvořeny dvě virtuální stanice, které představovaly koncové body komunikace a mezi kterými byly utvářeny linky pro síťovou komunikaci. Do těchto linek byly postupně zapojovány testované prvky. Testy byly realizovány pro rychlosti až 100 Gbit/s.

Úvodní testování bylo zaměřené na dosažení maximální propustnosti 100G linky. Provoz byl limitován virtualizovaným prostředím a proto byla provedena optimalizace. Nejdůležitějšími kroky optimalizace byla změna síťových parametrů v systémové konfiguraci stanic, povolení PCI passthrough pro síťové karty a zapnutí offloading mechanismů. Po optimalizaci byla průměrná hodnota maximální propustnosti 96,4 Gbit/s (TCP) při velikosti MTU 9000 B. Hodnota latence klesla na třetinu.

Po optimalizaci byly realizovány testy propustnosti nešifrovaného a šifrovaného TCP a UDP provozu, které byly provedeny pro různý počet jader CPU a různé velikosti MTU. Pro běznou velikost MTU – 1500 B byla průměrná propustnost nešifrovaného TCP provozu 85,4 Gbit/s. Jednalo se o maximální výkon s využitím všech CPU jader. Testování šifrovaného provozu zahrnovalo dva typy FPGA šifrátorů (UltraScale+ a Agilex hardware) a firewally Juniper SRX4200. Při testech s FPGA šifrátory a velikostí MTU 1500 B přesahovala propustnost 80 Gbit/s pro oba typy hardware. Limitace provozu samotnými šifrátory tak byla minimální. Pro FPGA Agilex (Intel) byly dále realizovány testy v režimu QKD a dlouhodobé testy s i bez QKD, tyto testy proběhly v pořádku a nebyla pozorována žádná další limitace. Firewally Juniper SRX4200 disponovaly pouze 10G rohzraním. Testy s firewally zahrnovaly různé kombinace parametrů IPsec a VPN. Nejvyšší vliv na výkon měl šifrovací algoritmus. Nejvýkonnější šifrovací algoritmus byl jednoznačně AES-GCM.

Na závěr proběhly testy se skutečným síťovým provozem. Tyto testy zahrnovaly webovou komunikaci, přenos videa v reálném čase a provoz přes vysokorychlostní přepínač. Všechny šifrátory dokázaly tyto typy provozu spolehlivě šifrovat a nebyly pozorovány další výrazné limitace.

Seznam symbolů a zkratek

CPU	Central Processing Unit
FPGA	Field-Programmable Gate Array
IPsec	Internet Protocol Security
MTU	Maximum Transmission Unit
OS	Operating System
PCI	Peripheral Component Interconnect
PSK	Pre-Shared Keys
QKD	Quantum Key Distribution
RAM	Random Access Memory
TCP	Transmission Control Procotol
UDP	User Datagram Procotol
VPN	Virtual Private Network