VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

BAKALÁŘSKÁ PRÁCE

Brno, 2019 MARTIN HRDÝ

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

ZÁSUVNÝ MODUL APACHE JMETER ANALYZUJÍCÍ PARAMETRY DATOVÝCH SÍTÍ
APACHE JMETER PLUGIN TO ANALYZE OF DATA NETWORK PARAMETERS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR
Martin Hrdý

VEDOUCÍ PRÁCE
SUPERVISOR
doc. Ing. Petr Číka, Ph.D.

BRNO 2019

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ

Bakalářská práce

bakalářský studijní obor **Informační bezpečnost**
Ústav telekomunikací

Student: Martin Hrdý ***ID:*** 195826
Ročník: 3 ***Akademický rok:*** 2018/19

NÁZEV TÉMATU:

Zásuvný modul Apache JMeter analyzující parametry datových sítí

POKYNY PRO VYPRACOVÁNÍ:

Cílem bakalářské práce je návrh a implementace zásuvného modulu do nástroje Apache JMeter pro analýzu přenosových parametrů datových sítí. Seznamte se s nástrojem Apache JMeter a s přenosovými parametry datových sítí. Parametry specifikujte a popište metodiky využívané pro jejich měření. Na základě rozboru navrhněte a implementujte zásuvný modul do programu Apache JMeter, který umožní komplexní měření datových sítí z hlediska přenosových parametrů. Funkčnost modulu otestujte na experimentálním pracovišti.

DOPORUČENÁ LITERATURA:

[1] HALILI, Emily H. Apache JMeter: A practical beginner's guide to automated testing and performance measurement for your websites. Packt Publishing Ltd, 2008.

[2] BRANDER, S. RFC 2544 - Benchmarking Methodology for Network Interconnect Devices, IETF, 1999.

Termín zadání: 1.2.2019 ***Termín odevzdání:*** 27.5.2019

Vedoucí práce: doc. Ing. Petr Číka, Ph.D.
Konzultant:

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

ABSTRAKT

Bakalářská práce pojednává o možnostech využití testovacího nástroje Apache JMeter pro vytvoření zásuvného modulu realizujícího měření parametrů sítí. Cílem práce je navrhnout a následně zrealizovat zásuvný modul, který bude v souladu s patřičnými doporučeními realizovat měření přenosových parametrů datových sítí. V první části práce je popsán nástroj iPerf3 a testovací nástroj Apache JMeter. Další část se zabývá doporučeními pro analýzu přenosových parametrů sítě. V praktické části práce je popsán návrh modulu, jeho implementace a testování. Nakonec jsou v práci zpracovány a objasněny výsledky měření.

KLÍČOVÁ SLOVA

Apache JMeter, iPerf3, RFC 2544, RFC 6349, propustnost TCP, propustnost UDP

ABSTRACT

Bachelor thesis discusses opportunites for utilization of the Apache JMeter testing tool to creating plug-in, which implements network parameters measuring. The goal of thesis is to design and implement a plug-in, which will handle data network transmission parameters measuring in accordance with appropriate recommendations. The first part describes the iPerf3 tool and the Apache JMeter testing tool. The next section deals with network transmission parameters recommendations. The practical part describes design, implementation and testing of created plug-in. Finally, the results of measurements are processed and clarified.

KEYWORDS

Apache JMeter, iPerf3, RFC 2544, RFC 6349, TCP throughput, UDP throughput

HRDÝ, Martin. *Zásuvný modul Apache JMeter analyzující parametry datových sítí*. Brno, 2019, 56 s. Bakalářská práce. Vysoké učení technické v Brně, Fakulta elektrotechniky a komunikačních technologií, Ústav telekomunikací. Vedoucí práce: doc. Ing. Petr Číka, Ph.D.

PROHLÁŠENÍ

Prohlašuji, že svou bakalářskou práci na téma „Zásuvný modul Apache JMeter analyzující parametry datových sítí“ jsem vypracoval samostatně pod vedením vedoucího bakalářské 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 bakalářské práce dále prohlašuji, že v souvislosti s vytvořením této bakalářské 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

PODĚKOVÁNÍ

Rád bych poděkoval vedoucímu bakalářské práce panu doc. Ing. Petrovi Číkovi, Ph.D. za odborné vedení, konzultace, trpělivost a podnětné návrhy k práci.

Brno ....

podpis autora

Tato práce vznikla jako součást klíčové aktivity KA6 - Individuální výuka a zapojení studentů bakalářských a magisterských studijních programů do výzkumu v rámci projektu OP VVV Vytvoření double-degree doktorského studijního programu Elektronika a informační technologie a vytvoření doktorského studijního programu Informační bezpečnost, reg. č. CZ.02.2.69/0.0/0.0/16_018/0002575.

EVROPSKÁ UNIE
Evropské strukturální a investiční fondy
Operační program Výzkum, vývoj a vzdělávání

MINISTERSTVO ŠKOLSTVÍ, MLÁDEŽE A TĚLOVÝCHOVY

Projekt je spolufinancován Evropskou unií.

Obsah

Seznam obrázků

Seznam tabulek

Seznam výpisů

Úvod

V poslední době stále roste počet zařízení připojených do sítě a rovněž roste množství přenášených dat. Výpočetní technika se rozvíjí a požadavky na kvalitu a rychlost internetového připojení se zvyšují. Pokud chtějí mít správci sítí kontrolu nad správnou funkčností sítě, je nutné parametry těchto sítí zkoumat, analyzovat a měřit.

Postupy měření a analyzování sítě jsou popsány a sjednoceny v mnohých doporučeních RFC (*Request For Comments*). Jedná se o řadu dokumentů, které popisují internetové protokoly. Dokumenty RFC jsou považovány za doporučení, nikoli normy. Práce se zaměřuje na RFC 2544 [1] a RFC 6349 [2].

Doporučení RFC 2544, které bylo publikováno roku 1999, vychází ze staršího RFC 1242 [3]. Doporučení definuje skupinu testů, které mohou být využity k testování a analýze síťových zařízení. Dokument popisuje parametry jako například propustnost, zpoždění, zotavení testovaného zařízení po přetížení nebo zotavení testovaného zařízení po restartu. Jde o časově náročnou metodiku, která je nevhodná pro komplexní měření a její použití není v dnešní době doporučeno.

Doporučení RFC 6349, vydané roku 2011, využívá modernější protokol TCP. Metodika se změřuje na parametry jako například maximální přenosovou jednotku cesty, nejnižší šířku pásma trasy, obousměrné zpoždění nebo propustnost TCP.

Dle zadání byl pro měření přenosových parametrů datových sítí vytvořen zásuvný modul v aplikaci Apache JMeter [4]. Jedná se o open-source testovací aplikaci určenou například pro testování serverů FTP (*File Transfer Protocol*) nebo HTTP (*Hypertext Transfer Protocol*). Existuje celá řada nástrojů měřících parametry IP sítí. Pro vyvinutý modul byl zvolen nástroj iPerf3 [5], který je multiplatformní a nabízí mnoho nastavitelných argumentů.

Pro navržení modulu, který bude efektivně a spolehlivě analyzovat přenosové parametry datových sítí bylo nutné danou problematiku nastudovat. Teoreticky byly rozebrány a nastudovány standardy RFC 2544 a RFC 6349, na jejichž základech a principech byl zásuvný modul navrhnut, vytvořen a otestován. Rovněž bylo nutné nastudovat funkčnost nástroje iPerf3 a aplikace Apache JMeter.

Teoretická část práce se věnuje právě aplikaci Apache JMeter, nástroji iPerf3 a doporučením RFC 2544 a RFC 2544. Praktická část popisuje implementaci zásuvného modulu, ověření funkčnosti a realizaci měření vytvořeného modulu.

1 Aplikace Apache JMeter

Apache JMeter je jedna z mnoha open-source testovacích aplikací. Byla vytvořena společností Apache Software Foundation primárně na testování výkonnosti Apache JServ servletu, později byla rozšířena o další funkce a dnes umožňuje komplexní testování například serverů FTP (*File Transfer Protocol*), serverů HTTP (*Hypertext Transfer Protocol*) nebo databázových serverů. Celá aplikace je vytvořena pomocí programovacího jazyku Java, z tohoto důvodu je multiplatformní a je zaručena její kompatibilita mezi různými zařízeními díky JVM (*Java Virtual Machine*). Zdrojový kód aplikace je veřejně dostupný [4]. Apache JMeter nabízí uživatelské rozhraní, které je implementováno pomocí knihovny Swing. Testy v aplikaci Apache JMeter probíhají v rámci testovacího plánu, který definuje průběh testu [6, 7].

1.1 Prvky testovacího plánu

Z důvodu sporných překladů budou v následujícím textu použity anglické názvy prvků aplikace Apache JMeter.

Prvky testovacího plánu musí zahrnovat alespoň jednu skupinu vláken (thread group). V rámci každé thread group je možno umístit kombinaci jednoho nebo více prvků. Apache JMeter obsahuje následující prvky:

- **thread group** (*skupina vláken*) – reprezentuje skupinu uživatelů,
- **sampler** (*vzorník*) – umožňuje definovat specifické typy požadavků na server,
- **logic controller** (*logický ovladač*) – dokáže nastavit pořadí zpracování samplerů v thread group,
- **listener** (*naslouchač*) – zobrazuje výsledky sampleru ve vizuální formě (grafu, tabulky, stromové struktury či textu),
- **timer** (*časovač*) – dovoluje pozasatavit průběh testu na určitou dobu mezi dvěma požadavky vytvořenými pomocí thread group,
- **assertions** (*tvrzení*) – umožňují přidat validační test na odpověď požadavku sampleru,
- **configuration elements** (*konfigurační prvky*) – pomocí konfiguračních prvků lze vytvořit proměnné k sampleru,
- **pre-processor elements** (*pre-procesorové prvky*) – prvky modifikující nastavení požadavků samplerů před jejich spuštěním,
- **post-processor elements** (*post-procesorové prvky*) – prvky, které jsou spuštěny po vykonání žádosti sampleru.

Apache JMeter tedy nabízí širokou škálu prvků, které jsou schopny zachytit informace o testované aplikaci, uložit je a zobrazit v různých formátech [7].

2 Testovací nástroj iPerf3

Nástroj iPerf3 slouží k aktivnímu měření maximální dosažitelné šířky pásma (*bandwidth*) v IP sítích. Nástroj byl vyvinut společnostmi ESnet a Lawrence Berkeley Nation Laboratory. Nástroj je multiplatformní – je kompatibilní s Windows, Linux a macOS. Program podporuje ladění různých parametrů jako časování, vyrovnávací paměť nebo typ protokolu (TCP, UDP, SCTP). U každého testu je uvedena ztrátovost, šířka pásma a další parametry [5].

V oficiální dokumentaci nástroje iPerf3 je uvedeno mnoho argumentů, které mohou být využity k různému nastavení měření. Argumenty jsou rozděleny na obecné (tab. 2.1), serverové (tab. 2.2) a klientské (tab. 2.3).

Tab. 2.1: Základní obecné argumenty nástroje iPerf3

Argument	Význam
`-B`	nastavení vysílacího rozhraní pro klienta a přijímacího rozhraní pro server, vhodné pro zařízení s více síťovými rozhraními
`--d`	vytvoření debugovacího výstupu určeného primárně vývojářům
`-J`	vytvoření výstupu ve formátu JSON
`-p`	nastavení čísla portu naslouchání na straně serveru nebo připojení na straně klienta (výchozí port je 5201)
`-V`	vytvoření detailního výstupu
`--logfile`	uložení výstupu měření do samostatného souboru

Tab. 2.2: Základní argumenty nástroje iPerf3 specifické pro server

Argument	Význam
`-D`	spustí server v pozadí jako démona
`-I`	vytvoří soubor s PID (*Process Identifier*), vhodné při spuštění v pozadí
`-s`	spustí iPerf v serverovém módu

Tab. 2.3: Základní argumenty nástroje iPerf3 specifické pro klienta

Argument	Význam
`-b`	cílová šířka pásma (*bandwidth*) v bitech za sekundu (výchozí hodnota pro UDP je 1 Mb/s, pro TCP není omezena)
`-c`	spustí iPerf v klientském módu
`-k`	počet paketů k zaslání (alternativa k argumentu `-t` nebo `-n`)
`-l`	délka zásobníku (vyrovnávací paměti) ke čtení či zápisu (výchozí hodnota pro TCP je 128 kB, pro UDP 8 kB)
`-n`	počet zásobníků, tento argument přepisuje výchozí chování (vysílání pod dobu 10 s – viz argument `-t`) a vysílá pole l bajtů n-krát nezávisle na době trvání
`-P`	počet paralelních připojení k serveru (výchozí hodnota je 1)
`-t`	doba přenosu v sekundách, iPerf ve výchozím stavu zasílá opakovaně pole l bajtů po dobu t sekund (výchozí hodnota je 10 s)
`-u`	upřednostnění protokolu UDP před TCP
`-4`	použití pouze IPv4
`-6`	použití pouze IPv6

3 Doporučení pro analýzu přenosových parametrů sítě

Níže jsou popsána doporučení RFC 2544 a RFC 6349, která budou implementována ve vytvořeném zásuvném modulu aplikace Apache JMeter.

3.1 Doporučení RFC 2544

Dokument RFC 2544 definuje specifickou skupinu testů, které mohou být využity k testování a analýze jednotlivých síťových zařízení. Metodika je časově náročná a neřeší všechny parametry, které moderní měření vyžaduje. RFC 2544 se nezmiňuje o variaci zpoždění (*jitter*) či SLA (*Service Level Agreement*). Výsledky testů poskytují uživateli srovnatelná data z různých zdrojů, pomocí kterých je možné vyhodnotit parametry zařízení. Dokument vychází z RFC 1242 [3]. Metoda není vhodná pro komplexní měření, její použití není v dnešní době doporučeno [1, 8].

Dle RFC 2544 by všechny testy měly být provedeny při různých velikostech rámců, které jsou definovány pro různé technologie. Pro testování na standardu Ethernet jsou definovány tyto velikosti rámců: 64, 128, 256, 512, 1024, 1280 a 1518 bajtů. Výčet velikostí zahrnuje minimální a maximální velikost rámce podporovanou standardem Ethernet. Testování hodnot definovaných mezi minimem a maximem zaručuje vyšší přesnost měření [1].

Doporučení RFC 2544 popisuje nutné nastavení DUT (*Device Under Test*) před zahájením testování. Předpokládáno je nakonfigurování a aktivování všech protokolů, které jsou daným zařízením podporovány. Během testování není dovoleno konfigurovat či jakkoliv nastavovat DUT, pokud to není vyžadováno pro konkrétní následující dílčí test [1].

Jedna z možností implementace testů je zapojení takového testovacího zařízení, které je vybaveno přijímacím i vysílacím portem. Data jsou při testu vyslána z vysílacího portu testovacího zařízení na přijímací port DUT a následně jsou přeposlána vysílacím portem DUT na přijímací port testovacího zařízení (viz obrázek 3.1). Stejné funkcionality může být dosaženo i při použití samostatného vysílacího a přijímacího zařízení (viz obrázek 3.2), nicméně v tomto případě je nutné počítat s nadbytečnou prací spojenou s úkony nutnými k zajištění přesnosti provedení některéch testů. Doporučení popisuje také implementaci více DUT (viz obrázek 3.3). Tato sestava může v mnoha případech přesněji simulovat chování v reálném světě – například propojení dvou LAN (*Local Area Network*) s WAN (*Wide Area Network*) [1].

Obr. 3.1: Zapojení obsahující testovací zařízení s přijímacím i vysílacím portem

Obr. 3.2: Zapojení obsahující samostatný přijímač a vysílač

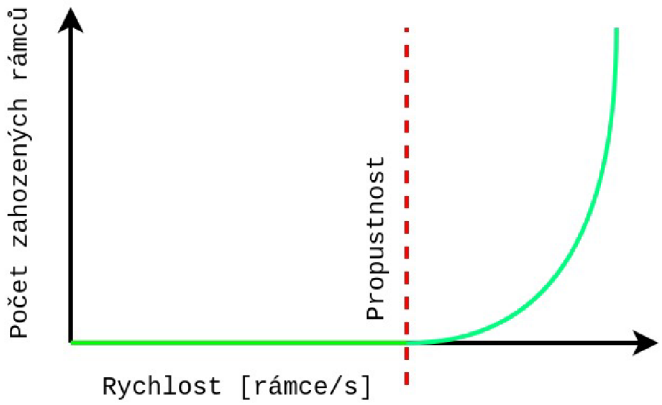
Obr. 3.3: Zapojení s více DUT

3.1.1 Propustnost

Propustnost (*Throughput*) je definována jako nejvyšší rychlost zasílání rámců, při které nejsou žádné zaslané rámce zahozeny. Propustnost je vyjádřena v rámcích, bajtech, či bitech za sekundu [1].

Průběh testování

Cílem testu je stanovit propustnost DUT dle definice z RFC 1242. Při testu propustnosti je zaslán konkrétní počet rámců o určité rychlosti přes DUT, následně jsou rámce přenášené přes DUT sečteny. Pokud je počet zaslaných rámců shodný s počtem rámců přijatých DUT, je rychlost zasílání rámců zvýšena, v opačném případě je rychlost zasílání snížena [1].



Obr. 3.4: Princip stanovení propustnosti dle RFC 2544

Výsledek testování

Výsledek testu propustnosti by měl být zpracován v podobě grafu, osa X by měla reprezentovat velikost rámců a osa Y rychlost odesílání rámců. Graf by měl obsahovat nejméně dvě křivky. První křivkou je vyjádřena teoretická rychlost zasílání rámců při různých velikostech rámců. Druhá křivka reprezentuje výsledky měření [1].

3.1.2 Zpoždění

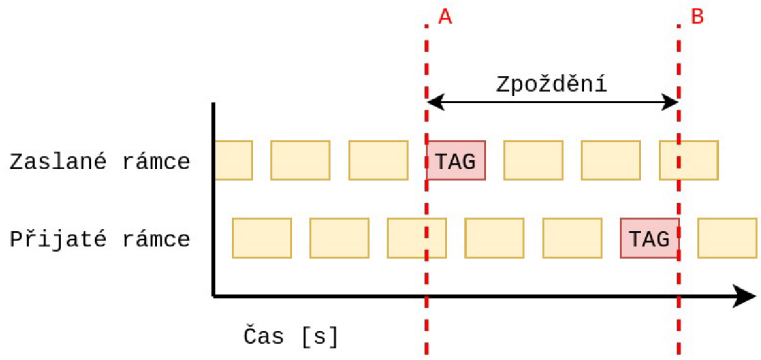
Zpoždění (*Latency*) je definováno jako doba, kterou zabere rámci cesta z jednoho místa na druhé.

Průběh testování

Cílem testu je stanovit zpoždění dle definice z RFC 1242. Před testováním zpoždění je nutno zjistit propustnost DUT pro každou ze stanovených velikostí rámců. Doba vysílání by měla trvat nejméně 120 vteřin, přičemž po šedesáté vteřině vysílání by měl být vložen do proudu identifikační štítek (*tag*). Čas zaslání tohoto rámce je zaznamenán (časová značka A). Tagovaný rámec musí být rozeznán druhou stranou (přijímačem) a čas přijetí je rovněž zaznamenán (časová značka B). Zpoždění je následně určeno jako rozdíl časové značky B a značky A. Testování musí být zopakováno minimálně dvacetkrát, jako výsledná hodnota je zaznamenán průměr naměřených hodnot. Výsledné zpoždění lze zapsat vzorcem:

$$\text{Zpoždění} = \frac{1}{n} * \sum_{i=1}^{n} B_i - A_i \quad [\text{s}], \tag{3.1}$$

kde n udává počet opakování testu, A_i čas vyslání rámce dílčího testu a B_i čas přijetí rámce dílčího testu [1].



Obr. 3.5: Princip stanovení zpoždění dle RFC 2544

Výsledek testování

Výsledky testu měření zpoždění by měly být vyobrazeny v podobě tabulky. Každá testovaná velikost rámce je zapsána do řádku. Ve sloupcích by měly být uvedeny různé rychlosti odesílání rámců [1].

3.1.3 Ztrátovost rámců

Ztrátovost rámců (*Frame Loss Rate*) uvádí počet rámců, které nebyly doručeny, z celkového počtu zaslaných rámců. Ztrátovost rámců je uvedena v procentech.

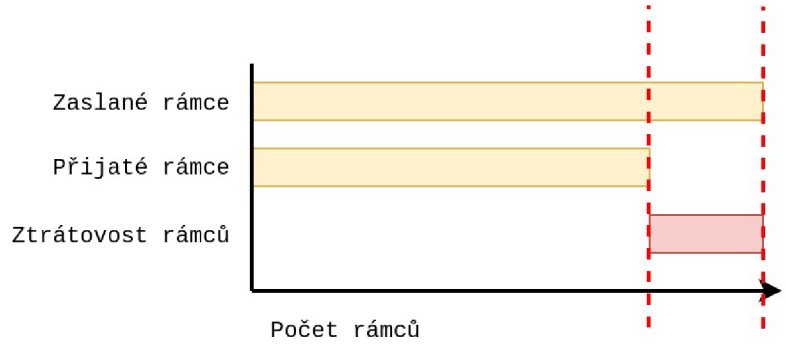
Průběh testování

Při testování ztrátovosti rámců je zaslán určitý počet rámců při dané rychlosti do DUT. Rámce přijaté DUT jsou sečteny. Ztrátovost rámců je vyjádřena vztahem:

$$\text{Ztrátovost rámců} = \frac{(\text{vstupní součet rámců} - \text{výstupní součet rámců}) * 100}{\text{vstupní součet rámců}} [\%]. \tag{3.2}$$

V prvním testování by měla rychlost odesílání rámců odpovídat maximální rychlosti (tedy 100 %) pro danou velikost rámce na vstupním médiu. Postup je opakován pro rychlost odpovídající 90 % maximální rychlosti a poté 80 % této rychlosti. Testování by mělo pokračovat snižováním intervalu (až na 10 %) dokud

nejsou nalezeny dva stejné testy bez ztráty rámců. Vyšší přesnost měření je zaručena snižováním rychlosti odesílání rámců v menších intervalech než je 10 % [1].



Obr. 3.6: Princip stanovení ztrátovosti rámců dle RFC 2544

Výsledek testování

Výsledek testování ztrátovosti rámců by měl být vykreslen v podobě grafu, přičemž osa X musí znázorňovat rychlost odesílání rámců jako procentuální hodnotu maximální teoretické rychlosti. Osa Y znázorňuje procentuálně ztrátovost rámců při dané rychlosti odesílání. Osy X i Y musí být vykresleny od 0 % do 100 %. Více křivek v grafu je použito ke znázornění různých velikostí rámců nebo protokolů [1].

3.1.4 Back-to-back rámce

Testování určuje schopnost zařízení DUT zpracovat back-to-back rámce dle doporučení RFC 1242 [3].

Průběh testování

V rámci testování je na DUT zaslán proud rámců s minimální mezerou mezi rámci (inter-packet gap). Rámce přeposlané DUT jsou sečteny. Pokud je počet rámců přeposlaných DUT stejný, jako počet zaslaných rámců, dojde k prodloužení doby zasílání a opětovnému testování. Pokud je naopak počet zaslaných rámců vyšší, než počet přeposlaných rámců zařízením DUT, doba zasílání je snížena a test je opakován.

Výsledná back-to-back hodnota je rovna počtu rámců v nejdelší testované době vysílání, při které nedošlo ke ztrátě rámců. Testování musí trvat nejméně 2 sekundy. Jako výsledná hodnota by měl být uveden aritmetický průměr z nejméně 50 dílčích testů [1].

Výsledek testování

Výsledky testu back-to-back rámců by měly být uvedeny v tabulce. V řádcích by měly být uvedeny testované velikosti rámců. Sloupce by měly obsahovat průměrný počet rámců pro každý typ datového toku. Je doporučeno uvést odchylku pro každé dílčí měření [1].

3.1.5 Zotavení testovaného zařízení po přetížení

Cílem měření je zjištění doby potřebné k obnovení DUT ze stavu přetížení. Před samotným testováním doby zotavení systému je nutné zjistit propustnost pro každou velikost rámce.

Průběh testování

Při měření jsou zaslány rámce takovou rychlostí, která odpovídá 110 % z naměřené hodnoty propustnosti pro danou velikost rámce nebo maximální danou rychlostí média (nelze překročit maximální podporovanou rychlost média). Zasílání rámců trvá alespoň 60 sekund. Časovou značkou A je označen moment, kdy se rychlost zasílání změní z 110 % na 50 %. Zachycení poslední ztráty rámce je označeno časovou značkou B. Čas zotavení systému je poté určen jako rozdíl značek A a B. Testování je doporučeno provést vícekrát, přičemž jako výsledná hodnota je uveden průměrný výsledek hodnot [1].

Výsledek testování

Výsledek měření doby zotavení systému by měl být uveden ve formátu tabulky, která obsahuje řádek pro každou velikost měřeného rámce. Ve sloupcích by měla být uvedena propustnost pro každý typ datového toku a naměřený čas nutný k zotavení [1].

3.1.6 Zotavení testovaného zařízení po restartu

Cílem testu je stanovení doby, která je potřebná pro zotavení DUT po softwarovém restartu nebo restartu celého zařízení.

Průběh testování

Před samotným testováním je nutno stanovit propustnost zařízení při nejmenší velikosti rámce. Při testování zotavení DUT po restartu je vyslán proud rámců, které mají rychlost dle dříve změřené propustnosti na nejmenší velikosti rámce. Následně je na DUT vyvolán restart. Při testu je zachycen čas přeposlání posledního rámce před

výpadkem (časová značka A) a čas, kdy je zaslán první rámec obnoveného toku po výpadku (časová značka B). Doba nutná pro zotavení je rovna rozdílu těchto časových značek. Test restartu pomocí odstavení napájení je prováděn stejným způsobem, přičemž napájení by mělo být vypnuto po dobu 10 sekund. Měření zotavení po restartu by mělo být prováděno pouze za použití rámců adresovaných na přímo připojené sítě k DUT z důvodu možného zpoždění vyvolaného aktualizacemi směrovacích tabulek. Test by měl obsahovat restarty softwarové, hardwarové i odstavení napájení [1].

Výsledek testování

Výsledky testování zotavení DUT po restartu by měly být uvedeny v podobě krátkých prohlášení. Každé prohlášení by mělo popisovat jeden typ restartu [1].

3.2 Doporučení RFC 6349

Zásadní výhodou doporučení RFC 6349 je použití protokolu TCP, který je v dnešních počítačových sítích dominantně využíván pro komunikaci. Při použití protokolu TCP musí být docíleno správného nastavení vysílacích a přijímacích oken (*transmit and receive windows*), zásobníků a velikosti paketů. Metoda je doporučena pro ověřování parametrů přístupu k internetové síti poskytovaného v pevném místě. Metodika RFC 6349 se zaměřuje na:

- maximální přenosovou jednotku cesty,
- nejnižší šířku pásma trasy,
- obousměrné zpoždění,
- propustnost TCP,
- minimální TCP RWND [2, 8].

3.2.1 Maximální přenosová jednotka cesty

Aby nedocházelo při měření k fragmentaci datagramů, a tedy k nepřesnému měření, je nutné určit maximální přenosovou jednotku dané měřené cesty (*Path Maximum Transmission Unit*). Techniky pro zjištění MTU (*Maximum Transmission Unit*) cesty jsou označeny jako PMTUD (*Path Maximum Transmission Unit Discovery techniques*). Technika PMTUD využívá zpráv ICMP (*Internet Control Message Protocol*) k určení MTU cesty. Pokud chce zařízení vyslat paket, který má v hlavičce bitový příznak DF (*don't fragment* – zákaz fragmentace) a zároveň má paket větší velikost, než je MTU následujícího směrovače (*next hop*), paket je zahozen. Při zahození je rovněž odeslána zpráva ICMP, která upozorňuje, že daný paket vyžaduje fragmentaci. Nevýhoda této metody spočívá v použití protokolu ICMP, jelikož ICMP

může být z bezpečtnostních důvodů blokován správcem dané sítě, a proto tato metoda není vždy spolehlivá [2].

Řešení zablokovaných ICMP zpráv přináší RFC 4821 [9] s technikou PLPMTUD (*Packetization Layer Path Maximum Transmission Unit Discovery*). Metoda může být použita s povolenými i zablokovanými zprávami ICMP. RFC 4821 specifikuje parametry `search_low` a `search_high` pro MTU, které jsou RFC 6349 doporučeny [2].

3.2.2 Nejnižší šířka pásma trasy

Před provedením testu propustnosti TCP by měla být definována nejnižší šířka pásma trasy (BB – *Bottleneck Bandwidth*) pomocí testu, který využívá bez-stavového (stateless) protokolu. Existuje mnoho známých technik, díky kterým lze změřit BB dané sítě. Běžnou praktikou pro poskytovatele internetového připojení je využití metody dle RFC 2544, nicméně je nutno podotknout, že tato metoda byla vytvořena pouze pro testování síťových prvků v laboratorních podmínkách, nikoli pro měření v reálném provozu. Další možností je použití technik definovaných v dokumentu RFC 5136 [10]. RFC 5136 je zaměřeno na měření v reálných podmínkách, avšak v dokumentu není uveden žádný konkrétní postup, definovány jsou pouze obecné matematické výpočty, a proto je využitelnost tohoto RFC minimální [2, 11].

3.2.3 Obousměrné zpoždění

Obousměrné zpoždění (RTT – *Round-trip Time*) je popsáno jako doba, která uplyne mezi odesláním prvního bitu segmentu TCP a přijetím posledního bitu odpovídajícího potvrzení segmentu TCP. Pro realizaci měření RTT lze použít více způsobů, které se od sebe mohou lišit například v přesnosti. RFC 6349 definuje jako vhodný nástroj k měření RTT iPerf, FTP nebo obdobné nástroje pracující na základě zachytávání paketů z testovaných relací TCP. Důležité je uvést, že výsledky založené na SYN – SYN-ACK na začátku relace TCP by neměly být použity, jelikož firewall může zpomalit navázání spojení (*3-way handshake*). K poskytnutí hodnoty RTT může být využito také ICMP pingu v případě, že je zohledněno využití různě velkých velikostí datagramů. Při použití ICMP pingu může nastat problém v případě, kdy je na některém z uzlů testované sítě (NUT – *Network Under Test*) definována kvalita služeb (QoS – *Quality of Service*). Tato metoda není považována za přesnou. Při testování na reálné síti také není zaručeno, že všechna zařízení odpovídají na ICMP ping (některá zařízení mohou pingy filtrovat) [2, 11].

3.2.4 Propustnost TCP

Metoda RFC 6349 konkrétně definuje měřicí techniky propustnosti TCP (*TCP Throughput*) pro ověření maximálního dosažitelného výkonu TCP v spravované podnikové síti. Za každých okolností je doporučeno testovat propustnost TCP nejdřív v obou směrech nezávisle na sobě a poté testovat v obou směrech najednou. Rovněž je doporučeno realizovat testy v různou dobu během dne (kvůli zatížení reálné sítě) [2].

Metrika poměr času přenosu

Metrika poměr času přenosu (TTR – *Transfer Time Ratio*) je definována jako poměr mezi aktuální hodnotou TCP TT (*Transfer Time*) a ideální hodnotou TCP TT. Jedná se o hodnotu bez jednotek. Tato metrika je dána rovnicí:

$$\text{TCP TTR} = \frac{\text{aTT [s]}}{\text{iTT [s]}}\text{[-]}, \tag{3.3}$$

kde aTT je aktuální a iTT ideální TT. Aktuální TCP TT je doba, za kterou je datový blok přenesen přes dané spojení TCP. Ideální TCP TT je předpovězená doba, za kterou by měl být datový blok přenesen. Je odvozena od maximálního průtoku TCP dat na transportní vrstvě ISO/OSI modelu.

Výpočet iTT vychází z rovnice:

$$\text{iTT} = \frac{\text{Objem přenosu [b]}}{\text{Teoretická propustnost TCP [b/s]}}\text{[s]}, \tag{3.4}$$

přičemž teoretickou propustnost TCP lze vyjádřit vztahem:

$$\text{Teor. p. TCP} = (\text{MTU} - 40) * 8 * \frac{\text{BB}}{\text{MTU} + \text{IFG} + \text{SFD} + \text{FCS} + \text{PR} + \text{ETH}}\text{[b/s]}, \tag{3.5}$$

kde IFG (*Inter-frame gap*) je mezera mezi rámci, SFD (*Start of frame delimeter*) označení začátku rámce, FCS (*Frame-check sequence*) kontrolní součet rámce, PR (*Premble*) preambule a ETH (*Ethernet*) je zdrojová, cílová adresa MAC (*Media Access Control*) a typ rámce.

Příkladem může být použití technologie 100BASE-TX:

$$\text{Teor. p. TCP} = (1500 - 40) * 8 * \frac{100\,\text{Mb/s}}{(1500 + 12 + 1 + 4 + 7 + 14) * 8} = 94{,}93\,\text{Mb/s}, \tag{3.6}$$

kdy teoretická TCP propustnost dosáhne hodnoty 94,93 Mb/s. Vyšších hodnot lze dosáhnout při použití tzv. jumbo rámců [2].

Metrika efektivita TCP

Metrika efektivita TCP (*TCP Efficiency*) reprezntuje procento bajtů, které nemusely být opětovně poslány. Metrika udává představu o chybovosti spojení TCP a nutnosti opakovaného zaslání paketů při výskytu chyb na lince. Metrika se dá vyjádřit vztahem:

$$\text{Efektivita TCP} = \frac{(\text{TB [B]} - \text{RTB [B]}) * 100}{\text{TB [B]}}[\%], \tag{3.7}$$

kde TB (*Transmitted Bytes*) udává počet přenesených bajtů a RTB (*Retransmitted bytes*) udává počet opětovně zaslaných bajtů [2].

Metrika zpoždění zásobníku

Třetí metrika je zpoždění zásobníku (BD – *Buffer Delay*). Zpoždění zásobníku lze vyjádřit jako vztah mezi nárůstem obousměrného zpoždění (RTT) během testu propustnosti TCP a ideálním RTT. Tuto metriku popisuje vzorec:

$$\text{Zpoždění zásobníku} = \frac{\text{ARTT [s]} - \text{BRTT [s]}}{\text{BRTT [s]}}[\%], \tag{3.8}$$

kde ARTT (*Average Round-Trip Time*) označuje průmerné obousměrné zpoždění a BRTT (*Baseline Round-Trip Time*) je obousměrné zpoždění měřené v čase mimo maximální využití. ARTT lze vyjádřit rovnicí:

$$\text{ARTT} = \frac{\text{TRTT [s]}}{\text{TD [s]}}[\text{s}], \tag{3.9}$$

kde TRTT (*Total Round-Trip Time*) je součet RTT za dobu přenosu a TD (*Transfer Duration*) je trvání přenosu. Při špatném poměru doby přenosu (tedy pokud je aTT mnohem vyšší než iTT) lze problém diagnostikovat pomocí porovnání metrik efektivity TCP a zpoždění zásobníku [2].

3.2.5 Minimální TCP RWND

Parametr TCP RWND ovlivňuje propustnost relace TCP. Velikost TCP RWND je určena 16ti bitovou hodnotou `window size` v hlavičce daného segmentu TCP [12]. Hodnota `window size` může být násobena konstantou `window scale option`, která se vyjednává při navázání spojení. Minimální požadované TCP RWND je vypočteno z BDP (*Bandwidth-Delay Product*) dle následující rovnice:

$$\text{BDP} = \text{RTT [s]} * \text{BB [b/s]} \quad \text{[b]}. \tag{3.10}$$

Pro získání minimálního TCP RWND v bajtech je nutno vydělit BDP číslem 8 podle vzorce:

$$\text{Minimální TCP RWND} = \frac{\text{BDP [b]}}{8}\text{[B]}. \tag{3.11}$$

Na asymetrických cestách je nutno provést samostatné výpočty pro trasy v obou směrech [2].

3.2.6 Počet současných spojení TCP

Volba jednoho nebo vícenásobného spojení TCP závisí na velikosti minimálního TCP RWND a hodnoty TCP RWND na přijímací stanici. V některých případech je nutné použít vícenásobné spojení k pokrytí celé kapacity měřené trasy. Pokud je hodnota minimálního TCP RWND vyšší, než TCP RWND přijímací strany, je nutno využít více spojení. Pro určení počtu spojení slouží rovnice:

$$\text{Počet TCP spojení} = \frac{\text{Minimální TCP RWND [B]}}{\text{TCP RWND [B]}}, \tag{3.12}$$

přičemž výsledná hodnota musí být zaokrouhlena nahoru na celé číslo. Obecně je vhodné využít vícenásobné spojení TCP i v takových situacích, kdy vícenásobné spojení TCP není zdálnivě vyžadováno. V reálných situacích může při velikosti TCP RWND větší než 64 kB docházet k přehlížení předem nastavené hodntoy nebo její rekonfiguraci na výchozí hodnotu (většinou 64 kB).

4 Implementace zásuvného modulu

Následující sekce popisují postup při vytváření zásuvného modulu do aplikace Apache JMeter realizující měření přenosových parametrů dle RFC 2544 a RFC 6349 s využitím nástroje iPerf3.

4.1 JMeter v prostředí Eclipse

Oficiální stránky společnosti Apache nabízí ke stažení dvě verze JMeteru – binary a source. Nejdříve je potřeba stáhnout verzi programu se zdrojovými kódy (source). V práci je použit Apache JMeter verze 5.0. Po stažení JMeteru je nutno soubor extrahovat. Jelikož je JMeter kompletně vytvořen pomocí jazyku Java, jako vývojové prostředí je použit program Eclipse IDE. Jde o open-source vývojovou platformu, která je určena především pro vývoj v jazyce Java. Ze staženého source souboru JMeteru lze vytvořit Eclipse projekt pomocí `File > New > Java Project`, přičemž jako `Location` musí být vybrána právě rozbalená složka JMeteru [4, 13]. Dle [14] je pro správnou kompilaci programu v souboru `NewDriver.java` nutno upravit proměnnou `tmpDir` podle výpisu 4.1.

Výpis 4.1: Úprava souboru NewDriver.java

```
if (tmpDir.length() == 0) {
    File userDir = new File(System.getProperty("user.dir"));
    // Old path
    // tmpDir = userDir.getAbsoluteFile().getParent();
    // New path
    tmpDir = userDir.getAbsolutePath();
}
```

Dále je dle návodu v souboru `eclipse.md` přejmenován soubor `eclipse.classpath` na `.classpath`. Pro kompilaci a sestavení aplikace používá JMeter nástroj Ant. Ve výchozím nastavení je v Antu obsaženo několik úloh (*tasks*), které mohou být snadno využity vývojáři. Pokud okno Antu není zobrazeno, lze ho zobrazit pomocí `Window > Show view > Ant`. Důležité je následné spuštění úlohy `download_jars`, která má za úkol stažení a následnou instalaci jar souborů potřebných pro správnou funkčnost JMeteru. Poté je nutné spustit úlohu `install default`. Nakonec lze provést kompilaci, vytvoření balíčků a start GUI (*Graphical User Interface*) z jar souborů pomocí úlohy `run_gui`. Pokud jsou při spuštění Ant úlohy `run_gui` vyžadovány soubory `log2j4.xml` a `jmeter.properties`, je možno dané soubory nakopírovat například z binary verze JMeteru z adresáře `bin` [4, 13, 14].

4.2 Adresářová struktura JMeteru

Apache JMeter je setříděn podle protokolů a funkcionality. Vývojáři proto nejsou nuceni sestavovat celou aplikaci, ale mohou vyvíjet nové soubory `jar` pouze pro jeden protokol či funkcionalitu. Následující tabulka 4.1 popisuje strukturu souborů JMeteru [15].

Tab. 4.1: Adresářová struktura programu JMeter

Adresář	Obsah
bin	soubory .bat a .sh potřebné ke startu JMeteru
docs	dokumentační soubory programu
extras	soubory Antu
lib	soubory jar pro JMeter
lib/ext	soubory jar jádra a jednotlivých protokolů
src	podsložky pro každý protokol a komponentu
src/components	protokolově nezávislé komponenty (vizualizéry apod.)
src/core	kód jádra JMeteru, obsahuje všechny abstraktní třídy a rozhraní jádra
src/examples	demonstrační kód sampleru
src/functinos	standardní funkce využívané všemi komponenty
src/htmlparser	snímek HtmlParseru, sponzorovaného HtmlParser projektem na sourceforge
src/jorphan	třídy poskytující běžné funkce utilit
src/monitor	monitorovací komponenty tomcatu 5
src/protocol	různé protokoly podporované JMeterem (např. FTP, HTTP, TCP)
test	určeno pro jednotkové testy
xdocs	soubory .xml pro dokumentaci (JMeter generuje dokumentaci ze souborů .xml)

4.3 Vytvoření zásuvného modulu

Zásuvný modul je vytvořen z již existující předlohy, která se nachází v adresáři `src/examples`. V prvním kroku realizace je vytvořena složka modulu. Modul má název `Analyzer` a nachází se v adresáři v `src/Analyzer`. Po zkopírování předlohy do adresáře nového modulu je nutno soubory upravit – například změnit názvy souborů,

tříd a konstruktorů. Dále je potřeba upravit soubor `build.xml`, díky kterému se při spuštění zkompiluje kromě vestavěných pluginů i nově vytvořený modul. Rovněž se zkopíruje šablona HTML (*Hypertext Markup Language*) potřebná k výpisu získaných výsledků. Úprava souboru `build.xml` je zobrazena ve výpisu 4.2. Po kompilaci lze v JMeteru přidat plugin pomocí `Add > Sampler > Analyzer` [14, 15].

4.4 Struktura modulu

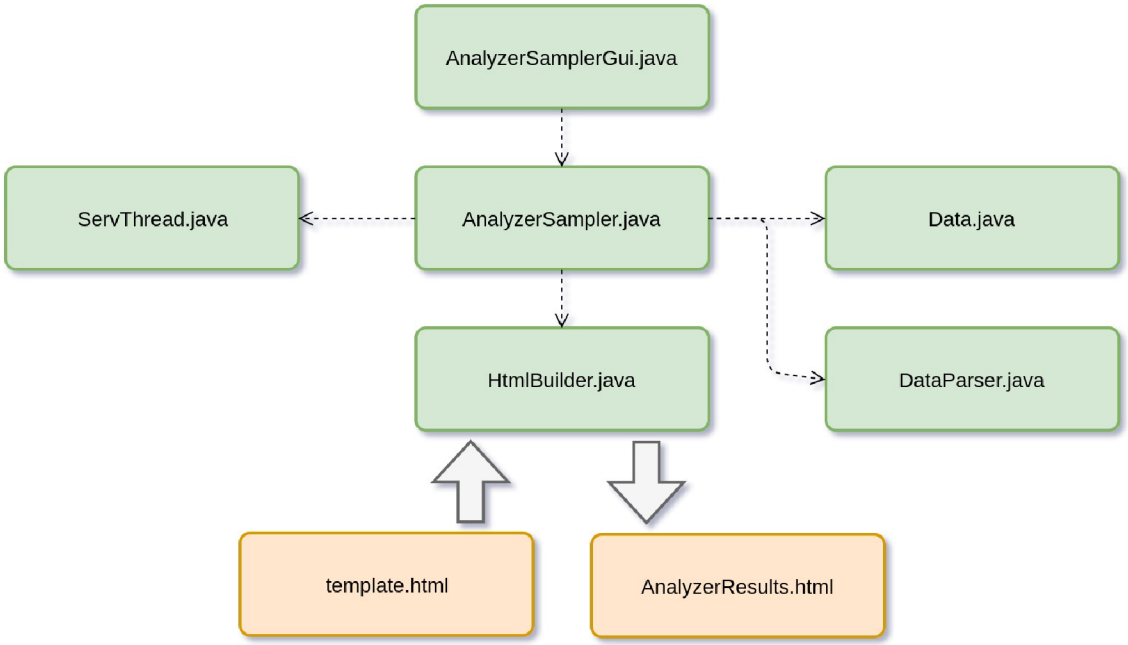
Struktura tříd a souborů vytvořeného zásuvného modulu je zobrazena na obrázku 4.1. Grafické uživatelské rozhraní je vytvořeno v objektu `AnalyzerSamplerGui`, který je provázán s `AnalyzerSampler`. Jádrem modulu je `AnalyzerSampler`, který uchovává další čtyři objekty – `Data`, `DataParser`, `ServThread` a `HtmlBuilder`. Objekt `AnalyzerSampler` rovněž řídí spouštění testů pomocí nástroje iPerf3 nebo systémové konzole. Pokud je modul v módu serveru, dochází k vytvoření objektu `ServThread`, který vytvoří nové vlákno pro běh serveru. Spouštění samotných konzolových příkazů, zachycení dat z konzolového výstupu a následné vytřídění potřebných dat zajišťuje objekt `DataParser`. Získaná data jsou poté ukládána do objektu `Data`. Po dokončení měření jsou data zaslána do objektu `HtmlBuilder`. Tento objekt načte šablonu výsledného souboru HTML a na základě dat v objektu `Data` nahradí obecné proměnné. Následně je vytvořen a otevřen výsledný soubor HTML s výsledky měření.

4.5 Podpora operačního systému

Při návrhu zásuvného modulu Apache JMeter bylo zamýšleno vytvoření verze pro linuxové systémy i systém Windows. Při tvorbě grafického uživatelského rozhraní sampleru modulu nastal problém s některými vstupními parametry. Jelikož lze nástroj iPerf3 na rozdíl od Windows v linuxových systémech spustit z jakéhokoliv umístění, verze modulu pro Windows by musela obsahovat nadbytečné vstupy. Tyto vstupy by byly nutné například pro zadání cesty k spustitelnému souboru programu iPerf3. Rozdílná je i práce s procesy, která je nutná při manipulaci s iPerf3 serverem. Dalším rozdílem ve verzích pro OS Windows a Linux je rozdílná implementace příkazu `ping`. Příkaz `ping` je využíván pro určení maximální přenosové jednotky cesty a pro obousměrné zpoždění. Implementace příkazu `ping` v OS Linux nabízí vyšší přesnost měření a je tedy mnohem vhodnější. Z důvodu zachování elementárnosti grafického uživatelského rozhraní a dosažení nejvyšší přesnosti modulu bude kompatibilita zaručena pouze s linuxovými systémy.

Výpis 4.2: Úprava skriptu build.xml

```
...
<class location="${dest.jar}/Analyzer.jar"/>
...
<property name="src.Analyzer" value="src/Analyzer"/>
...
<pathelement location="${src.Analyzer}"/>
...
<target name="compile-Analyzer" depends="compile-jorphan,
compile-core,compile-components">
  <mkdir dir="${build.Analyzer}"/>
  <javac srcdir="${src.Analyzer}" destdir="${build.Analyzer}"
  source="${src.java.version}" optimize="${optimize}"
  debug="on" target="${target.java.version}"
  includeAntRuntime="${includeAntRuntime}"
  deprecation="${deprecation}" encoding="${encoding}">
    <include name="**/*.java"/>
    <classpath>
      <pathelement location="${build.jorphan}"/>
      <pathelement location="${build.core}"/>
      <pathelement location="${build.components}"/>
      <path refid="classpath"/>
    </classpath>
  </javac>
  <jar destfile="${dest.jar}/Analyzer.jar"
  basedir="${build.Analyzer}" />
  <copy file="${src.Analyzer}/template.html"
  todir="${dest.jar}"/>
</target>
...
<target name="compile-protocols" depends="compile-http,
compile-ftp,compile-jdbc,compile-java,compile-ldap,
compile-mail,compile-tcp,compile-Analyzer"
description="Compile␣all␣protocol-specific␣components."/>
...
<property name="build.Analyzer" value="build/Analyzer"/>
...
```



Obr. 4.1: Struktura tříd a souborů zásuvného modulu

4.6 Limitace nástroje iPerf3

Nástroj iPerf3 dokáže v reverzním módu (parametr `-P`) zasílat data z cílové stanice do klientské. Tento mód je využit pro test propustnosti TCP příchozích dat (*download*). Při reverzním módu jsou ovšem konzolové výpisy značně omezeny. Ve výpisech reverzního módu není například uvedeno obousměrné zpoždění, které je klíčové pro stanovení metriky zpoždění zásobníku i poměru času přenosu. Z tohoto důvodu jsou metriky uvedeny pouze pro test propustnosti TCP odchozích dat (*upload*).

Další limitací nástroje iPerf3 je absence oboustranného testování (bidirectional testing), kdy dochází k testu odchozích a příchozích dat současně. Tento krok je v RFC 6349 nepovinný, nicméně za předpokladu, že by byl nástrojem iPerf3 podporován, by nebyl náročný na implementaci. Tato funkcionalita byla dle webové stránky programu iPerf3 [5] podporována ve verzi iPerf2, nicméně do současné verze nebyla přepsána.

4.7 Módy zásuvného modulu

Zásuvný modul Apache JMeter využívá nástroj iPerf3, který vyžaduje mimo konfigurace klienta také konfiguraci serveru (cílové stanice). Pro zjednodušení konfigurace je v zásuvném modelu na výběr mezi dvěma módy – klient nebo server. Klient realizuje příkazy, zachycuje odezvu a následně zpracovává hodnoty do přehledné

podoby, server přijímá příkazy ze strany klienta. Při nastavení zásuvného modulu do módu serveru je po kliknutí na tlačítko „Start“ vytvořeno nové vlákno, které spustí server iPerf3. Hlavní vlákno vyčkává na ukončení vlákna spuštěného serveru, a proto je modul spuštěn až do přerušení na vyžádání uživatele pomocí tlačítka „Stop“. Ukončení běžícího serveru probíhá pomocí detekce jeho jedinečného čísla procesu (PID) a následného ukončení procesu.

4.8 Grafické uživatelské rozhraní modulu

Do grafického uživatelského rozhraní vytvořeného zásuvného modulu bylo nutné přidat prvky, pomocí kterých může uživatel aplikaci předávat vstupní data a parametry. Podoba grafického rozhraní zásuvného modulu je zachycena na obrázku 4.2. Jako první parametr lze nastavit mód daného zařízení (viz sekce 4.7). Na základě nastaveného módu je zneaktivněna ta část parametrů, kterou není nutno vyplnit (při nastavení zařízení do módu serveru jsou zneaktivněna pole nutná pro nastavení klienta a naopak).

Dále obsahuje grafické uživatelské rozhraní dvě sekce. Sekce nastavení klienta obsahuje pole IP adresy cílové stanice a jejího portu. Dále lze nastavit výstupní adresář pro vygenerování souboru HTML s výsledky měření, přičemž cestu k adresáři lze zadat ručně do textového pole, nebo lze využít vestavěný systémový prohlížeč souborů pomocí tlačítka „Select directory“. Poslední obecný nastavitelný parametr klientské sekce je nejnižší šířka pásma trasy.

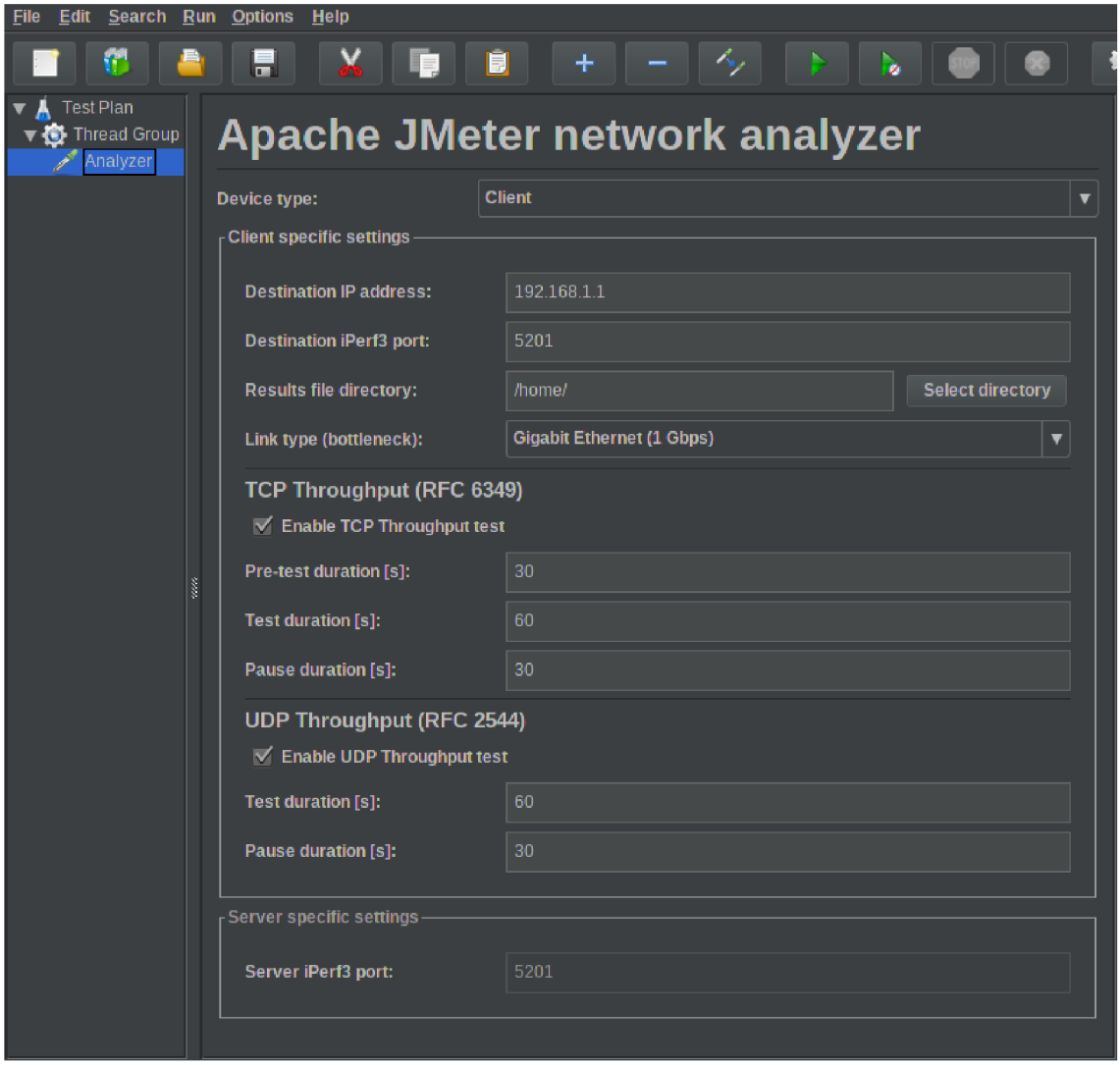
Následující pole jsou již parametry samotných testů propustností UDP a TCP. Každý test obsahuje zaškrtávací políčko (*checkbox*) na aktivování daného testu. Pokud není určitý test aktivován, nedochází k jeho vykreslení do výsledného souboru HTML. U testů lze nastavit délku trvání testu i délku vyčkávání mezi testy. Test propustnosti TCP obsahuje navíc pole pro zadání délky pre-testu, při kterém jsou vypočteny základní hodnoty před započetím samotného měření.

Textová vstupní pole jsou předvyplněna výchozími hodnotami, které jsou nastaveny na minimální přípustné hodnoty dle daného doporučení (RFC 2544 nebo RFC 6349). Čísla portů nástroje iPerf3 jsou rovněž předvyplněna na hodnotu, se kterou se iPerf3 spustí ve výchozím stavu.

Sekce nastavení serveru obsahuje pouze pole pro nastavení čísla portu.

4.9 Ošetření vstupních hodnot

Před spuštěním samotného testování je nutné zkontrolovat správné nastavení hodnot.



Obr. 4.2: Grafické uživatelské rozhraní zásuvného modulu

Jako první je kontrolována dostupnost cílové IP adresy, přítomnost aktivního iPerf3 serveru a správná cesta adresáře výstupního souboru. Pokud se nástroj iPerf3 nemůže spojit s cílovou IP adresou, vypíše chybovou hlášku. Tato situace je typická pro připojení typu klient-DUT-server. Problematické je ovšem zapojení, kdy je klient připojený rovněž do globální internetové sítě. V tomto případě může hledat klient iPerf3 servery i v globální internetové síti a při zadání nesprávné IP adresy se dokáže zacyklit. Tento problém je ošetřen přidáním příkazu `timeout`, který omezí dobu hledání. V případě zadání nesprávné adresy je testování ukončeno a uživatel je informován o nesprávnosti zadaných údajů v konzolovém výstupu a na konci výstupního souboru HTML. Čísla portů jsou při zadání nesprávné hodnoty nastavena na výchozí hodnotu nástroje iPerf3 (5201).

Výpis 4.3: Příkaz pro kontrolu existence iPerf3 serveru

```
timeout 5 iperf3 -c getIpAddr() -t 1
```

Ošetřeno je také zadávání vstupních parametrů testů TCP a UDP. Uživatel je při zadání záporných nebo nulových hodnot trvání testu (popřípadě doby čekání mezi testy) informován, že hodnota byla automaticky změněna na minimální možnou hodnotu a není nucen zadat opětovně parametry. V případě automatické úpravy časových hodnot je uživatel opět informován stejnou formou jako u IP adresy.

4.10 Reprezentace výsledků

Ke grafickému zobrazení naměřených a vypočtených hodnot slouží výstupní soubor HTML. Tento výstupní soubor je generován z obecné šablony, která obsahuje místo hodnot proměnné ohraničené speciálním znakem. Šablona je při zkompilování zásuvného modulu díky skriptu `build` zkopírována do složky `lib/ext` k hlavním souborům modulu. Při spuštění testu je šablona načtena do zásuvného modulu, následně dochází k dosazení hodnot na místo proměnných. Po dosazení všech hodnot dojde k vygenerování výsledného souboru HTML. Uživatel může v GUI modulu sám nastavit výstupní adresář tohoto souboru. Po dokončení testování je soubor s výsledky automaticky otevřen.

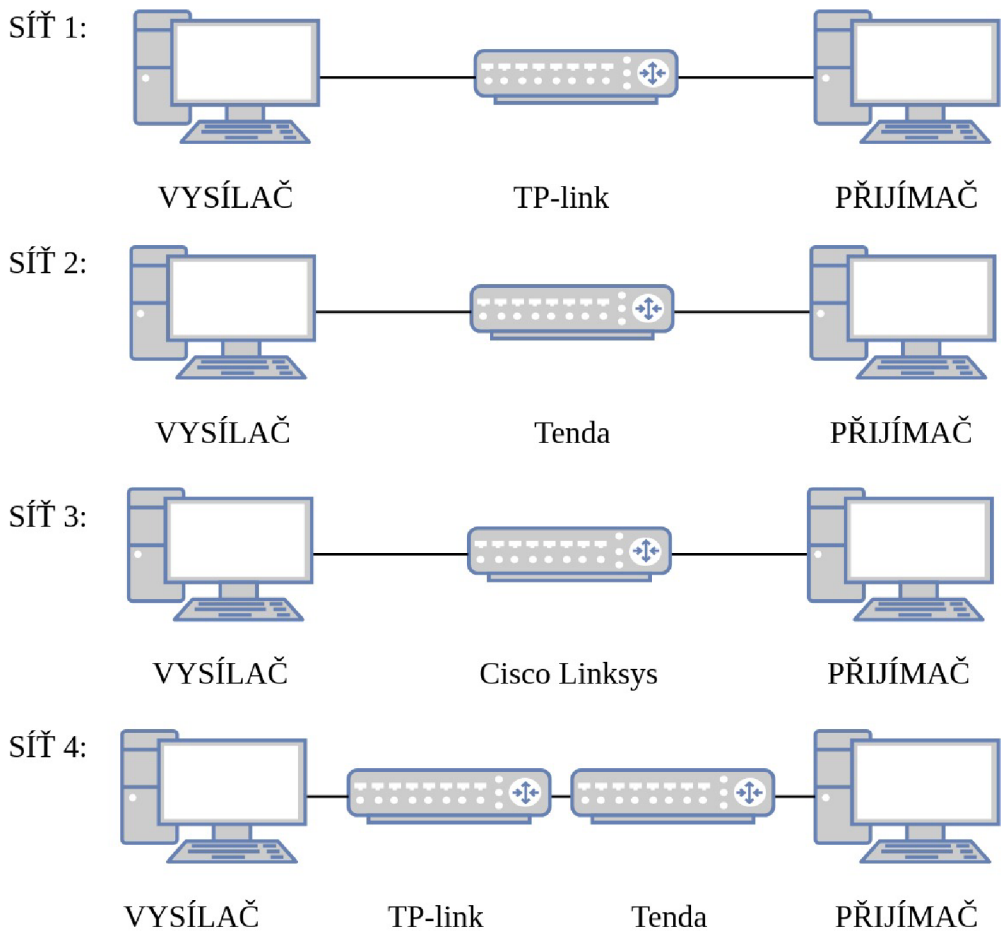
Výsledný soubor HTML zobrazuje výsledky měření v přehledných tabulkách a grafech. Pro vykreslení grafů je využita javascriptová open-source knihovna ChartJs [16], která nabízí mnoho typů grafů a je jednoduchá na implementaci.

Pro usnadnění stylování souboru HTML bylo využito open-source nástroje Bootstrap [17], který nabízí předchystané třídy pro nadpisy, tabulky a další elementy HTML. Výsledný soubor HTML je plně responzivní a je možné ho zobrazit i na starších zařízeních s nižším rozlišením.

Vygenerovaný soubor HTML je validní v souladu se standardem W3C (*World Wide Web Censortium*) [18], a proto je zaručena jeho podpora a správné zobrazení v různých prohlížečích.

5 Ověření funkčnosti a realizace měření

Měření a ověření funkčnosti vyvinutého modulu probíhalo za pomoci dvou počítačů osazených síťovými kartami Gigabit Ethernet. Jako DUT byly použity tři různé směrovače – Cisco Linksys WRT610N s rychlostí portů 1 Gb/s, Tenda N300 s rychlostí 100 Mb/s a TP-link TL-WR741ND s rychlostí 100 Mb/s. Využito bylo různých zapojení sítě dle potřeb konkrétního testu (viz obr. 5.1). Pro ověření správnosti přeposílaných dat byl použit program Wireshark [19].



Obr. 5.1: Schéma zapojení pro testování zásuvného modulu Apache JMeter

Kompletnímu testu s výpisem všech hodnot předcházelo testování dílčích algoritmů pro určení jednotlivých parametrů a metrik a ověření jejich správnosti.

5.1 Maximální přenosová jednotka cesty

Vytvořený modul dokáže získat hodnotu maximální přenosové jednotky cesty v souladu s doporučením RFC 6349. Využito je paketů ICMP, které jsou nastaveny na největší možnou velikost (65 535 bajtů). Od této velikosti je nutno odečíst velikost hlaviček [20]. Pro záhlaví IP je rezervováno 20 bajtů a pro ICMP echo request je alokováno dalších 8 bajtů. Po spuštění příkazu 5.1 je z konzole zachycen chybový výstup, který udává, jakou nejvyšší velikost MTU lze použít.

Výpis 5.1: Příkaz pro detekci maximální přenosové jednotky cesty

```
ping getIpAddr() -M do -s 65507 -c 1
```

Pro ověření funkčnosti algoritmu bylo využito tří zapojení (sítě 1, 2 a 4 viz obrázek 5.1). U těchto zapojení bylo upraveno výchozí nastavení směrovačů. Směrovači Tenda byla zachována výchozí hodnota MTU – 1500 B. Pro testovací účely bylo směrovači TP-link sníženo MTU z výchozí hodnoty na 1200 B. Z výsledků měření (tabulka 5.1) lze vypozorovat, že algoritmus měření maximální přenosové jednotky trasy modulu funguje správně.

Tab. 5.1: Výsledky testování maximální přenosové jednotky cesty

Síť	MTU směrovačů [B]	Výsledek měření [B]
Síť 1	1200	1200
Síť 2	1500	1500
Síť 4	1200, 1500	1200

5.2 Obousměrné zpoždění

Při testování je měřeno minimální, maximální a průměrné obousměrné zpoždění. Doporučení RFC 6349 nabízí více metod měření, během této práce byly testovány a porovnány metody pomocí nástroje iPerf3 a ICMP pingu. Testy proběhly na zařízeních TP-link a Cisco (sítě 1 a 3 viz obr. 5.1). Výsledky jsou zobrazeny v tabulce 5.2. Z výsledků měření lze vypozorovat, že průměrné obousměrné zpoždění má u těchto dvou metod malý rozdíl (1,64 ms) a zejména na směrovači Cisco s rozhraními Gigabit Ethernet jsou zpoždění téměř totožná (rozdíl 0,028 ms). Metoda využívající zprávy ICMP má mnohem větší odchylky dílčích měření a také není zaručeno, že zprávy ICMP nebudou blokovány při reálném nasazení měřícího zásuvného modulu v praxi. Z těchto důvodů je ve výsledném modulu použita pouze metoda pomocí nástroje iPerf3.

Tab. 5.2: Výsledky měření obousměrného zpoždění

Směrovač	Cisco		TP-link	
Metoda	ICMP	iPerf3	ICMP	iPerf3
Min RTT [ms]	0,448	0,931	0,711	2,582
Max RTT [ms]	1,113	1,086	1,947	2,883
Avg RTT [ms]	1,002	1,030	1,034	2,674

5.3 Propustnost TCP

5.3.1 Postup testování

Zásuvný modul realizuje měření propustnosti TCP dle doporučení RFC 6349. Samotnému spuštění testu předchází prvotní test (pre-test), při kterém jsou zjištěny potřebné hodnoty pro nastavení parametrů měření. Během pre-testu je zjištěna maximální přenosová jednotka cesty, změřeno obousměrné zpoždění, dále je dopočítána velikost okna TCP a počet současných spojení TCP. Před zahájením měření jsou získány parametry testu zadané uživatelem – trvání testu a doba čekání mezi testy. Na základě těchto parametrů jsou vygenerovány dva spouštěcí příkazy nástroje iPerf3 (5.2, 5.3), které slouží k vykonání testu propustnosti TCP příchozího a odchozího spojení. Příkaz pro příchozí spojení se liší pouze v parametru `-P`, který zajišťuje reverse mode (mód, při kterém serverová stanice vysílá a klientská přijímá). Jak je popsáno v sekci 4.6, současný test příchozího a odchozího spojení není pomocí nástroje iPerf ve verzi 3 realizovatelný, nicméně jde pouze o volitelnou součást celkového testu.

Výpis 5.2: Sestavený příkaz pro spuštění testu propustnosti TCP odchozího spojení

```
iperf3 -c getIpAddr() -w D.getTcpRwnd()/D.getTcpCon() -J -t
getTcpDur() -l D.getBDP()/8 -P D.getTcpCon()
```

Výpis 5.3: Sestavený příkaz pro spuštění testu propustnosti TCP příchozího spojení

```
iperf3 -c getIpAddr() -R -w D.getTcpRwnd()/D.getTcpCon() -J
-t getTcpDur() -l D.getBDP()/8 -P D.getTcpCon()
```

5.3.2 Výpočet metrik

Po dokončení měření propustnosti TCP jsou vypočteny metriky – poměr času přenosu a zpoždění zásobníku. Metriky jsou vypočteny pouze pro test odchozího spojení, jelikož nástroj iPerf3 nezobrazuje v reverzním módu potřebné údaje (viz kapitola

4.6), nicméně pokud je trasa měření symetrická, metriky spojení TCP by neměly být výrazně odlišné. Pro získání metrik v obou směrech je možné po dokončení prvního testu zaměnit módy koncových zařízení a provést měření také v opačném směru.

Metrika poměru času přenosu je získána pomocí hodnot ideálního a aktuálního času přenosu. Z výsledků testu odchozího spojení je získán počet přenesených bajtů. Tato hodnota je po převedení využita ve vzorci 3.4 pro získání ideálního času přenosu. Hodnota aktuálního trvání přenosu je rovněž získána z výsledků testu odchozího spojení, nicméně hodnota se téměř neliší od hodnoty doby trvání testu zadané uživatelem (odchylka $\approx 0{,}1\,\mathrm{ms}$).

Při výpočtu metriky zpoždění zásobníku jsou využity hodnoty průměrného a minimálního obousměrného zpoždění při testu odchozího spojení, přičemž minimální hodnota je považována za hodnotu při prázdném (nezahlceném) zásobníku. Pomocí těchto hodnot je metrika vypočtena dle vzorce 3.8;

5.3.3 Měření a zhodnocení výsledků

Měření propustnosti TCP probíhalo na DUT Cisco (s rychlostí portů 1 Gb/s) a TP-link (s rychlostí portů 100 Mb/s). Výsledky měření jsou zobrazeny na snímcích 5.2 a 5.3. Na snímích jsou zobrazeny výsledky pre-testu, dopočtené hodnoty, propustnosti TCP v obou směrech, obousměrné zpoždění testu odchozího spojení a metriky TCP testu odchozího spojení. Grafy jsou zachyceny na snímcích 5.4 a 5.5.

Z naměřených hodnot při testu propustnosti TCP lze usoudit, že cesty jsou symetrické, neboť se výsledky pro odchozí a příchozí data téměř neliší. Lze také předpokládat, že metriky získané při testu propustnosti TCP příchozích dat budou velmi obdobné jako u testu dat odchozích. U DUT s porty Fast Ethernet je v porovnání s DUT s porty Gigabit Ethernet získáno vyšší obousměrné zpoždění a zpoždění zásobníku. Hodnota metriky poměru času přenosu je obdobná. Rozdílný je počet nutných spojení TCP – zařízení s rozhraními Gigabit Ethernet už vyžaduje dvě současná spojení. Oběma zařízením byla zjištěná výchozí hodnota maximální přenosové jednotky – 1500 bajtů.

Test parameters:

MTU	Maximum transmission unit:	1500 B
BB	Bottleneck bandwidth:	100.0 Mb/s
MinRTT	Minimum round-trip time (pre-test):	1.389 ms
MaxRTT	Maximum round-trip time (pre-test):	1.529 ms
AvgRTT	Average round-trip time (pre-test):	1.429 ms
BDP	Bandwidth delay product:	142900 b
TCP RWND	TCP Reciever window size:	17.862 KB
TCP CON	Number of TCP connections:	1

Measured throughputs:

TH UP	Measured upload throughput:	94,07 Mb/s
TH DOWN	Measured download throughput:	94,13 Mb/s

Round-trip time during test (upload):

MinRTT	Minimum round-trip time:	1.422 ms
MaxRTT	Maximum round-trip time:	2.21 ms
AvgRTT	Average round-trip time:	1.882 ms

TCP Metrics (upload):

iTT	Ideal transfer time:	59,4988 s
aTT	Actual transfer time:	60,0401 s
TTR	Transfer time ratio:	1,0091
BD	Buffer delay [%]:	32,35 %

Obr. 5.2: Výsledky měření propustnosti TCP na zařízení s rozhraními Fast Ethernet

Test parameters:

MTU	Maximum transmission unit:	1500 B
BB	Bottleneck bandwidth:	1000.0 Mb/s
MinRTT	Minimum round-trip time (pre-test):	0.665 ms
MaxRTT	Maximum round-trip time (pre-test):	0.979 ms
AvgRTT	Average round-trip time (pre-test):	0.853 ms
BDP	Bandwidth delay product:	853000 b
TCP RWND	TCP Reciever window size:	106.625 KB
TCP CON	Number of TCP connections:	2

Measured throughputs:

TH UP	Measured upload throughput:	939,6 Mb/s
TH DOWN	Measured download throughput:	940,29 Mb/s

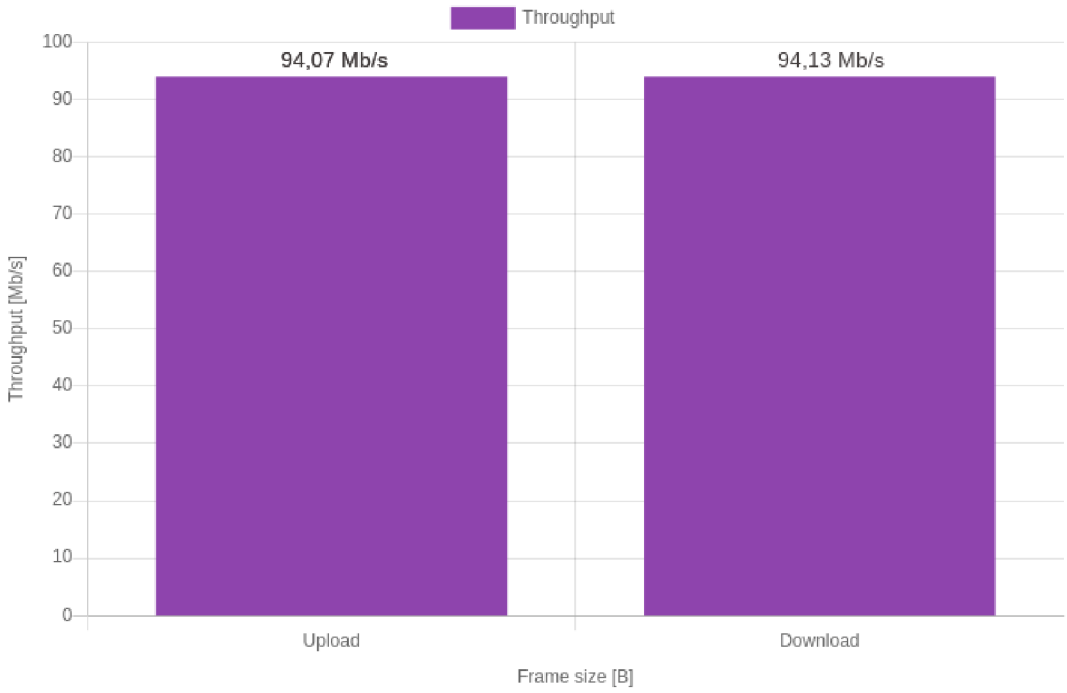
Round-trip time during test (upload):

MinRTT	Minimum round-trip time:	0.592 ms
MaxRTT	Maximum round-trip time:	0.686 ms
AvgRTT	Average round-trip time:	0.632 ms

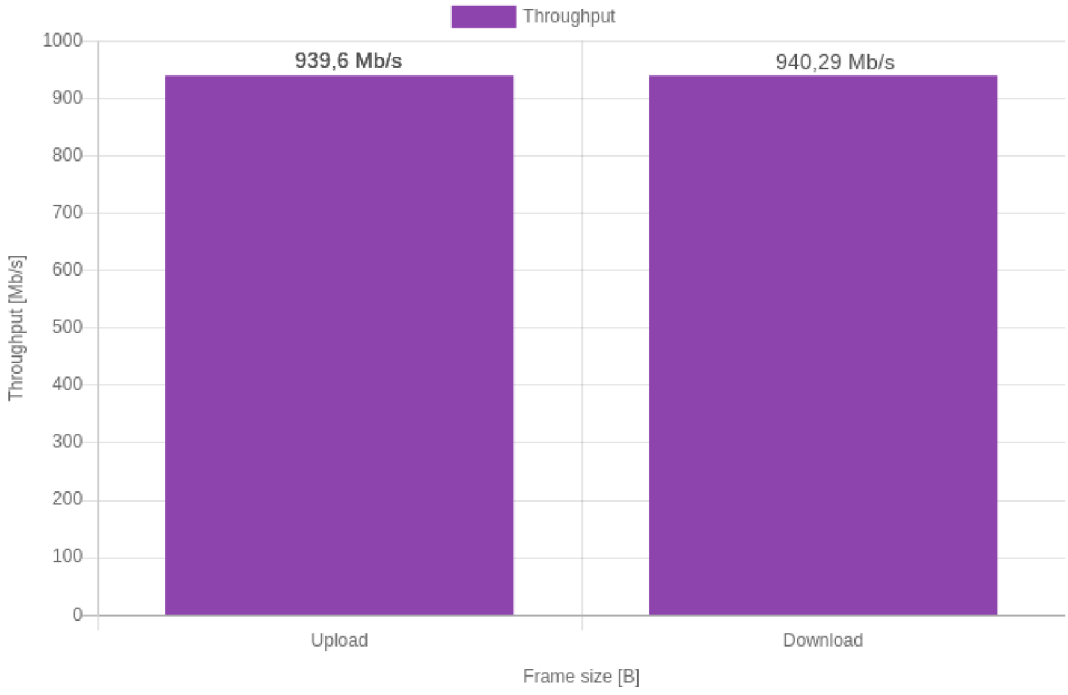
TCP Metrics (upload):

iTT	Ideal transfer time:	59,4261 s
aTT	Actual transfer time:	60,038 s
TTR	Transfer time ratio:	1,0103
BD	Buffer delay [%]:	6,76 %

Obr. 5.3: Výsledky měření propustnosti TCP na zařízení s rozhraními Gigabit Ethernet



Obr. 5.4: Graf propustnosti TCP na zařízení s rozhraními Fast Ethernet



Obr. 5.5: Graf propustnosti TCP na zařízení s rozhraními Gigabit Ethernet

5.4 Propustnost UDP

5.4.1 Postup testování

Modul realizuje měření propustnosti UDP podle doporučení RFC 2544. V rámci testování je postupně měřena propustnost pro každou velikost rámců, které jsou v RFC 2544 pevně definovány. Nástroj iPerf3, který je pro toto měření využit, měří propustnost na 4. vrstvě (aplikační) modelu TCP/IP. Hodnoty pro ostatní vrstvy modelu jsou dopočítány. Výpočty spočívají v procentuálním zastoupení hlaviček pro každou velikost rámce.

Pokud je dána velikost rámce 128 B, pak je velikost dat v rámci pouze 82 B (zbylých 46 B tvoří záhlaví). Na třetí vrstvě modelu (transportní) je vyhrazeno 8 B pro záhlaví protokolu UDP. Druhá vrstva modelu (síťová) alokuje 20 B pro hlavičku protokolu IP. Nejkomplikovanější vrstva je první vrstva – vrstva síťového rozhraní. Na této vrstvě jsou do ethernetového rámce přidány 4 B kontrolního součtu rámce (FCS), 12 B pro zdrojovou a cílovou MAC adresu, 7 B pro preambuli, 1 B oddělovače začátku rámce a 2 B označující typ rámce (*ethertype*). Každé dva rámce jsou navíc odděleny mezerou mezi pakety (*interpacket gap*), která má 12 bajtů. Ethernetový rámec je ve dvou vrstvách modelu ISO/OSI – fyzické a spojové, přičemž do fyzické spadá prembule, začátek rámce a mezera mezi rámci.

Hlavičky byly kontrolovány a zkoumány síťovým analyzátorem Wireshark. Při empirickém zkoumání bylo zjištěno, že Wireshark není schopen zobrazit fyzickou vrstvu rámce, jelikož přebírá data až po průchodu síťovou kartou a operačním systémem. Dále také neumí zobrazit FCS, který je ovšem na vrstvě spojové. U výpočtů je proto chápán jako celý rámec ethernetový rámec na spojové vrstvě. Preambule, oddělovač začátku rámce a mezery mezi pakety jsou zanedbány.

Jako příklad výpočtu může být uveden přepočet propustnosti z aplikační vrstvy na vrstvu transportní při velikosti rámce 128 B. Pokud byla při měření nástrojem iPerf3 získána propustnost 54,31 Mb/s (vrstva aplikační), propustnost na transportní vrstvě je vypočtena následujícím vzorcem:

$$\begin{aligned}\text{Propustnost L3} &= \frac{\text{Propustnost L4} * (\text{Velikost rámce [B]} - 46 + \text{Záhlaví UDP [B]})}{\text{Velikost rámce [B]} - 46} \\ &= \frac{54{,}31 * (128 - 46 + 8)}{128 - 46} \\ &= 59{,}64\,\text{Mb/s}.\end{aligned}$$

Výpočet pro první a druhou vrstvu modelu TCP/IP probíhá analogicky, pouze jsou připočteny i záhlaví těchto vrstev (20 B na vrstvě síťové a 18 B na vrstvě síťového

rozhraní). Vypočtené hodnoty propustností nižších vrstev mají spíše orientační charakter a nelze je srovnat s hodnotami naměřenými pomocí nástrojů, které pracují na těchto nižších vrstvách.

Na největší testované velikosti rámce (1518 B) může dojít k poklesu propustnosti oproti předešlé velikosti rámce (1280 B). Případný pokles propustnosti je způsoben fragmentací datagramů zařzením DUT. Směrovač fragmentuje datagramy, které jsou vyšší, než jeho hodnota MTU (což je 1500 B ve výchozím stavu). Při fragmentaci datagramů dochází na DUT k nárůstu zatížení procesoru, vyššímu zpoždění a také může dojít k snížení propustnosti. Fragmentace může být zobrazena například pomocí síťového analyzéru Wireshark. Nejvyšší propustnosti bývá často dosaženo při použití velikosti rámce 1280 B, jelikož nedochází k fragmentaci datagramů a záhlaví datagramu zabírá nejmenší část v poměru k celé velikosti zasílaného datagramu.

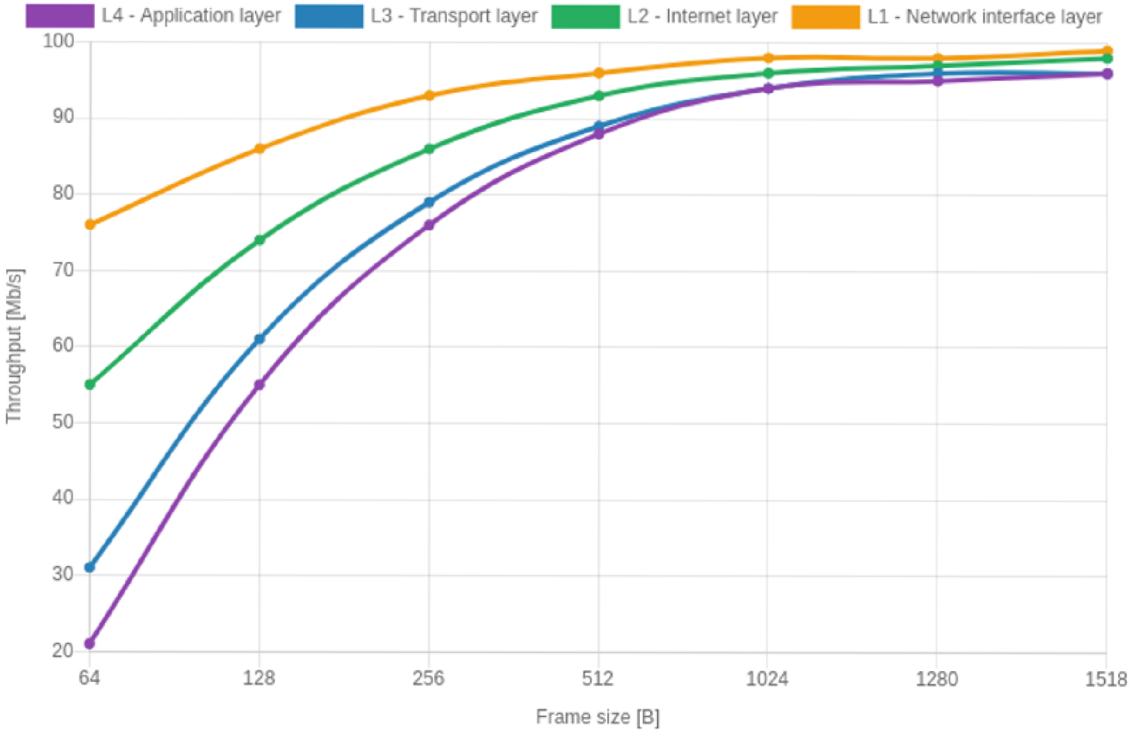
5.4.2 Měření a zhodnocení výsledků

Měření propustnosti UDP probíhalo na DUT Cisco (1 Gb/s) a Tenda (100 Mb/s). Výsledek měření propustnosti UDP je zobrazen na obrázku 5.6 pro DUT Tenda a obrázku 5.6 pro DUT Cisco.

Z výsledků měření je patrné, že s rostoucí velikostí rámce roste i dosažitelná propustnost. Na vzrůstající rychlost mají vliv hlavičky rámců. Při velikosti rámce 64 B je payload pouze 18 B, hlavička tedy zabírá téměř 72% rámce. Navíc je mezi každým rámcem oddělení rámce (*interpacket gap*). Pokud je ale velikost rámce 1518 B, payload má 1472 B a hlavička tedy zabírá pouhé 3% rámce. Při větší velikosti rámce je tedy více prostoru pro payload, je zasíláno menší množství oddělení rámců a propustnost je vyšší.

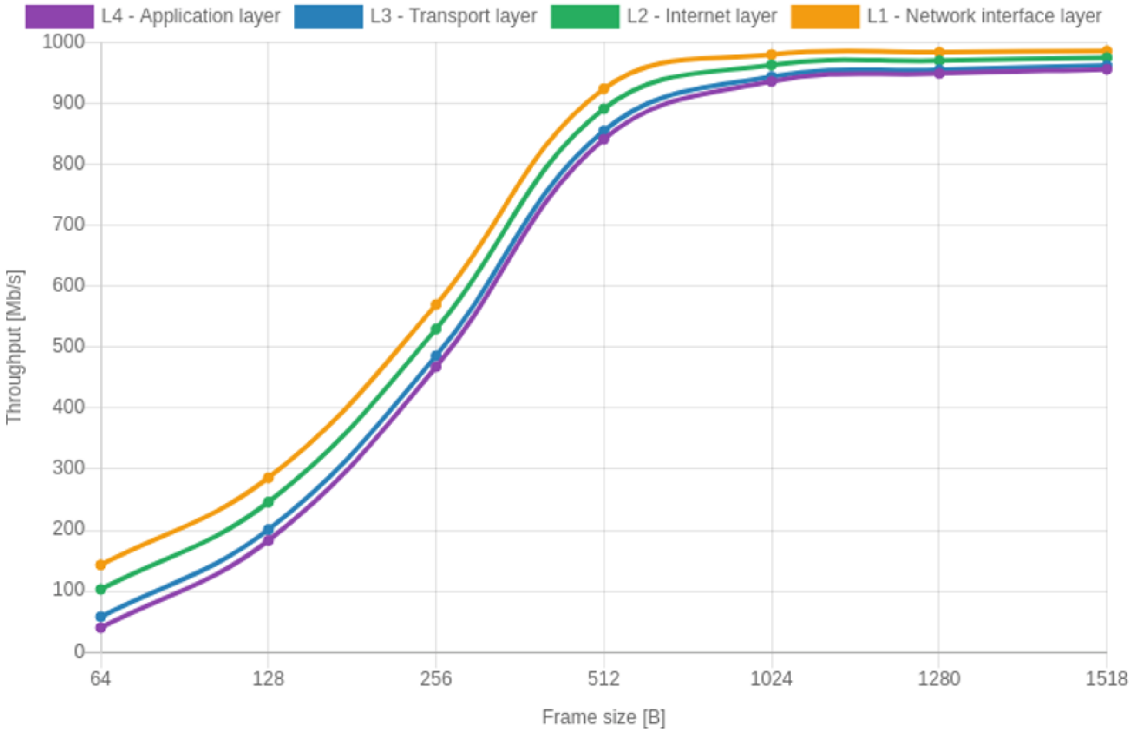
V laboratorních podmínkách bylo na vrstvě síťového rozhraní u Fast Ethernet směrovače dosaženo maximální propustnosti 98,69 Mb/s a na Gigabit Ethernet směrovače 986,31 Mb/s, což lze považovat za adekvátní výsledek blížící se fyzikálním limitům technologie.

Frame size	64 B	128 B	256 B	512 B	1024 B	1280 B	1518 B
L4 - Application layer:	21,43 Mb/s	55,39 Mb/s	76,08 Mb/s	87,59 Mb/s	93,66 Mb/s	94,92 Mb/s	95,7 Mb/s
L3 - Transport layer:	30,95 Mb/s	60,79 Mb/s	78,97 Mb/s	89,09 Mb/s	94,43 Mb/s	95,53 Mb/s	96,22 Mb/s
L2 - Internet layer:	54,76 Mb/s	74,3 Mb/s	86,22 Mb/s	92,85 Mb/s	96,34 Mb/s	97,07 Mb/s	97,52 Mb/s
L1 - Network interface layer:	76,18 Mb/s	86,46 Mb/s	92,74 Mb/s	96,23 Mb/s	98,07 Mb/s	98,45 Mb/s	98,69 Mb/s



Obr. 5.6: Výsledky měření propustnosti UDP na zařízení s rozhraními Fast Ethernet

Frame size	64 B	128 B	256 B	512 B	1024 B	1280 B	1518 B
L4 - Application layer:	40,29 Mb/s	183,49 Mb/s	467,75 Mb/s	840,75 Mb/s	936,36 Mb/s	948,83 Mb/s	956,42 Mb/s
L3 - Transport layer:	58,2 Mb/s	201,39 Mb/s	485,57 Mb/s	855,18 Mb/s	944,02 Mb/s	954,98 Mb/s	961,62 Mb/s
L2 - Internet layer:	102,98 Mb/s	246,14 Mb/s	530,11 Mb/s	891,27 Mb/s	963,17 Mb/s	970,36 Mb/s	974,61 Mb/s
L1 - Network interface layer:	143,27 Mb/s	286,42 Mb/s	570,21 Mb/s	923,74 Mb/s	980,4 Mb/s	984,2 Mb/s	986,31 Mb/s



Obr. 5.7: Výsledky měření propustnosti UDP na zařízení s rozhraními Gigabit Ethernet

6 Závěr

V rámci řešení bakalářské práce byly rozebrány přenosové parametry sítí dle standardů RFC 2544 a RFC 6349. Dále byl navržen a implementován zásuvný modul do programu Apache JMeter. Pro realizaci modulu bylo použito vývojové prostředí Eclipse IDE, které nabízí kompilátor Ant a je vhodné pro práci v jazyku Java.

V práci je popsána problematika a rozdíly s během zásuvného modulu v linuxových operačních systémech a systémech s Windows.

Vytvořené grafické uživatelské rozhraní modulu obsahuje vstupní pole, díky kterým může uživatel nastavit parametry testů, adresář výstupního souboru HTML a IP adresu cílové stanice. Vstupní pole zásuvného modulu byla ošetřena proti vložení chybných údajů. Při zadání nesprávné cílové adresy je zabráněno zacyklení nástroje iPerf3. Při zadání záporných nebo nulových časových hodnot dojde k jejich přenastavení na minimální hodnotu a vyrozumění uživatele o jejich změně formou konzolového výstpu i údaji v souboru HTML.

Zásuvný modul dokáže realizovat měření propustnosti TCP dle RFC 6349. Měření předchází výpočet počátečních hodnot, které jsou potřebné k nastavení parametrů vysílací a přijímací stanice. Před testem je takto stanoven počet současných spojení TCP, velikost minimálního TCP RWND, velikost maximální přenosové jednotky nebo hodnota BDP. Po dokončení měření propustnosti modul vypisuje metriky TCP.

Modul rovněž realizuje měření propustnosti UDP dle RFC 2544. Měřeny jsou všechny velikosti rámců dle definice v RFC 2544. V programu Wireshark byla zkoumána fragmentace při měření propustnosti u nejvyšší velikosti rámce. Po otestování propustnosti provádí modul přepočet propustností pro ostatní vrstvy TCP/IP modelu, jelikož nástroj iPerf3 pracuje na aplikační vrstvě modelu. Při tvorbě algoritmu na přepočet propustností bylo využito teoretických znalostí a programu Wireshark, pomocí kterého byla empiricky zkoumána a kontrolována přenášená data a záhlaví nástroje iPerf3. Výsledné hodnoty měření jsou přehledně vykresleny v tabulce i grafu, který je v souladu s RFC 2544.

Měření byla realizována na DUT s rychlostmi portů Fast Ethernet i Gigabit Ethernet. Získané výstupní soubory měření jsou přiloženy na CD.

Jako výstup měření zásuvným modulem Apache JMeter je vygenerován soubor HTML. Soubor vypisuje základní informace, hodnoty testů TCP i UDP v podobě přehledných tabulek a grafů. K vykreslení grafů byla využita open-source javascriptová knihovna ChartJs. Soubor HTML je generován z vytvořené šablony a po dokončení testu je automaticky otevřen.

Během vytváření zásuvného modulu byly objeveny limitace nástroje iPerf3. Reverse mode nedokáže zobrazit potřebné údaje (obousměrné zpoždění), a proto jsou měřené metriky TCP uvedeny pouze pro test propustnosti TCP odchozího spojení.

Obousměrný TCP test (současné odchozí a příchozí spojení) není kvůli limitacím nástroje iPerf verze 3 realizovatelný, nicméně tato funkcionalita byla podporována ve verzi 2. Verze 3 je novou implementací původního iPerf verze 2, která není zpětně kompatibilní a mnohým funkcím byla ukončena podpora. V případě rozšíření této bakalářské práce by bylo vhodné zvážit současné využití obou verzí nástroje, které by přineslo rozšíření funkcionalit zásuvného modulu.

Literatura

[1] BRADNER, S. Benchmarking Methodology for Network Interconnect Devices. *RFC 2544* [online]. Cambridge: Bradner, 1999 [cit. 2018-10-11]. Dostupné z: https://tools.ietf.org/html/rfc2544

[2] CONSTANTINE, Barry, Gilles FORGET, Ruediger GEIB a Reinhard SCHRAGE. Framework for TCP Throughput Testing. *Framework for TCP Throughput Testing* [online]. 2011 [cit. 2018-12-01]. Dostupné z: https://tools.ietf.org/html/rfc6349

[3] BRADNER, S. Benchmarking Terminology for Network Interconnection Devices. *RFC 1242* [online]. Harvard University, 1991 [cit. 2019-04-21]. Dostupné z: https://tools.ietf.org/html/rfc1242

[4] Apache JMeter. *Apache JMeter* [online]. Maryland: Apache Software Foundation, c1999-2018 [cit. 2018-10-16]. Dostupné z: http://jmeter.apache.org/

[5] DUGAN, Jon, Seth ELLIOTT, Bruce A. MAH, Jeff POSKANZER a Kaustubh PRABHU. IPerf - The ultimate speed test tool for TCP, UDP and SCTP. *IPerf: Test the limits of your network + Internet neutrality test* [online]. 2018 [cit. 2018-10-31]. Dostupné z: https://iperf.fr/

[6] BUIZA, Dani. Load testing with jMeter the ultimate guide. *Java Code Geeks* [online]. Španělsko: Exelis Media P.C., 2014, 2015 [cit. 2018-10-10]. Dostupné z: https://www.javacodegeeks.com/2014/11/jmeter-tutorial-load-testing.html

[7] H. HALILI, Emily. Apache JMeter: *A practical beginner's guide to automated testing and performance measurement for your websites.* Birmingham: Packt Publishing, 2008. ISBN 978-1-847192-95-0.

[8] *Metodika pro měření a vyhodnocení datových parametrů pevných sítí elektronických komunikací: Metodický postup* [online]. 2018, 21.12.2016, , 1-38 [cit. 2018-11-02]. Dostupné z: https://www.vnictp.cz/sites/default/files/obsah/novinky/metodika-mereni-vyhodnocovani-datovych-parametru-pevnych-komunikacnich-siti/soubory/metodika-pevne-site-ctu.pdf

[9] MATHIS, M a J HEFFNER. RFC 4821. *Packetization Layer Path MTU Discovery* [online]. Network Working Group, 2007 [cit. 2018-12-08]. Dostupné z: https://www.ietf.org/rfc/rfc4821.txt

[10] CHIMENTO, P a J ISHAC. RFC 5136. *Defining Network Capacity* [online]. JHU Applied Physics Lab: Network Working Group, 2008 [cit. 2018-12-08]. Dostupné z: https://tools.ietf.org/html/rfc5136

[11] Měření datových parametrů sítí pomocí TCP protokolu, verze 2.0, který je zveřejněn a je ze strany ČTÚ uplatňován v případě kontrolních měření na pevných i mobilních sítích. *Tiskopis ČTÚ* [online]. Praha: Český telekomunikační úřad, 2018 [cit. 2018-12-01]. Dostupné z: https://www.ctu.cz/sites/default/files/obsah/stranky/937/soubory/merenidatovychparametrusitipomocitcpprotokoluverze2.0.pdf

[12] ROHÁČ, Michal. *Optimalizace protokolu TCP* [online]. Ostrava, 2006 [cit. 2018-12-02]. Dostupné z: http://www.cs.vsb.cz/grygarek/TPS/projekty/0506Z/roh035-TCP-Optimization.pdf. Semestrální projekt. Vysoká škola Báňská.

[13] Eclipse desktop & web IDEs. *Eclipse desktop & web IDEs* [online]. Ottawa: Eclipse Foundation, c1999-2018 [cit. 2018-10-31]. Dostupné z: https://www.eclipse.org/ide/

[14] ŠVEHLÁK, Milan. *Rozšíření nástroje JMeter.* Brno, 2017. Diplomová práce. Vysoké učení technické v Brně. Vedoucí práce Ing. Zdeněk Martinásek, Ph.D.

[15] STOVER, Mike a Peter LIN. *How to Write a plugin for JMeter.* Maryland: Apache Software Foundation, 2005.

[16] Chart JS. *GitHub* [online]. GitHub, 2019 [cit. 2019-04-04]. Dostupné z: https://github.com/chartjs/Chart.js

[17] Bootstrap: The most popular HTML, CSS and JS library in the world. *Bootstrap* [online]. Bootstrap team, 2019 [cit. 2019-04-11]. Dostupné z: https://getbootstrap.com/

[18] World Wide Web Consortium (W3C). *Leading the web to its full potential* [online]. 2019 [cit. 2019-04-26]. Dostupné z: https://www.w3.org/

[19] *Wireshark* [online]. Wireshark Foundation, c1998-2018 [cit. 2018-12-10]. Dostupné z: https://www.wireshark.org/

[20] TP-LINK. *How to find the proper MTU size for my network* [online]. United States / English: TP-Link Technologies Co., 2019 [cit. 2019-04-11]. Dostupné z: https://www.tp-link.com/us/support/faq/190/

Seznam symbolů, veličin a zkratek

BB *Bottleneck Bandwidth* – nejnižší šířka pásma na celé testované trase sítě

BDP *Bandwidth-Delay Product* – součin obousměrného zpoždění a nejnižší šířky pásma trasy

DF *Don't Fragment* – zákaz fragmentace, bitový příznak ICMP pingu

DUT *Device Under Test* – testované zařízení

FCS *Frame Check Sequence* – kontrolní součet rámce vložený na konec ethernetového rámce

FTP *File Transfer Protocol* – protokol pro přenos souborů

GUI *Graphical User Interface* – grafické uživatelské rozhraní

HTML *Hypertext Markup Language* – značkovací jazyk určený pro tvorbu webových stránek

HTTP *Hypertext Transfer Protocol* – protokol určený k přenosu hypertextových dokumentů

ICMP *Internet Control Message Protocol* – protokol na síťové vrstvě pro odesílání služebních informací

IP *Internet Protocol* – protokol na síťové vrstvě pro směrování paketů

JVM *Java Virtual Machine* – sada počítačových programů a datových struktur, využívající virtuálního stroje ke spuštění dalších programů vytvořených pomocí jazyku Java

LAN *Local Area Network* – místní síť

MAC *Media Access Control* – jednoznačný identifikátor síťového rozhraní použitý na spojové vrstvě

MTU *Maximum Transmission Unit* – maximální přenosová jednotka

NAT *Network Address Translation* – překlad síťových adres

NUT *Network Under Test* – testovaná cesta na IP síti

OP *Operating System* – základní programové vybavení počítače

PID *Process Identifier* – jedinečné číslo procesu

QoS *Quality of Services* – protokol pro zajištění vyhrazení přenosové kapacity

RFC *Request for Comments* – řada dokumentů popisující internetové protokoly

RTT *Round-Trip Time* – obousměrné zpoždění

SCTP *Stream Control Transmission Protocol* – protokol transportní vrstvy, obdobný protoklům TCP nebo UDP

SLA *Service Level Agreement* – smlouva, kdy jsou parametry služby sjednány mezi poskytovatelem služby elektronických komunikací a

	zákazníkem
TCP	*Transmission Control Protocol* – protokol na transportní vrstvě pro spolehlivý přenos dat
TT	*Transfer Time* – čas přenosu
TTR	*Transfer-Time Ratio* – poměr mezi aktuální a ideální hodnotou přenosu
UDP	*User Datagram Protocol* – protokol na transportní vrstvě pro nespolehlivý přenos dat
WAN	*Wide Area Network* – rozlehlá síť
W3C	*World Wide Web Consortium* – mezinárodní sdružení dohlížející na vývoj internetových standardů

Seznam příloh

A Obsah přiloženého CD

- `/` ..... kořenový adresář přiloženého CD
  - `Analyzer.jar` ..... spustitelný jar soubor modulu
  - `BP.pdf` ..... bakalářská práce ve formátu PDF
  - `build.xml` ..... Ant skript pro kompilaci
  - `examples` ..... ukázky výstupních souborů měření
    - `fastEthernet.html`
    - `gigabitEthernet.html`
  - `src` ..... zdrojové soubory modulu
    - `Analyzer`
      - `AnalyzerSampler.java`
      - `AnalyzerSamplerGui.java`
      - `Data.java`
      - `DataParser.java`
      - `HtmlBuilder.java`
      - `ServThread.java`
      - `template.html`

B Návod k instalaci a obsluze modulu

Následující sekce popisuje úkony nutné k realizaci měření přenosových parametrů sítí pomocí zásuvného modulu Apache JMeter.

B.1 Instalace

Pro instalaci zásuvného modulu je nutno:

1. nainstalovat nástroj iPerf3 zadáním příkazu `sudo apt-get install iperf3`,
2. zkontrolovat přítomnost JRE pomocí příkazu `java --version`, v případě nutnosti lze JRE doinstalovat pomocí příkazu `sudo apt-get install openjdk-9-jre`,
3. stáhnout aplikaci Apache JMeter ze stránky `https://jmeter.apache.org/download_jmeter.cgi` (pro pouhé spouštění modulu stačí binární verze),
4. rozbalit stažený soubor ZIP do libovolného adresáře (například `/home/user/jmeter/`),
5. přemístit vytvořený modul `Analyzer.jar` do podsložky Apache JMeter `/home/user/jmeter/lib/ext/`.

Instalaci je nutno provést na obou stanicích (klient a server).

B.2 Měření

Pro spuštění modulu Apache JMeter je potřeba:

1. změnit umístění pomocí příkazu `cd /home/user/jmeter/bin/`,
2. spustit aplikaci Apache JMeter příkazem `./jmeter`,
3. v aplikaci kliknout pravým tlačítkem na `Test Plan` a přidat `Thread Group` pomocí `Add > Threads (Users) > Thread Group`,
4. kliknout pravým tlačítkem na `Thread Group` a vybrat `Add > Sampler > Analyzer`,
5. vybrat typ zařízení v poli `Device type`.

B.2.1 Server

Parametry nutné pro konfiguraci serveru:

- **Server iPerf3 port** – číslo portu vytvořeného serveru.

Server lze spustit tlačítkem `Start` a ukončit pomocí tlačítka `Stop`.

B.2.2 Klient

Obecné parametry nutné pro konfiguraci klienta:

- **Destination iPerf3 address** – adresa IP serveru,
- **Destination iPerf3 port** – číslo portu serveru,
- **Results file directory** – cesta k adresáři pro výstupní soubor HTML,
- **Link type (bottleneck)** – nejnižší šířka pásma měřené sítě (často omezena směrovačem).

Parametry nutné pro konfiguraci testu propustnosti TCP:

- **Enable TCP Throughput test** – aktivuje/deaktivuje test propustnosti TCP,
- **Pre-test duration** [s] – doba trvání testu nutného k určení parametrů před samotným měřením propustnosti,
- **Test duration** [s] – doba trvání měření propustnosti TCP v jednom směru,
- **Pause duration** [s] – doba čekání mezi dílčími měřeními.

Parametry nutné pro konfiguraci testu propustnosti UDP:

- **Enable UDP Throughput test** – aktivuje/deaktivuje test propustnosti UDP,
- **Test duration** [s] – doba trvání měření propustnosti UDP jedné velikosti rámce,
- **Pause duration** [s] – doba čekání mezi dílčími měřeními.

Měření lze spustit tlačítkem `Start`, po dokončení měření se modul sám ukončí.