**ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE**

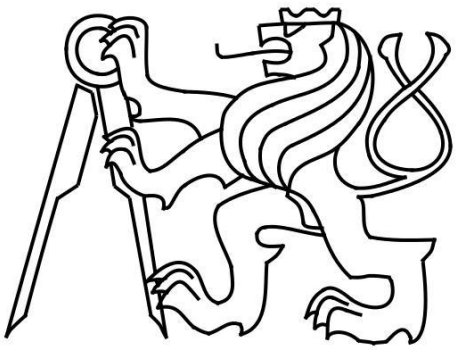
Fakulta elektrotechnická

# DIPLOMOVÁ PRÁCE

**2020** **Bc. Tomáš Straka**

ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE

Fakulta elektrotechnická
Katedra telekomunikační techniky



Diplomová práce

# Testbed IDS/IPS bezpečnostní sondy SonIoT

Testbed of IDS/IPS Security Probe SonIoT

Leden 2020

Diplomant: Bc. Tomáš Straka
Vedoucí práce: Ing. Bc. Marek Neruda, Ph.D.

# Čestné prohlášení

Prohlašuji, že jsem zadanou diplomovou práci zpracoval sám s přispěním vedoucího práce a konzultanta a používal jsem pouze literaturu v práci uvedenou. Dále prohlašuji, že nemám námitek proti půjčování nebo zveřejňování mé diplomové práce nebo její části se souhlasem katedry.

V Praze dne 7. 1. 2020 ……

Podpis diplomanta

# ZADÁNÍ DIPLOMOVÉ PRÁCE

## I. OSOBNÍ A STUDIJNÍ ÚDAJE

Příjmení: **Straka** Jméno: **Tomáš** Osobní číslo: **426065**

Fakulta/ústav: **Fakulta elektrotechnická**

Zadávající katedra/ústav: **Katedra telekomunikační techniky**

Studijní program: **Elektronika a komunikace**

Studijní obor: **Komunikační systémy a sítě**

## II. ÚDAJE K DIPLOMOVÉ PRÁCI

Název diplomové práce:

**Testbed IDS/IPS bezpečnostní sondy SonIoT**

Název diplomové práce anglicky:

**Testbed of IDS/IPS Security Probe SonIoT**

Pokyny pro vypracování:

Navrhněte a sestavte pracoviště pro testování vlivu použití IDS/IPS bezpečnostní sondy SonIoT, na vybrané provozní parametry dotčeného přenosového kanálu. Uvažujte provoz zejména při využití standardních přenosových IoT protokolů (MQTT, CoAP, …), určených pro datové toky s nižšími přenosovými rychlostmi. Zaměřte a prozkoumejte především ty parametry, které souvisí s QoS a QoE provozovaných služeb IoT a které může případná aplikace sondy ovlivnit.

Seznam doporučené literatury:

[1] ITU, Quality of Service Regulation Manual, 2017, ITU Thematic reports, dostupné na:http://handle.itu.int/11.1002/pub/8108e11f-en [on-line]
[2] K. Nowicki, T. Uhl, QoS/QoE in the Heterogeneous Internet of Things (IoT), 2017. In: Batalla J., Mastorakis G., Mavromoustakis C., Pallis E. (eds) Beyond the Internet of Things. Internet of Things (Technology, Communications and Computing). Springer, Cham
[3] L. Mejzrová a kolektiv, Projekt: Vývoj sondy pro preventivní ochranu IoT zařízení před pokusy o jejich převzetí, interní projektová dokumentace, FEL-ČVUT v Praze
[4] J. Vodrážka a kolektiv, Projekt: Flow Tester, interní projektová dokumentace, FEL-ČVUT v Praze

Jméno a pracoviště vedoucí(ho) diplomové práce:

**Ing. Marek Neruda, Ph.D., katedra telekomunikační techniky FEL**

Jméno a pracoviště druhé(ho) vedoucí(ho) nebo konzultanta(ky) diplomové práce:

**doc. Ing. Lukáš Vojtěch, Ph.D., katedra telekomunikační techniky FEL**

Datum zadání diplomové práce: **11.02.2019** Termín odevzdání diplomové práce: **07.01.2020**

Platnost zadání diplomové práce: **20.09.2020**

| Ing. Marek Neruda, Ph.D.<br>podpis vedoucí(ho) práce | podpis vedoucí(ho) ústavu/katedry | prof. Ing. Pavel Ripka, CSc.<br>podpis děkana(ky) |
|---|---|---|

## III. PŘEVZETÍ ZADÁNÍ

Diplomant bere na vědomí, že je povinen vypracovat diplomovou práci samostatně, bez cizí pomoci, s výjimkou poskytnutých konzultací. Seznam použité literatury, jiných pramenů a jmen konzultantů je třeba uvést v diplomové práci.

. _____
Datum převzetí zadání

_____
Podpis studenta

# Poděkování

Rád bych tímto poděkoval Ing. Bc. Marku Nerudovi, Ph.D. za cenné rady a konzultace týkající se diplomové práce v průběhu semestru.

**Anotace:**

Tato práce je zaměřena na návrh a sestavení pracoviště pro testování vlivu použití IDS/IPS bezpečnostní sondy SonIoT na vybrané provozní parametry dotčeného přenosového kanálu a parametry spojené s IoT službami. V teoretické části je obecně popsána problematika bezpečnosti v datových sítích se zaměřením na IDS/IPS systémy a sondu SonIoT, způsoby ověřování parametrů přenosového kanálu a také QoS v oblasti IoT s bližším zaměřením na datové IoT protokoly. V praktické části je popsáno sestavené testovací pracoviště (testbed), s pomocí kterého jsou následně provedeny testy prokazující vliv sondy na QoS parametry přenosového kanálu. Následně je zkoumán vliv sondy na datové IoT protokoly z pohledu bezpečnosti a vlivu na parametry QoS a QoE.

**Klíčová slova:**

bezpečnostní sonda SonIoT, IDS/IPS systémy, internet věcí, QoE, QoS, testbed

**Summary:**

The focus of this thesis is the design and construction of workspace for testing the influence of IDS/IPS security probe SonIoT on selected operational parameters of the affected transmission channel and parameters related to IoT services. The theoretical part contains general description of problematics of security in data networks with focus on IDS/IPS systems and probe SonIoT, means of verifying transmission channel parameters, and IoT services with closer look at data IoT protocols. The constructed testing workspace (testbed) is described in practical part. This is then utilised to execute tests, which prove probes influence on QoS parameters of the transmission channel. Afterwards the probes influence on data IoT protocols is examined in respect to security and effect on QoS and QoE parameters.

**Index Terms:**

IDS/IPS systems, Internet of Things, QoE, QoS, security probe SonIoT, testbed

# Obsah

# 1 Úvod

Téma „bezpečnost“ je dnes a denně skloňováno ve spojitosti s telekomunikačními sítěmi a službami, které tyto sítě umožňují provozovat. V dnešní době má přístup k celosvětově rozšířené síti Internet miliardy lidí, ať už je to z mobilních telefonů pomocí technologií 2G/3G/4G, či pomocí fixních přípojek typu DSL apod. Na internetu sdílíme svá data, převážně dobrovolně, avšak velmi často nastávají situace, kdy nám jsou data odcizena vlivem nedostatečného zabezpečení, které může být zanedbáno na mnoha úsecích komunikace od zdroje k cíli. Únik osobních, či firemních dat může v dnešní době, kdy má velká část populace svá osobní data uschována v rozsáhlých datových uložištích firem, představovat velký problém, ať už v podobě ztráty citlivých informací, nebo finanční ztráty a poškození dobrého jména firmy.

Jedním z bezpečnostních prvků, které přispívají k celkové obraně/ochraně sítí jsou systémy pro detekci a prevenci průniku. Ty mohou doplňovat stávající konfiguraci bezpečnosti sítě, která se většinou skládá pouze z firewallu a antivirového programu na koncové stanici. Jedním z takových prvků je i bezpečnostní sonda SonIoT, která v této práci představuje testovaný síťový prvek, a to ve dvou verzích – Home a Industry. Výkon každého softwaru je do jisté míry omezen výkonem hardwaru, který představuje platformu pro daný software, ať se jedná o výkon CPU, síťovou propustnost, či jiný parametr, který ovlivňuje výsledné chování zařízení.

Pro správnou funkčnost provozovaných služeb v dané síťové infrastruktuře je třeba, aby byly známy všechny omezující faktory využitých síťových zařízení, které mohou danou službu negativně ovlivnit. Jelikož je sonda SonIoT nově dostupným zařízením, zaměřuje se tato práce na prozkoumání těch omezujících faktorů, které mají vliv na dotčený přenosový kanál, a s tím spojenou funkčnost provozovaných služeb, především pak služeb v oblasti IoT.

Aby mohly být jednotlivé omezující faktory zjištěny a prozkoumány, bylo v rámci praktické části této práce navrženo testovací pracoviště (testbed), s pomocí kterého jsou zkoumány vlivy sondy na parametry sítě jako je propustnost, obousměrné zpoždění a ztrátovost paketů. Jako testovací nástroje jsou použity hned tři unikátní zařízení, které byly vyvinuty na Fakultě elektrotechnické ČVUT v Praze. Krom sondy SonIoT to je také zařízení FlowTester a aplikace FlowPing.

Pro měření vlivu sondy na parametry sítě byla navržena aplikace s grafickým prostředím, která monitoruje parametry jako je využití CPU, paměti RAM a síťové propustnosti přímo na ethernetových rozhraních sondy. Tyto parametry umožňují

identifikaci omezujících parametrů vlivem hardwaru během testování a poskytují tak cenné informace o dění v sondě během testování.

V poslední kapitole praktické části je zkoumán vliv sondy na komunikaci s využitím IoT datových protokolů. K tomuto účelu byly vytvořeny dvě aplikace. Pomocí jedné lze měřit vliv bezpečnostních procesů sondy na komunikaci s využitím IoT protokolů jako MQTT, CoAP atd. a druhá aplikace umožňuje měřit vliv sondy na celkové zpoždění doručení zpráv.

# 2. Testovací pracoviště – testbed

Testbed lze obecně označit za výzkumné, či experimentální pracoviště, jehož cílem je testování a inovace návrhů, metod, procesů, či technologií [1]. Ve slovníku výrazů pro softwarové testování [2] je testbed doslovně popsán jako: *„Prostředí nakonfigurované pro testování. Skládá se ze specifického HW, SW, operačního systému, síťové konfigurace, testovaného produktu a dalších systémových a aplikačních produktů*.“. Lze tedy říct, že testbed je kombinací hardwarových a softwarových prostředků v určité konfiguraci, která tvoří platformu pro testování produktu. Pro přiblížení tohoto pojmu lze zmínit některé běžné příklady [1]:

- Testbed softwaru (SW) – testbed v prostředí vývoje SW si lze představit jako testování v sandboxu, izolované zóně, která umožňuje vývojářům testovat nové funkce, aniž by tím ovlivnili chod již funkčního celku.
- Testbed v telekomunikacích – testbed se v telekomunikacích může objevovat např. v podobě testování routerů pro technologie 2G/3G/4G a aktuálně i pro 5G, před jejich uvedením do prodeje. Během testování v rámci testbedu mohou být simulovány různé reálné scénáře v laboratorních podmínkách, pomocí kterých lze následně odstranit problémy s routery, které se v průběhu testování objevily.
- Testbed v letectví – když jsou vyvinuty nové letecké motory, tak jsou na testovacím pracovišti testovány na reálném letadle, nejčastěji zároveň s původními motory, tzv. „flying testbed“, neboli „letecký testbed“.
- Testbed v zemědělství – v zemědělství se může vyskytovat testbed např. jako modelová farma, kde jsou testovány nové zemědělské techniky.

V rámci této diplomové práce je v praktické části popsáno zrealizované pracoviště pro testování bezpečností sondy SonIoT. Pracoviště je ukázáno v několika konfiguracích pro testování požadovaných parametrů. Při návrhu testbedu bylo dbáno především na opakovatelnost, transparentnost a správnost měření a získaných výsledků.

# 3. Síťová bezpečnost

Telekomunikační síť je dnes součástí téměř každé domácnosti, firmy i rozsáhlých nadnárodních společností. Například na činnosti celosvětové sítě Internet stojí mnohamiliardový byznys a s trochou nadsázky lze říct, že je součástí každodenního života nás všech a denně ovlivňuje naše životy. Téma telekomunikačních sítí je nesporně velmi rozsáhlé a jeho popis by vydal na mnoho diplomových prací, a proto se v této práci zaměřím pouze na téma „bezpečnost“, které je jeho neoddělitelnou součástí.

*„Všude tam, kde je dostupné připojení k síti, se vyskytují i bezpečnostní hrozby.“* [Soriano, 2017].

Bezpečnost sítě lze chápat jako proces, spíše než stav, který musí být neustále sledován, aby bylo možné udržovat požadovanou úroveň bezpečí sítě proti množství hrozeb. Ty vznikají denně a představují tak potencionální hrozbu, jelikož technologický pokrok jde kupředu velmi rychle a nechává za sebou prostor mezi nasazením technologie a mezi zajištěním všech bezpečnostních opatření [4].

Když se řekne pojem „potencionální hrozby sítě“, vybaví se většině lidí hackerské útoky (pokus o přístup k datům bez potřebné autorizace), avšak to je jen část potencionálních hrozeb. Mezi ty méně závažné hrozby můžou být zařazeny např. vliv přírodních jevu, či lidský faktor, především pak zaměstnanci, kteří svojí neznalostí mohou představovat hrozbu pro chráněná a citlivá data a údaje společnosti [4]. Závažnější hrozbu představuje nedbalost zaměstnanců či úmyslné krádeže dat, které mohou vést např. k pošpinění dobrého jména firmy či k odcizení citlivých dat, případně neúmyslnému vytvoření prostoru pro útoky z venkovní sítě a zjednodušit tak přístup k datům externím útočníkům [4].

K efektivnímu zabezpečení sítě by mělo být při jejím návrhu, či inovaci, přistupováno s úvahou všech možných rizik a útoků od přechodu z WAN (Wide Area Network) sítě, tedy sítě poskytovatele internetových služeb, do LAN (Local Area Network) sítě firmy, kde většinou začíná správa všech zařízení pověřenou osobou, přes komunikační kanál a aktivní/pasivní síťové prvky až po jednotlivá koncová zařízení v síti. Z opačné strany je nutné dbát na bezpečnost koncových zařízení a zajistit dostatečnou ochranu proti malwaru (viry, trojské koně, červy atd.), spywaru (SW pro odcizení dat ze zařízení bez vědomí uživatele) a adwaru (SW s obtěžujícím reklamním obsahem) [3]. Také je třeba zajistit bezpečnost přenášených dat od

koncových zařízení z LAN až k jejich cílům v rámci jedné LAN sítě a hlavně mimo ní, jelikož WAN sítě nebývají pod správou dané firmy, a tak nelze bez patřičných bezpečnostních úkonů zajistit jejich bezpečnost.

Mezi výhody, které přináší efektivní zabezpečení sítě, patří [3]:

- Důvěra zákazníka k bezpečnosti jeho dat.
- Mobilita – zabezpečený přístup bez narušení viry a jinými hrozbami.
- Vyšší produktivita – výrazně méně stráveného času se spamem a viry.
- Ekonomický přínos – výpadek sítě vede k možné finanční ztrátě.

Aktuálně je na trhu dostupné velké množství různých bezpečnostních produktů v SW i hardware (HW) podobě v cenových kategoriích od několika stovek korun až po statisíce korun, které jsou zaměřeny na různá odvětví bezpečnosti v síti. Produkty se stejným zaměřením, např. firewally, v různých cenových kategoriích sice zajišťují různou úroveň obrany, avšak není v jejich silách např. ochrana koncového zařízení proti škodlivým aplikacím, proto se využívá kombinace bezpečnostních produktů a jejich vzájemnou spoluprací lze docílit velmi efektivní obrany/ochrany sítě.

V Tab. 3.1 lze vidět porovnání bezpečnostních produktů běžné domácí sítě v porovnání s komplexní ochranou firemní sítě.

Tab. 3.1 Porovnání běžné a velmi dobré ochrany sítě, myšlenka převzata z [3; 4].

| Domácí síť | Firemní síť |
|---|---|
| Router se základním firewallem | Router s firewallem |
| Switch bez podpory QoS a bez podpory zabezpečení | Switch s podporou QoS a zabezpečení (DHCP snooping, port security a další) |
| Zabezpečení koncového zařízení (firewall, antivirový a antispywarový program | Zabezpečení koncového zařízení (firewall, antivirový antispywarový program) |
| | Systémy prevence a detekce průniku (IDS/IPS) |
| | Centrální AAA server (autorizace, autentifikace a účtování) |
| | Síťová monitorovací stanice |
| | Vlastní poštovní server |
| | Centrální firewall |
| | Zálohovací server |
| | VPN pro vzdálený přístup |

Vzhledem k zaměření této práce na bezpečností systém pro detekci (IDS) a prevenci (IPS) průniku je dále tato práce zaměřená na popis právě těchto systémů.

## 3.1 Systémy pro detekci a prevenci průniku

Se systémy pro detekci (Intrusion Detection Systems, zkratka IDS) a systémy pro prevenci (Intrusion Prevention Systems, zkratka IPS) průniku, dále jen IDS a IPS systémy, se často setkáváme ve spojitosti se síťovou bezpečností telekomunikačních sítí a v dnešní době tvoří nedílnou součást komplexní ochrany sítě. Jak jde vývoj technologií a služeb v oblasti telekomunikací neustále vpřed, tak vzniká také nový prostor pro zranitelnosti systémů a proti těmto zranitelnostem, resp. proti útokům využívající tyto zranitelnosti, je třeba chránit soukromá data s maximální snahou.

Jako příklad lze uvést situaci, kdy firewall blokuje přístup ke všem nežádoucím a nepovoleným portům, avšak nechává otevřené porty, které jsou využívány aplikacemi v dané síti, např. port 80, který je defaultně určen pro protokol HTTP (Hypertext Transfer Protocol), který je dnes a denně využíván pro přístup k webovým stránkám. Pokud by tedy útočník využil k útoku protokol HTTP, tak bude útok před firewallem skryt v legálním provozu [5].

Princip fungování IDS/IPS systémů je založen na monitorování síťového provozu z venkovní sítě, ale také uvnitř LAN sítě a identifikování podezřelých a škodlivých aktivit a jejich záznamu, tj. logování, pro následné vyhodnocení správcem systému.

Často se lze v literatuře na téma „IDS/IPS“ setkat s označením „nástroj IDS/IPS“, nebo „zařízení IDS/IPS“. Tento výraz představuje spojení IDS nebo IPS systému a hardwaru, který poskytuje systému výpočetní a síťovou kapacitu.

Dle [5] jsou specifické vlastnosti IDS systémů:

- Detekce útoků pocházejících od osob a programů.
- Zaznamenávat vzorce útoků pro následné zlepšení detekce.
- Generování a zaznamenávání upozornění na incident.
- Uchovávání záznamů incidentů pro případné budoucí vyhodnocování.

IPS pak navíc oproti IDS disponuje funkcí částečného, nebo úplného potlačení podezřelé aktivity, např. zahazováním paketů od/pro určité IP adresy [5].

Dnes lze na trhu nalézt mnoho variant IDS/IPS systémů, a to jak v softwarové variantě, tak i jako kompletní řešení v rámci SW/HW produktu. Mezi nejznámější open-source IDS/IPS projekty se v roce 2019 dle [6] řadí:

- Snort
- Suricata
- OSSEC
- Bro

- Sagan
- Samhain Labs

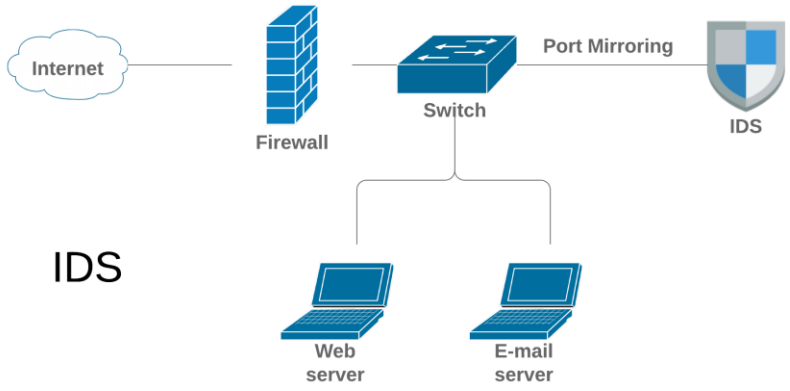
V nabídce komerčních IDS/IPS produktů lze nalézt např. tyto:

- IBM Proventia
- Juniper Networks IDP
- Cisco Secure IDS
- Sonda SonIoT

Z těchto systémů je blíže popsán v kapitole 3.2 IDS/IPS nástroj „Suricata“, který je součástí praktické části této diplomové práce a také je v kapitole 3.3 blíže popsán produkt „sonda SonIoT“, který byl vyvinut na katedře telekomunikační techniky Fakulty elektrotechnické ČVUT v Praze a jehož součástí je právě zmiňovaný systém Suricata.

### 3.1.1 IDS

Systémy pro detekci průniku lze označit jako efektivní sekundární ochranu, nebo také jako doplňkový prostředek posílení bezpečností sítě k běžným obraným prvkům sítě, jako je např. firewall, který je běžně také součástí routeru [5]. Vzhledem ke způsobu činnosti IDS systému bývá zařízení s IDS nejčastěji umístěno v tzv. demilitarizované zóně (fyzické, nebo logické vyčlenění zařízení do oddělené podsítě, zkratka DMZ) a veškerý provoz na něj bývá přeposílán pomocí tzv. port mirroringu, zrcadlením provozu z ostatních portů ve switchi na port IDS systému, viz. Obr. 3.1. Nutno podotknout, že v dostupné literatuře se o IDS systémech dozvídáme především jako o síťovém prvku (dále v textu bude uvedeno jako NIDS) a tak je také popisován, avšak dále v textu této práce je IDS systém popsán také jako uživatelsky orientovaná aplikace na hostitelské zařízení, např. osobní počítač.




Obr. 3.1 Schéma běžného zapojení sítě v režimu IDS. Firewall představuje kombinaci router/firewall.

U IDS systémů dochází k detekování hrozeb pomocí dvou primárních detekčních technik [7]:

- Detekce na základě anomálií (Anomaly-based detection)
- Detekce na základě pravidel (Signature-based detection)

Systémy většinou využívají jednu, nebo druhou techniku, ale mnoho dnešních IDS může pracovat s oběma technikami najednou [9].

**Detekce na základě anomálií**

Mechanismus detekce na základě anomálií je založen na principu strojového učení a vytvoření takového síťového modelu, který je považován za běžnou síťovou komunikaci. Počáteční síťový profil je vygenerován za určité časové období, nejčastěji za několik dní až týdnů [8]. Následná detekce neobvyklého chování v síti probíhá tak, že je síťový provoz porovnáván s vytvořeným síťovým modelem a pokud je nalezena odchylka, generuje se alarm s popisem [8]. Výhodou oproti detekci založené na pravidlech je schopnost rozpoznat hrozbu, která nebyla doposud analyzována a nenachází se tedy ani v databázi známých hrozeb, resp. v pravidlech. V tomto případě může být výhoda zároveň i nevýhodou, jelikož systém může na základě častých anomálií generovat také mnoho falešně pozitivních alarmů a znesnadnit tak ve velkém množství alarmu detekci opravdové hrozby [8].

**Detekce na základě pravidel**

Tento typ mechanismu je založen na porovnávání analyzovaných dat s databází známých vzorů jednotlivých hrozeb. Pokud je systémem nalezena shoda mezi kontrolovanými daty a vzorem z databáze, tak je vygenerován alarm [7]. Vzhledem k rostoucí povaze hrozeb je pro spolehlivost tohoto mechanismu důležité, aby

databáze s pravidly zůstávala neustále aktuální. Na internetu jsou k dispozici volně dostupné sady pravidel, které jsou kompatibilní s mnoha open-source IDS/IPS systémy, např. Suricata, či Snort.

Nevýhodou tohoto mechanismu může být fakt, že přístup k obecně známým a světově rozšířeným pravidlům může mít přístup i potencionální útočník a může si tak útok přizpůsobit, aby se detekci vyhnul. Další nevýhodou je fakt, že lze detekovat pouze ty hrozby, jejichž popis k detekci je uveden v pravidlech a také množství falešně pozitivních alarmů, které jsou generovány díky nedostatečně ošetřeným thresholdům pro detekci [9]. Threshold neboli práh, představuje hranici, např. množství opakujících se stejných paketů v určitém časovém intervalu, po jejímž překročení je alarm vygenerován.

IDS systémy lze také rozdělit do kategorií podle oblasti působnosti detekce na [8]:

- Síťově orientované IDS (Network-based IDS, zkratka NIDS)
- Uživatelsky orientované IDS (Host-based IDS, zkratka HIDS)
- Distribuované IDS (Distributed IDS, zkratka DIDS)

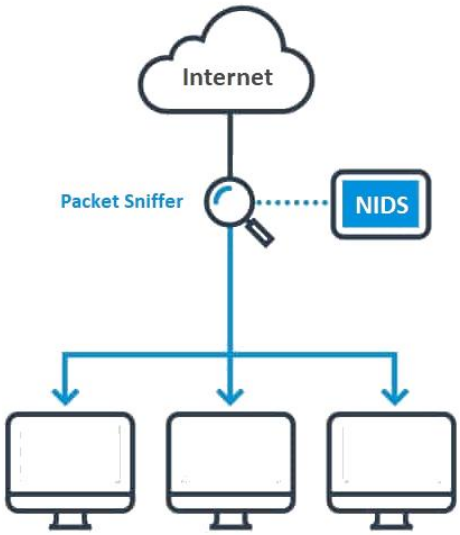
V některé literatuře se lze setkat také např. s pojmy [10]:

- Bezdrátové IDS (Wireless IDS, zkratka WIDS)

Ve většině dostupné literatury ale převažuje rozdělení pouze na NIDS, HIDS a DIDS.

**Sítově orientované IDS**

NIDS systémy pracují na principu analýzy síťového provozu dané sítě za účelem monitorování, detekce a logování podezřelé síťové aktivity. Nejčastěji se lze setkat s NIDS v podobě stand-alone zařízení, avšak můžeme se setkat i s implementací do síťového prvku, jako je např. switch Cisco Catalyst 3850. Obvykle je zařízení NIDS umístěno v demilitarizované zóně a systém musí pracovat v tzv. promiskuitním režimu. Promiskuitní režim zajišťuje, že systém bude přijímat a zpracovávat veškeré pakety, tedy i ty, které nejsou určeny pro něj [5]. Výhodou takového umístění je, že nedochází k omezování provozu v reálném čase, např. vytvořením bottlenecku (místa se sníženou síťovou propustností) ve struktuře sítě.




Obr. 3.2 Schéma zapojení NIDS v síti, převzato a upraveno z [6].

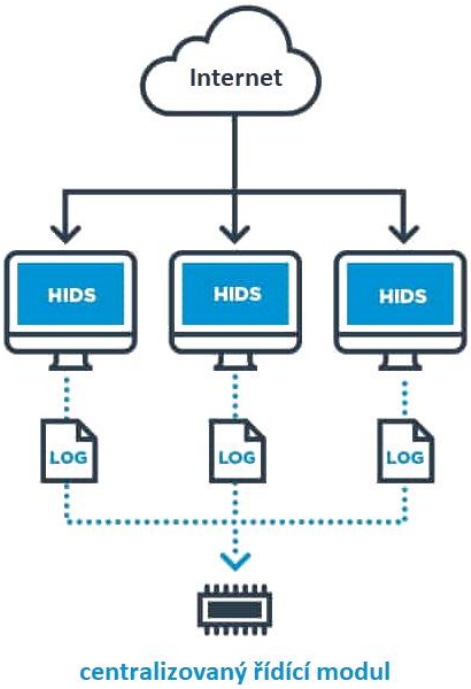
**Uživatelsky orientované IDS**

HIDS systémy, na rozdíl od NIDS systémů, monitorují podezřelé interní aktivity hostitelského zařízení. Většinou se jedná o aplikaci instalovatelnou na hostitelský operační systém (OS) [8]. Dle [11] HIDS zastávají především tyto aktivity na hostitelském zařízení:

- Vyhledávání neobvyklé, či škodlivé činnosti zkoumáním logů vytvořených OS.
- Vyhledávání změn v důležitých systémových souborech.
- Monitorování činností jiných SW aplikací systému.
- Monitorování síťového provozu hostitelského zařízení.

HIDS systém nemá vlastnost škodlivou aktivitu zastavit, ale stejně jako u NIDS by mělo dojít k vygenerování upozornění pro uživatele, či administrátora zařízení.

Typické je při použití HIDS systému, že se nachází na každém koncovém zařízení v dané síti, aby byla maximalizována bezpečnost a robustnost sítě a zároveň mohlo docházet k vzájemnému porovnávání hrozeb z jiných zařízení. Pro vyhodnocování z více stanic je vhodné mít v dané síti řešení v podobě centrálního sběrného a vyhodnocovacího bodu, Obr. 3.3 [6]. Funkci centrálního sběrného bodu podporuje např. HIDS open-source Samhain.




Obr. 3.3 Schéma zapojení HIDS v síti, převzato a upraveno z [6].

**Distribuované IDS**

Distribuované IDS systémy lze označit za skupinu spolupracujících systémů, které mohou být HIDS i NIDS v rámci jedné sítě a komunikují buď přímo mezi sebou, nebo pomocí centrálního bodu. Ten kromě centrálního řízení jednotlivých systémů vykonává také funkci sběru dat ze systémů a usnadňuje tak přehled o celkovém dění v síti jejímu správci [8].

**Bezdrátové IDS**

Dle [10] lze WIDS doslovně definovat jako: *„WIDS je jakékoli fyzické zařízení nebo softwarová aplikace, která aktivně monitoruje bezdrátovou síť na škodlivé činnosti a upozorní správce, když detekuje útok“.* Zařízení s WIDS systémem může tedy nabývat podobu pouhé softwarové implementace instalovatelné na existující zařízení v síti, či může být jako součást HW platformy, které je v síti jako stand-alone. Hlavním úkolem WIDS systémů je sledování radiového spektra pro včasnou detekci hrozby. Mezi WIDS dostupné na trhu se řadí např. produkty AirMagnet a AirDefense.

V článku o open-source IDS [11], který je z listopadu 2018, vytvořil autor CryptoCypher seznam nejznámějších IDS a rozdělil je do kategorií dle oblasti působení detekce:

- Síťově orientované IDS
    - Snort

- Suricata
- Bro

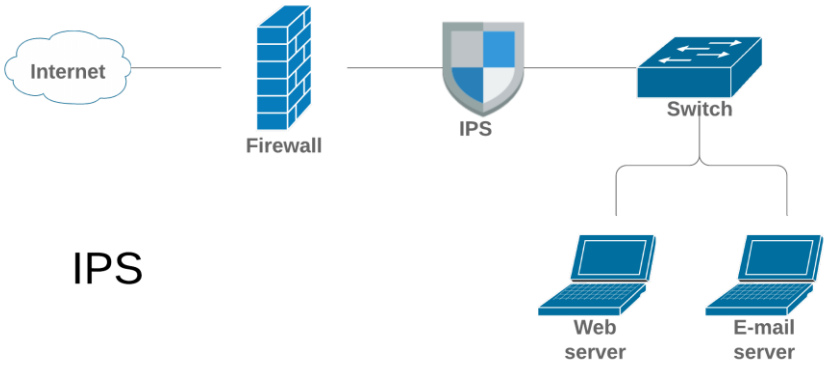
- Uživatelsky orientované IDS
  - OSSEC
  - Samhain

### 3.1.2 IPS

Systémy pro prevenci průniku, někdy také označovány jako „reaktivní“ IDS, jak již napovídá název, fungují na principu detekce hrozeb (stejně jako IDS), ale jsou také schopny částečně, nebo úplně omezit síťovou komunikaci při detekci hrozby zahazováním paketů, a tak zabránit možnému potencionálnímu útoku [8].

*„Hlavní rozdíl mezi IDS a IPS systémy je v tom, že IDS je monitorovací systém, zatímco IPS je kontrolní systém.“* [Petters, 2018].

IPS systémy jsou taktéž jako u IDS v literatuře prezentovány jako síťové prvky (dále v textu jako HIPS), avšak mohou nabývat také jiných podob, např. podobu uživatelsky orientovaného systému pro hostitelská zařízení.




Obr. 3.4 Schéma běžného zapojení sítě v režimu IPS. Firewall představuje kombinaci router/firewall.

IPS systémy jsou logickým vyústěním z IDS systémů, jelikož hrozba, která projde přes primární ochranu sítě, tedy firewall, je v IDS detekována, avšak bez patřičné akce hrozba není zastavena a doputuje do cílového bodu [8].

U IPS probíhá detekce hrozeb pomocí několika detekčních mechanismů [8]:

- Detekce na základě anomálií (Anomaly-based detection)
- Detekce na základě pravidel (Signature-based detection)

- Detekce pomocí stavové analýzy protokolů (Stateful protocol analysis detection)

Většina IPS systémů může využívat kombinace detekčních mechanismů, což vede k přesnějšímu detekování hrozby [8]. Detekční mechanismus na základě anomálií a pravidel byl již popsán v kapitole 3.1.1. Popis detekce pomocí stavové analýzy protokolů následuje.

**Detekce pomocí stavové analýzy protokolů**

Tento způsob detekce je založen na rozpoznávání odlišností v kontrolovaných protokolech. Protokol je po přijetí systémem porovnáván s obecně uznávaným profilem podoby protokolu neškodlivého charakteru a následně vyhodnocen [8]. V případě detekce škodlivého provozu je generován alarm a dle pravidel IPS je síťový provoz buď pouze zaznamenán, nebo je i zastaven. Na rozdíl od detekce založené na anomáliích, která se pro detekci přizpůsobuje charakteristickému chování uživatele/ů, pracuje tento mechanismus s univerzálními profily, které udávají, jaké podoby mohou dané protokoly nabývat a jak mohou být použity [8].

Stejně jako u IDS, i u IPS existuje několik různých typů systému dle jejich oblasti působení detekce v síti [8]:

- Síťově orientované IPS (Network-based IPS, zkratka NIPS)
- Uživatelsky orientované IPS (Host-based IPS, zkratka HIPS)
- Analýza chování sítě (Network Behavior Analysis, zkratka NBA)
- Bezdrátové IPS (Wireless IPS, zkratka WIPS)

**Síťově orientované IPS**

NIPS systémy, na rozdíl od NIDS systémů, umožňují zabránit hrozbě zahazováním paketů obsahujících potencionálně škodlivý obsah. Vzhledem k funkci zahazování paketů musí být zařízení s NIPS umístěno v síti tak, aby přes něj procházel buď částečný, nebo veškerý provoz (in-line), Obr 3.4. Kromě odhalení hrozby a následnému protiopatření by měly NIPS jednotlivé události logovat pro následné vyhodnocení administrátorem [8].

Vzhledem k in-line řešení NIPS systémů je pravidelná správa zařízení administrátorem velmi žádaná, jelikož bez prvotního přizpůsobení různých výjimek chování v síťové architektuře muže NIPS systém generovat mnoho falešných alarmů

a bez pravidelného korigování thresholdů by mohlo dojít k nechtěnému omezení provozu, což by mohlo např. negativně ovlivnit chod firmy [6].

NIPS systémy využívají několik technik pro zastavení hrozby [8]:

- Ukončení síťového připojení, nebo relace, která je používána k útoku.
- Blokování útočníka pro určitou koncovou stanici např. blokováním IP adresy, či jiného síťového atributu.
- Blokování veškerého provozu od/k útočníkovi.

**Uživatelsky orientované IPS**

HIPS systémy, stejně jako HIDS, monitorují podezřelé interní aktivity hostitelského zařízení. Na rozdíl od HIDS však mají tyto systémy možnost v případě hrozby, např. pokud chce škodlivý SW poškodit jiný SW, nebo změnit důležité systémové soubory, tuto akci zakázat, nebo si vyžádat od uživatele povolení k zákazu akce [8]. Proces kontroly může být spuštěn akcí uživatele na hostitelském zařízení, nebo může být prováděn periodicky. HIPS lze nalézt jako implementovanou funkci např. ve známém antivirovém programu ESET Internet Security.

**Analýza chování sítě NBA**

Tato analýza je založena na zkoumání síťového provozu pro detekování nežádoucího chování a nalezení hrozeb. Detekce probíhá na základě permanentního vyhodnocování statistik z provozu dané sítě. Oproti NIDS, HIDS a WIPS, které se zaměřují spíše na analýzu jednotlivých hrozeb, NBA monitoruje celkové chování zařízení v síti [12]. NBA je efektivní pro detekování hrozeb, které generují DDoS (Distributed Denial of Service neboli distribuované odepření služby) útoky, či např. pro detekování různých forem malwarů [8].

**Bezdrátové IPS – WIPS**

WIPS systémy stejně jako WIDS systémy kontrolují radiové spektrum, především ve standardu IEEE 802.11 dané sítě, avšak oproti WIDS disponují funkcemi pro eliminaci detekované hrozby [10]. Mezi typy hrozeb, které WIPS systém může detekovat a eliminovat patří např. tyto [13]:

- Cizí (rouge) Access Point (AP) – cizí AP v lokální síti pod správou cizí osoby.
- Neoprávněný připojení – snaha o přístup do LAN sítě neautorizované osoby.
- MAC spoofing – Odcizení MAC adresy již autorizovaného zařízení v LAN.
- Evil twin – Interní zařízení se bez jeho vědomí připojuje k externí síti.

## 3.2 Open-source IDS/IPS nástroj Suricata

Mezi nejznámější open-source IDS/IPS patří nástroje Suricata a Snort. Jelikož se v této diplomové práci objevuje Suricata jako IDS a IPS implementace v IDS/IPS zařízení „Sonda SonIoT“, tak je jejímu popisu věnována kapitola 3.2.1 a následně jsou v kapitole 3.2.2 porovnány oba zmíněné nástroje a popsány jejich rozdíly.

### 3.2.1 Suricata

Open-source nástroj Suricata, který vlastní a podporuje nadace OISF (Open Information Serucity Foundation) je rychlý a robustní nástroj pro detekci (IDS) a prevenci (IPS) průniku, který jako detekční metodu hrozeb využívá mechanismus detekce na základě pravidel (signatur) [15]. Suricata je krom základních a ručně vytvořených pravidel schopna využívat také např. specializovanou sadu pravidel Emerging Threats, která je poskytována zdarma. V placené verzi lze sadu nalézt pod názvem Emerging Threats Pro. Suricata je jako instalovatelný SW balíček dostupná pro OS Windows, macOS a také OS Linux. Mezi nejběžnější protokoly, které dokáže Suricata zpracovat pro následnou analýzu patří na aplikační vrstvě TCP/IP architektury [15]:

- HTTP, SSL, TLS, SMB, DCERPC, SMTP, FTP, SSH, DNS a další

A na nižších vrstvách.:

- IPv4, IPv6, TCP, UDP, SCTP, ICMPv4, ICMPv6 a další

Z dostupné dokumentace lze vyzdvihnout některé z významných vlastnosti Suricaty [17]:

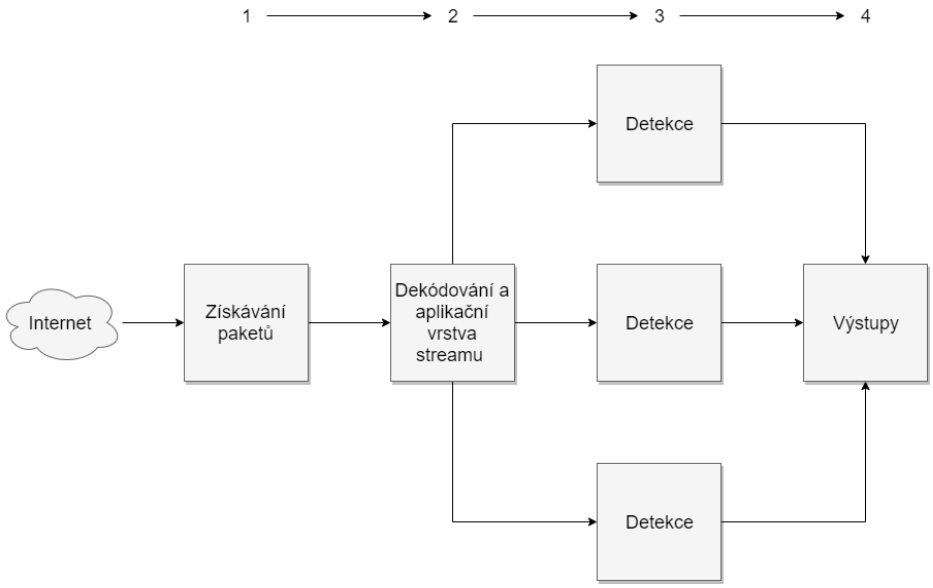
- Vícevláknový režim (multi-threading)
- Vysoký výkon
- Automatické detekce protokolů
- Engine pro monitorování síťové bezpečnosti (Network Security Monitoring engine, zkratka NSM)
- Vstupy a výstupy v čitelné podobě (dané formáty)

**Vícevláknový režim (multi-threading)**

Plná podpora vícevláknového režimu v rámci procesu pro efektivní využití vícejádrových procesorů. Díky tomu je Suricata schopna zpracovávat velké množství

síťových paketů najednou. Zpracování probíhá ve čtyřech krocích, resp. ve čtyřech modulech, Obr. 3.5:

1. Získávání paketů – zachycení paketů na síťovém rozhraní.
2. Dekódování a aplikační vrstva streamu – dekódování dat a rekonstrukce původního datového toku.
3. Detekce – vícevláknová paralelní detekce podle pravidel (signatur).
4. Výstupy – zpracování výstražných upozornění na výstupech.




Obr. 3.5 Schéma vícevláknového zpracování paketů, převzato a upraveno z [17].

**Vysoký výkon**

Jedna instance Suricaty je schopna kontrolovat až multi-gigabitový provoz. Hlavním omezujícím elementem tak je výkonnost použitého HW.

**Automatické detekce protokolů**

Suricata je schopna automaticky detekovat některé aplikační protokoly jako je HTTP, DNS, FTP, TLS a další, pokud komunikují přes nestandardní porty. Pokud Suricata detekuje např. HTTP protokol, tak se aplikují pravidla pro HTTP bez ohledu na porty uvedené v pravidle.

**Engine monitorování síťové bezpečnosti (NSM)**

Suricata má také engine NSM k podpoře detekce, která je založená na shromažďování většího množství dat pro provedení následné detekce a analýzy. Mezi shromažďovaná data patří např.:

- Logování HTTP requestů

- Logování a ukládání TLS certifikátů
- Logování DNS dotazů/odpovědí

**Vstupy a výstupy**

Vstupní a výstupní data Surikaty jsou dostupné v běžně užívaných formátech, např. JSON, či YAML, a tím je zjednodušeno jejich další zpracování.

### 3.2.2 Porovnání Suricata a Snort

Jelikož jsou nástroje Suricata a Snort v konkurenčním postavení, je dále uvedeno tabulkové porovnání jednotlivých nástrojů, Tab. 3.2 a shrnuty nejdůležitější rozdíly.

Tab. 3.2 Porovnání vlastností nástrojů Suricata a Snort, převzato z [11; 18].

| | Suricata | Snort |
|---|---|---|
| **Zpracování dat** | Vícevláknové | Jednovláknové |
| **Služby navíc** | Extrakce souborů, Detekce pomocí stavové analýzy protokolů, Vestavěná hardwarová akcelerace | Velká podpora komunity, Plugin framework (možnost implementace z velkého množství pluginů) |
| **Ruční psaní pravidel** | Ano | Ano |
| **Snort VRT pravidla** | Ano | Ano |
| **Emerging Threats pravidla** | Ano | Ano |
| **Platformy** | Linux, Windows, MacOS | Windows, Linux |
| **NIDS/NIPS** | oboje | oboje |

Toto porovnání lze brát pouze jako stručný přehled, jelikož jednotlivých rozdílů by se dalo najít mnoho, např. v hloubce detekčních mechanismů. Obecně lze říct, že oba nástroje pracují na obdobném principu a taktéž generují obdobné výstupy. Ten nejvýraznější rozdíl lze vidět v řádku „Zpracování dat“. V době, kdy je běžné mít i v těch nejjednodušších výpočetních jednotkách vícejádrové procesory, je pro zefektivnění práce výhodné využívat systémy s podporou vícevláknového zpracování.

## 3.3 Sonda SonIoT

Sonda SonIoT je bezpečnostní sonda, zařízení s implementovanou funkcí IPS a s možností přepnutí na IDS a vice versa, vyvinuté na ČVUT v Praze. Je navržena jako síťově orientovaná (NIDS/NIPS) a lze ji specifikovat jako: *„Zařízení pro preventivní ochranu IoT zařízení v síti před pokusy o jejich převzetí.“* [19]. V závislosti na požadovaném využití sondy se sonda může nacházet ve struktuře sítě jako in-line prvek (IPS), nebo v demilitarizované zóně (IDS), kdy je na sondu veškerý provoz pouze přeposílán.

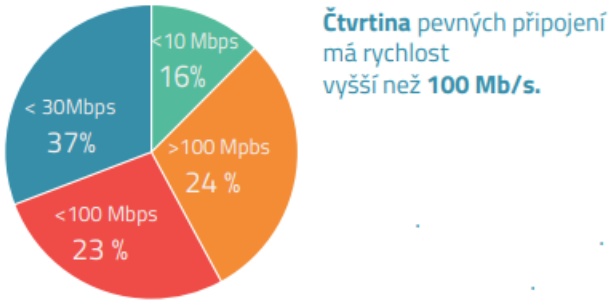
SW nástrojem pro detekci a prevenci sondy je open-source IDS/IPS nástroj Suricata. Mezi základní vlastnosti sondy patří [19]:

- Dvě HW varianty pro různá síťová řešení.
- Připojení do stávajících sítí standardu IEEE 802.3 a TCP/IP.
- Čistě lokální provoz sondy – detekce, rozhodování a vyhodnocování pouze v rámci sondy.
- Možnost lokální a vzdálené zabezpečené správy sondy.
- Ruční i automatický update pravidel.
- Správa sondy pomocí Android aplikace.

Sonda pracuje se sadou pravidel, které jsou periodicky 1x za den aktualizovány. Tím je zajištěna maximální ochrana sítě. Dostupná je ve dvou verzích – Home a Industry, a ty jsou dodávány v základní konfiguraci jako IPS, resp. NIPS s možností rekonfigurace do IDS, resp. NIDS.

### 3.3.1 SonIoT – verze Home

Dle dostupné dokumentace je sonda ve verzi Home dostupná se síťovou propustností v režimu IPS do 100 Mbit/s ve směru downstream (směrem k uživateli), což je rychlost, do které v roce 2017 dle ČTÚ (Český telekomunikační úřad) spadala cca 75 % fixních přípojek, Obr. 3.6.




Obr. 3.6 Poměr rychlosti fixních přípojek v roce 2017, zdroj ČTÚ [20].

A taktéž je nutno započítat mobilní internet od tří mobilních operátorů v ČR, Obr. 3.7. Tyto statistiky poskytuje ČTÚ pomocí své aplikace netmetr.cz [21].

| Operátor | Download | Upload |
|---|---|---|
| O2 | 25,13 Mb/s | 14,76 Mb/s |
| T-Mobile CZ | 21,57 Mb/s | 10,00 Mb/s |
| Vodafone cz | 26,89 Mb/s | 10,97 Mb/s |

Obr. 3.7 Medián naměřených hodnot od listopadu 2018 do listopadu 2019, převzato z [21].

K verzi Home je pro správu a vyhodnocování bezpečnostních incidentů k dispozici mobilní aplikace pro zařízení se systémem Android.

### 3.3.2 SonIoT – verze Industry

Verze Industry nabízí síťovou propustnost v režimu IPS do 300 Mbit/s. Jaké vhodné se jeví použití pro domácí síť, či pro síťovou infrastrukturu menší firmy. Pro tuto variantu je k dispozici pro správu a vyhodnocování bezpečnostních incidentů aplikace pro systém Android a pouze pro vyhodnocování je k dispozici také webové rozhraní.

# 4 Měření parametrů datových sítí

Dle portálu statista.com [22] bylo v říjnu 2019 na internetu aktivních necelých 4,5 miliardy uživatelů (osob), což je číslo, které bude nadále růst a s ním i množství internetových přípojek do domácností a firem. Na těchto přípojkách jsou provozovány služby od poskytovatelů internetového připojení (anglicky Internet Service Provider, zkratka ISP), kteří mají službu, ať už přístupu na internetu, nebo např. i IPTV, či VoIP služby, smluvně dohodnutou se zákazníkem a součástí smlouvy jsou také definované určité parametry služby SLA (Service-level agreement) jako je rychlost stahování (downstream), rychlost nahrávání (upstream) a případně další parametry. Tyto hodnoty rychlostí bývají často udávané jako maximální možné dosažitelné a ty se mnohdy velmi liší od těch reálně dosažitelných. To je jeden z mnoha důvodů, proč vzniklo několik standardů, které definují metody ověřování parametrů síťových prvků i celých sítí.

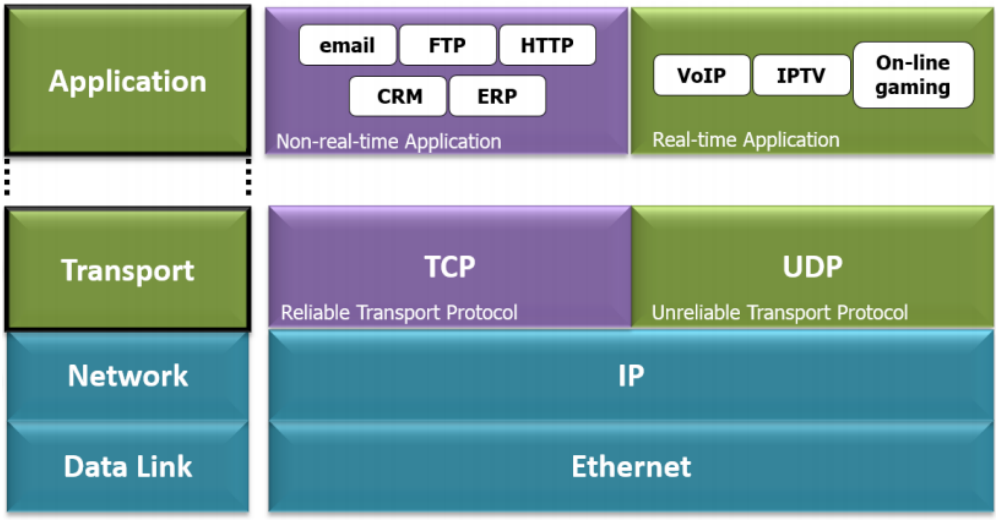
Vyhodnocování parametrů sítí můžeme také rozdělit podle těchto pohledů k dané službě [23]:

- Pohled provozovatele služby, např. služby internetu – parametry jako je propustnost, ztrátovost paketů, či zpoždění a kolísání zpoždění (QoS, Quality of Service)
- Pohled uživatele služby – kvalita požitku z dané služby (QoE, Quality of Experience)

Pokud bychom se zaměřili na jednotlivé služby, které lze na dané přípojce provozovat, mohli bychom je rozdělit do dvou hlavních kategorií na [24]:

- Real-time služby (VoIP, IPTV, On-line gaming apod.)
- Non-real-time služby (email, FTP, HTTP, CRM, ERP apod.)

Ty se liší především různorodou tolerancí vůči jednotlivým QoS parametrům, kdy např. zpoždění načtení stránky o vteřinu zákazník nezaznamená, tak zpoždění hovoru o stejnou dobu způsobí nesrozumitelnost hovoru. Často se také uvádí jejich rozdělení podle využitého transportního protokolu, Obr. 4.1 na TCP a UDP protokol.

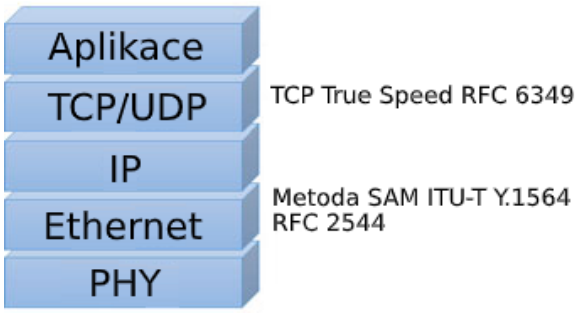



Obr. 4.1 Služby aplikační vrstvy TCP/IP, převzato a upraveno z [24].

V této práci je dále zaměřeno na měření pevných sítí a síťových prvků založených na TCP/IP architektuře vzhledem k využití sondy SonIoT, která je určena pro sítě využívající komunikační standardy IEEE 802.3, resp. IEEE 802.3xx.

## 4.1 Metody pro měření parametrů datových sítí a síťových prvků

Pro tuto práci jsou vybrány a popsány 3 metody – RFC 2544, ITU-T Y.1564 a RFC 6349, které jsou celosvětově rozšířené v oblasti měření datových sítí. Tyto známé metody mohou být aplikovatelné např. na pevné, či mobilní datové sítě, také s nimi lze měřit parametry jednotlivých síťových prvků, úseků sítí i celých síťových tras od poskytovatele až k zákazníkovi a vice versa. Lze je rozdělit podle vrstev na kterých probíhá měření parametrů, Obr. 4.2.




Obr. 4.2 Testy podle vrstev komunikace, obrázek vytvořen na základě [24].

### 4.1.1 IETF RFC 2544

Metoda RFC 2544 pro měření parametrů síťových prvků a datových sítí byla publikována v březnu 1999 a popisuje řadu testů, pomocí kterých lze měřit výkonost sítě [26]. Metoda, oficiálním názvem „Benchmarking Methodology for Network Interconnect Devices“ byla původně určena pro měření síťových prvků, nikoliv celých sítí nebo většího počtu služeb, avšak byla k tomu účelu přizpůsobena [27]. Testovat lze především tyto parametry [26]:

- Throughput – propustnost (rámce/byty za sekundu).
- Latency – zpoždění při přenosu.
- Back-to-back frames – Měří se maximální počet rámců přijatých při plné rychlosti linky, dokud není rámec ztracen.
- Frame Loss – ztrátovost rámců.

Metoda neumožňuje měření dnes běžně požadovaných parametrů jako je jitter (kolísání zpoždění) a SLA [27]. I když byla metoda přizpůsobena pro měření parametrů sítí, tak se dle doporučení IETF RFC 6815, celým názvem „Applicability Statement for RFC 2544: Use on Production Networks Considered Harmful“, doporučuje pro měření komplexních parametrů sítě využít např. metodu ITU-T Y.1564 [27].

### 4.1.2 ITU-T Y.1564

Tato metoda, celým názvem „Ethernet service activation test methodology“, zkráceně pak „EtherSAM“, byla publikována v roce 2011 s aktualizací v roce 2016 a hlavním rozdílem oproti metodě IETF RFC 2544 je, že umožňuje komplexní ověření kvality služeb podle SLA [27]. Metoda umožňuje definovat více datových streamů s různými parametry, které odpovídají reálným službám (VoIP, IPTV atd.) a ty jsou pak testovány pro jejich funkčnost na měřeném síťovém prvku, či úseku sítě [27]. Během měření jsou sledovány parametry jako [28]:

- Throughput – propustnost.
- Latency – zpoždění při přenosu.
- Jitter – kolísání zpoždění.
- Frame Loss – ztrátovost rámců.
- Max. doba výpadku služby.
- Burstability – Zatížitelnost spoje.

Testování probíhá ve dvou krocích [28]:

- Testování jednotlivých definovaných služeb pro ověření jejich správné konfigurace pomocí ramp testu (service configuration test) - testem se stupňovitě zvyšovanou přenosovou rychlostí.
- Testování všech služeb najednou pro ověření kvality jednotlivých služeb za delší časovou periodu (service performance test).

Výhodou této metody je, že umožňuje ověření funkčnosti služeb, především pak real-time služeb, na dané síťové architektuře ještě před reálným provozováním těchto služeb a tím např. umožňuje snížit náklady za síťovou přípojku, pokud by byla naddimenzovaná, nebo naopak předejít ztrátám při nefunkčnosti služeb v dané síti.

### 4.1.3 IETF RFC 6349

Tato metoda je, oproti přechozím zmíněným, založena na měření propustnosti sítě na transportní vrstvě TCP/IP architektury s využitím transportního protokolu TCP, který je využíván především pro non real-time služby jako přenos dat, FTP download/upload a obecně přístup k internetu [27; 29].

Sdružení BEREC (Body of European Regulators for Electronic Communications), které poskytuje administrativní a odbornou podporu Sdružení evropských regulačních orgánů v oblasti elektronických komunikací ve svém dokumentu [30] říká, že rychlosti downstream a upstream služby internet by měly být počítány na základě obsahu IP paketů, např. s využitím TCP protokolu transportní vrstvy.

Zásadní rozdíl oproti měření na vrstvě Ethernetu je fakt, že tyto testy nevypoví o propustnosti sítě tolik, co testy založené na metodě RFC 6349, jelikož mnoho protokolů aplikační vrstvy, např. HTTP, FTP, SMTP, POP3 a další, využívají jako transportní protokol právě TCP, u kterého je propustnost závislá hned na několika faktorech. Jedná se např. o velikost nastaveného TCP okna (TCP window size), velikost vyrovnávací paměti (buffer size) síťových uzlů, či na zpoždění paketů na síťové vrstvě, které způsobují snížení účinnosti přenosové rychlosti na transportní vrstvě [27].

Parametry a metriky metody, které lze měřit, či dopočítat [23; 29]:

- TCP throughput – velikost datového toku, který je měřen v určitém bodě sítě při využití TCP protokolu (v bitech za sekundu).
- BDP (Bandwidth Delay Product) – násobek kapacity datového spoje (v bitech za sekundu) a zpoždění mezi oběma konci spoje (v sekundách).

- RTT (Round-Trip-Time) – rozdíl času od odeslání prvního bitu zprávy příjemci po doručení posledního bitu příslušného potvrzení TCP segmentu.
- BB (Bottleneck Bandwitdh) – nejnižší hodnota přenosové kapacity celé měřené trasy.
- Send and Receive Socket Buffers – velikost vysílací a přijímací vyrovnávací paměti.
- Minimum TCP RWND (Receive Window) – velikost okna pro potvrzování přijetí paketů
- Path MTU (Maximum Transmission Unit) – maximální velikost paketu (bez nutnosti segmentace) použitelná pro datový spoj.

Ve specifikaci metody se však uvádí, že nelze provádět objektivní měření TCP propustnosti na nefunkční sítí, resp. na síti se zhoršenými parametry jako je vysoký jitter a zvýšená ztrátovost paketů. Dle specifikace se za nepřesné měření propustnosti považují hodnoty ztrátovosti paketů nad 5 % a jitter nad 150 ms. Z tohoto důvodu se doporučuje nejprve ověřit, zda jsou na měřeném úseku sítě tyto hodnoty nižší.

## 4.2 Parametry měření QoS v pevných sítích

ČTÚ na svých oficiálních stránkách představil dokument „Stanovení základních parametrů a měření kvality služby přístupu k síti internet“ [25], který popisuje postup ČTÚ v oblasti stanovení základních parametrů kvality služby (QoS) přístupu k síti internet a způsobu jejich měření a ověřování. Obecně řečeno, tento dokument popisuje, které měřitelné síťové parametry byly ČTÚ vybrány jako parametry, které popisují kvalitu nabízené služby přístupu k internetu. Také stanovuje vhodné měřící a vyhodnocovací metody a prostředky pro měření kvality služby, tzn. jestli je zákazníkovi poskytována služba ve sjednané kvalitě.

ČTÚ se při volbě parametrů zaměřovalo především na aspekty [25]:

- Srozumitelnost parametrů z pohledu běžného uživatele internetu.
- Aby parametry popisovaly službu internetu jako celek, nikoliv jen aplikace, které mohou na službě fungovat.
- Přihlédnutí k parametrům, kterými prezentují poskytovatelé své služby.

Nakonec byly vybrány 3 základní parametry, které určují QoS přístupu k internetu a to [25]:

- Propustnost
  - Download/Downstream – přenosová rychlost směrem k uživateli, vyjádřená v Mbit/s.
  - Upload/Upstream – přenosová rychlost směrem od uživatele, vyjádřená v Mbit/s).
- Delay (také v dokumentu jako RTT) - doba mezi odesláním prvního bitu segmentu TCP a příjmem posledního bitu odpovídajícího potvrzení segmentu TCP.

Tyto parametry, resp. jejich hodnoty představují pro koncového zákazníka služby základní přehled o kvalitě dané služby a představují tak i rozhodující parametry, které jsou předmětem kontroly. ČTÚ také definuje v dokumentu „Měření datových parametrů sítí pomocí TCP protokolu“ [31], že měření těchto 3 parametrů bude v pevných sítích prováděno pomocí TCP protokolu v souladu s metodou RFC 6349. ČTÚ také připouští možnost měření parametrů pomocí protokolu UDP na základě metodiky ITU-T Y.1564, avšak pouze pokud je měřená síť pod kompletní správou, nebo je měření koordinováno přímo s ISP. V případě, že je síť pod cizí správou se měření pomocí UDP nedoporučuje [27].

### 4.2.1 Metodika měření QoS přístupu k internetu dle ČTÚ

Účelem dokumentu [31] je, aby byla metodika vedena v obecné rovině takové, aby byla srozumitelná jak pro zákazníky, tak pro poskytovatele internetových služeb a také, aby byla oproštěna od měření na fyzické vrstvě a tím bylo měření nezávislé na použité přenosové technologii. Obecně řečeno, tato metodika je aplikovatelná i na jiné než jen pevné sítě. Samotná metodika vychází z doporučení IETF RFC 6349 a měření probíhá s využitím TCP protokolu na 4. transportní vrstvě referenčního ISO/OSI modelu.

Před samotným měřením by mělo být zajištěno [31]:

- Nezávislost měření (během měření negenerovat jiný provoz).
- Dostupnost služeb na očekávaných portech.
- Ověření traffic polingu (zda nedochází k vyloučení provozu na základě při překročení sjednanému limitu) a trafic shapingu (zda nedochází ke zpoždění, nebo vyloučení provozu určité služby).

Následně by měly být také identifikovány vhodné parametry pro nastavení měřícího zařízení, aby nedocházelo k omezení vlivem špatně zvolených parametrů. Mezi hlavní parametry, které je potřeba zjistit před měřením, nebo jsou součástí dalších výpočtů patří [31]:

- MTU (Maximum Transmission Unit) – maximální velikost IP paketu, který je možné přenést sítí.
- Měření RTT (Round-Trip-Time) – nezbytné k výpočtu BDP, TCP RWND a velikostí Socket bufferů.
- Měření BB (Bottleneck Bandwitdh), nebo jeho odhad v měřeném úseku.

Některá měřící zařízení umožňují nastavení pouze některých parametrů, jako je např. MTU a ostatní mohou být nastaveny napevno, nebo jsou laděny a měřeny měřícím zařízením v průběhu měření. Mezi populární open-source nástroje, které umožňují měřit TCP propustnost v režimu klient/server patří např. nástroj Iperf, viz kapitola 4.2.2.

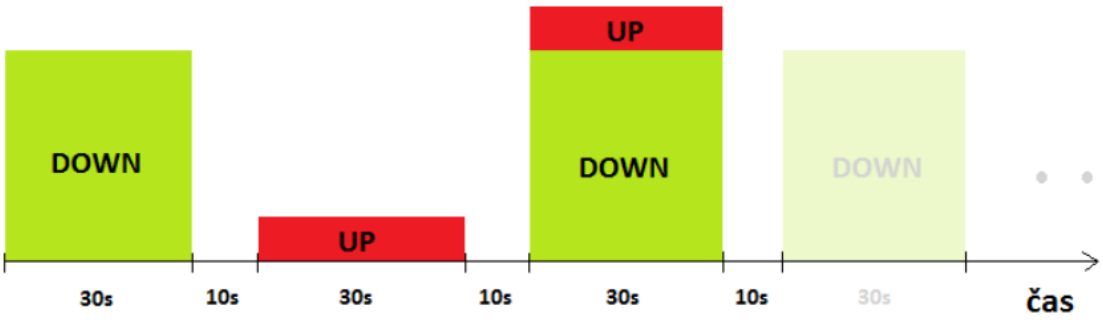
ČTÚ specifikuje v dokumentu sekvence měření, aby bylo dosaženo maximálního možného přesného výsledku TCP propustnosti a zpoždění, Obr. 4.3.

**Povinné sekvence**

1. min. 30 s – Downlink test
2. min. 10 s – Pauza
3. min. 30 s – Uplink test
4. min. 10 s – Pauza

**Volitelné sekvence**

5. min. 30 s – Downlink a Uplink dohromady
6. min. 10 s – Pauza




Obr. 4.3 sekvence jednotlivých testů, převzato z [31].

Tato sekvence není striktně vyžadována v takovéto podobě, lze libovolně měnit prodlevy a pořadí sekvencí, pokud je to vyžadováno [31].

Výsledkem z měření by měly být především tyto hodnoty:

- Hodnoty propustnosti pro každou testovanou hodnotu RWND, pokud není pevně nastavena.
- Volitelně i výsledky RTT.
- Údaje o místě, času, postupu, technologii a sekvenci měření.
- Údaje o nastavení měřícího zařízení.

### 4.2.2 FlowTester

Unikátní zařízení FlowTester, dále jen FT, které bylo vyvinuto týmem vědců z Fakulty elektrotechnické ČVUT v Praze, je určeno pro diagnostiku a monitorování sítí založených na TCP/IP architektuře s využitím znalostní databáze, pomocí které může FT predikovat problémy v sítích a zároveň poskytnout podklady pro návrh způsobu jejich řešení [32]. FT nabízí ověření parametrů jako je spolehlivost a výkonost, a to jak pro konkrétní službu, tak i pro síťovou infrastrukturu jako celek. Také poskytuje mnoho užitečných grafických a datových výstupů, které mohou přispět k nalezení síťových limitů, jako je např. odhalení bottlenecku, či pomoci při dimenzování síťové infrastruktury [33].

FT se také uplatňuje v oblasti IoT, např. v senzorových sítích menšího rozsahu, ale i velkých průmyslových sítí. V článku o FT [34] se lze také dočíst, že již v průběhu roku 2016 byl FT testován pro využití u technologií internetu věcí jako je LoRa a SIGFOX.

FT je využitelný v těchto oblastech [35]:

- Datové sítě
- Optické sítě
- Přístupové sítě (NGA)
- Vyhodnocování a měření QoS a QoE
- Smart Grids, Internet of Things, Industry 4.0
- Kyberbezpečnost

Také nabízí různé druhy testů, jako je stress test, test stability, či testování protokolu/služby. Testování a měření lze provádět pomocí nástrojů FlowPing a Iperf3 s využitím protokolů transportní vrstvy [35]:

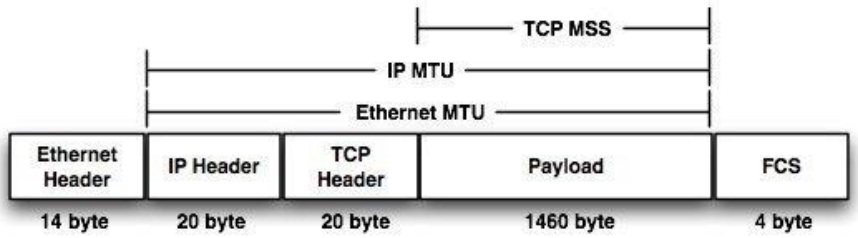
- TCP (měřeno: skutečná propustnost, RTT)
- UDP (měřeno: ramp test, RTT, ztrátovost paketů)

### Iperf3

Iperf, resp. jeho verze Iperf3, je open-source nástroj pro měření výkonnosti sítě, který je dostupný pro platformy jako je OS Windows a OS Linux. Je založen na principu klient/server a generované datové toky mohou být dle volby buď TCP, nebo UDP [37]. Mezi hlavní vlastnosti nástroje patří:

- Měření propustnosti sítě
- Reportování velikosti MSS (Maximum Segment Size), pokud není nastavena napevno
- Reportování velikosti MTU (Maximum Transmission Unit)
- Podpora velikosti TCP okna (TCP window size) skrze vyrovnávací paměť

Pro lepší pochopení pojmů MSS a MTU lze přihlédnout k Obr. 4.4, který jednotlivé pojmy graficky znázorňuje.




Obr. 4.4 Komunikační „jednotka“ přenášená v síti na technologii Ethernet (TCP), převzato z [36].

### FlowPing

FlowPing, dále jen FP, je volně dostupný nástroj pod licencí GNU GPLv3 a stejně jako Iperf umožňuje testování výkonosti sítě jako jsou zátěžové testy, či testy propustnosti sítě. FP se částečně podobá aplikaci Ping, ale místo protokolu ICMP pracuje s protokolem UDP v režimu klient/server. Na rozdíl od nástroje Iperf umožňuje FP vytvářet testy s proměnnou velikostí generovaného provozu [38]. Tato funkce dává prostor k testování propustnosti sítě, a s tím spojenou kvalitou real-time služeb, např. streaming videa, kdy je umožněno generovat např. rušivý provoz společně s daným streamem a lze pozorovat vliv jednotlivých nastavených parametrů testů na kvalitu videa a zároveň sledovat parametry jako je propustnost sítě, či ztrátovost paketů [39].

## 4.3 Kvalita požitku uživatele – QoE

Ve spojitosti s kvalitou služeb (QoS) se často setkáváme také s pojmem, který lze volně přeložit jako „kvalita prožitku“ (Quality of Experience, zkratka QoE).

*„Kvalita prožitku je subjektivním měřítkem prožitků zákazníka“* [Osipov, 2010].

Tyto dva pojmy jdou ruku v ruce, avšak ze dvou různých pohledů. Zatímco poskytovatel internetové služby řeší kvalitativní parametry přenosové trasy k zákazníkovi jako je propustnost, ztrátovost paketů a zpoždění, aby zákazníkovi zajistil smluvenou kvalitu služby, tak z opačné strany nahlíží zákazník na službu s očekáváním určité kvality a hodnotí, jaký má ze služby prožitek – QoE představuje určitou formu analýzy spokojenosti [40]. Jako příklad lze uvést situaci, kdy zákazník využívá IPTV na dané přípojce, která nedosahuje dostatečné propustnosti v požadovaném směru a zákazníkovi se tak obraz seká a kostičkuje – jeho prožitek ze služby je „špatný“. Často se lze setkat se studií kvality uživatelského prožitku při testování nových produktů a technologií před jejich uvedením do prodeje.

Hodnotitelem prožitku je tedy zákazník, či uživatel služby, jehož hodnocení je subjektivního charakteru, tzn. je zcela individuální, ovlivněno sociálním faktorem a předchozími zkušenostmi zákazníka se stejnou, či podobnou službou [40].

Trasu od zdroje služby (video, audio, VoIP apod.) až k cílovému uživateli můžeme identifikovat a následně rozdělit na pomyslné úseky, ve kterých může při průchodu docházet k ovlivnění uživatelského prožitku [41]:

- Kvalita služby (video/audio obsahu) ve zdroji, tzn. pokud je např. zhoršená kvalita videa jeho součásti.
- QoS parametry síťové trasy – propustnost, ztrátovost paketů, zpoždění, jitter.
- Lidský faktor – očekávání, ambice, přechozí zkušenost atd.

Výsledné QoE je pak dáno vlivy těchto tří úseků. O měření parametru QoS již bylo diskutováno v kapitole 4, resp. 4.2.

Pro vyhodnocení QoE lze využít hned několika metod. Jednou z nejznámějších je hodnocení pomocí metody MOS (Mean Opinion Score), která je definována normou ITU-T P.10 jako hodnota z předdefinované stupnice, pomocí které hodnotí uživatelé služby jejich prožitek [202]. Dále dle doporučení ITU-T P.800 existuje více druhů stupnic pro různé druhy experimentů, avšak nejčastěji se lze setkat s pětibodovou

stupnicí poslechu kvality, kde číslo 5 představuje hodnocení „vynikající“ prožitek a naopak číslo 1 vyjadřuje hodnocení „špatné“, Tab. 4.1. Minimální akceptovatelné hranici kvality odpovídá hodnota do 3,5 [41].

Tab. 4.1 Hodnoty stupnice MOS. Vytvořeno dle [43].

| MOS | Kvalita | Znehodnocení |
|---|---|---|
| **5** | Vynikající | Nepostřehnutelné |
| **4** | Dobrá | Postřehnutelné, ale neobtěžující |
| **3** | Průměrná | Mírně obtěžující |
| **2** | Nízká | Obtěžující |
| **1** | Špatná | Velmi obtěžující |

Měření a vyhodnocování QoE může být rozděleno na [42]:

- Měření přímou metodu
- Měření nepřímou metodu

**Přímá metoda** (také jako subjektivní metoda) je časově náročná a nákladná, jelikož spočívá v provádění testu se zákazníkem ohledně jeho pocitového vjemu ze služby, ale je také poměrně přesná [42]. Výsledná hodnota MOS dané služby je dána z aritmetického průměru všech hodnocení. **Nepřímá metoda** (také jako objektivní) naopak nevyžaduje účast zákazníků na měření, jelikož je založená na principu porovnání hodnot QoS a QoE – jaké limitní parametry QoS odpovídají limitním hodnotám QoE. Tato metoda ale vyžaduje určitě referenční scénáře, které identifikují závislost QoS na QoE službách [43]. Vyhodnocování QoE dle MOS lze vztáhnout nejen na testování kvality zvuku, ale obecně na audio/video služby a lze se setkat také s dalšími výrazy jako např. gMOS (Game Mean Opinion Score) model, který je určen pro subjektivní hodnocení uživatelského prožitku z mobilních her.

### 4.3.1 Vliv parametrů QoS na QoE

V kapitole 4.3 bylo zmíněno, že lze vyhodnocovat výsledné QoE pomocí objektivní metody. V literatuře se nejčastěji setkáváme s rozdělením provozovaných služeb na ty nejběžněji uživatelem provozované na internetu, Tab. 4.2. Každá z těchto služeb má nějaký „nedostatek“ v podobě kriticky důležitých parametrů QoS, při jejichž nedodržení, resp. při nedodržení minimálních limitů pro správnou funkci služby, dochází k degradaci služby, a s tím spojeného uživatelského prožitku (QoE).

Tab. 4.2 Parametry, které nejvíce ovlivňují jednotlivé služby. Převzato z [24].

| Služba/Aplikace | Datová přenosová rychlost | | Zpoždění | Jitter | ztrátovost paketů | chybovost paketů |
|---|---|---|---|---|---|---|
| | Download | Upload | | | | |
| **Online browsing (texty)** | ++ | - | ++ | - | +++ | +++ |
| **Online browsing (média)** | +++ | - | ++ | + | +++ | +++ |
| **Stahování souborů** | +++ | - | + | - | +++ | +++ |
| **Transakce** | - | - | ++ | - | +++ | +++ |
| **Streamování médií** | +++ | - | + | - | + | + |
| **VoIP** | + | + | +++ | +++ | + | + |
| **Gaming** | + | + | +++ | ++ | +++ | +++ |

-: irelevantní; +: mírně relevantní; ++: relevantní; +++: velmi relevantní

V knize „End-to-end QoS network design“ [44] z roku 2014 autoři sestavili seznam limitních QoS parametrů v souladu s patřičnými standardy pro jednotlivé služby v oblasti datových sítí, pomocí kterých lze určit také limity QoE. Pro zachování přehlednosti byly jednotlivé služby a parametry sepsány do přehledné tabulky, Tab. 4.3.

Tab. 4.3 Limitní hodnoty QoS/QoE, převzato a zpracováno do tabulky z [44].

| Služba/Aplikace | Jednosměrná latence [ms] | Jednosměrný jitter [ms] | ztrátovost paketů [%] | Min. šířka pásma |
|---|---|---|---|---|
| **VoIP** | <150 | <30 | <1 | Desítky – stovky kb/s |
| **Video Streaming** | <4000 | - | <5 | - |
| **Videokonference** | <150 | <30 | <1 | Stovky kb/s – jednotky Mbit/s |
| **datové služby (e-mail, FTP apod.)** | Variabilita podle služby, avšak požadujeme doručení všech paketů při využití TCP. | | | - |
| **IPTV*** | <100 | <50 | <0,001 | Jednotky – desítky Mbit/s |

* převzato z doporučení ITU-T Y.1541 [45]

Nejedná se o hodnoty, které by znamenaly nefunkčnost služby, ale spíše o doporučení limitů v souladu se standardy, které by měly být pro „srozumitelnost“ služby dodržovány.

# 5 QoS v oblasti IoT

Jak byl na začátku čtvrté kapitoly zmíněn počet internetových přípojek a počet uživatelů, tak v oblasti IoT (Internet of Things) se lze bavit o ještě výrazně vyšších počtech komunikujících zařízení, které společně generují enormní množství dat.

Pokud dnes mluvíme o službách provozovaných na internetu, stěží se můžeme omezit jen na tzv. „surfování na netu“, stahování a nahrávání obsahu a audio/video služby, jelikož tento seznam se v několika posledních letech, mimo jiné, rozrostl také o IoT služby, které nelze nikterak ignorovat, jelikož se staly součástí našeho každodenního života, ať už v podobě senzorových sítí, které řídí osvětlení, či určují zaplněnost parkoviště, či pomocí nich vzdáleně ovládáme naší chytrou domácnost. IoT tak najdeme např. v oblasti chytrých měst, Průmyslu 4.0, elektronického zdravotnictví, inteligentního zemědělství a v mnoha dalších oblastech. Samotný fakt, že pojem IoT se rozšířil do velkého portfolia odvětví, znamená, že vzniklo také mnoho nových služeb IoT s různými nároky na kvalitu služeb, resp. na jednotlivé parametry QoS.

Dle mezinárodní telekomunikační unie (International Telecommunication Union, ITU), resp. dle doporučení ITU-T E.800 lze QoS doslovně definovat jako: *„Souhrn vlastností telekomunikační služby, které souvisejí s jejich schopností uspokojovat stanovené a předpokládané potřeby uživatele služby.“* [46]. Laicky řečeno, nástroje QoS se užívají proto, aby nedocházelo ke snížení kvality uživatelských služeb, tedy např. aby bylo v dané síti upřednostněn datový tok video hovoru před stahováním dat, jelikož zpožděné a „kostičkované“ video je nekvalitní a v pojetí QoE snižuje uživatelský prožitek z provozované služby [47].

Zaměříme-li se na QoS nejen v IoT, ale v celé oblasti telekomunikačních sítí, je třeba přistupovat ke QoS jako komplexnímu řešení na jednotlivých vrstvách architektury. To znamená, že pokud bychom řešili QoS na aplikační vrstvě, tedy např. pouze jistotu doručení zprávy, mohlo by bez QoS na síťové vrstvě docházet např. k výrazné ztrátovosti paketů, a to by mohlo ovlivnit i snahu QoS na vrstvě aplikační.

QoS lze typicky kategorizovat podle přístupu k zajištění QoS pro danou aplikaci na [48]:

- Best Effort (bez QoS)
- Differentiated Services (soft QoS)
- Guaranteed services (hard QoS)

**Best Effort** (česky „nejlepší úsilí“) znamená, že služba je poskytovaná bez záruky kvality, tedy, že zákazník dostane takovou kapacitu, takové prostředky, které jsou dostupné. **Soft QoS** poskytuje kvalitu pro rozdílné služby. To znamená, že některá služba, např. VoIP dostane v síti přednost před stahováním dat. Nejčastěji je tato kategorie QoS v IP světě realizována pomocí tabulkové klasifikace paketů [49]. Hlavní výhodou soft QoS je jeho velká škálovatelnost. **Hard QoS** je založen na rezervaci zdrojů aplikace dle její žádosti po celou dobu daného datového toku. Aplikace tak získá záruku, že bude mít po celou dobu jistotu přiřazených zdrojů [49].

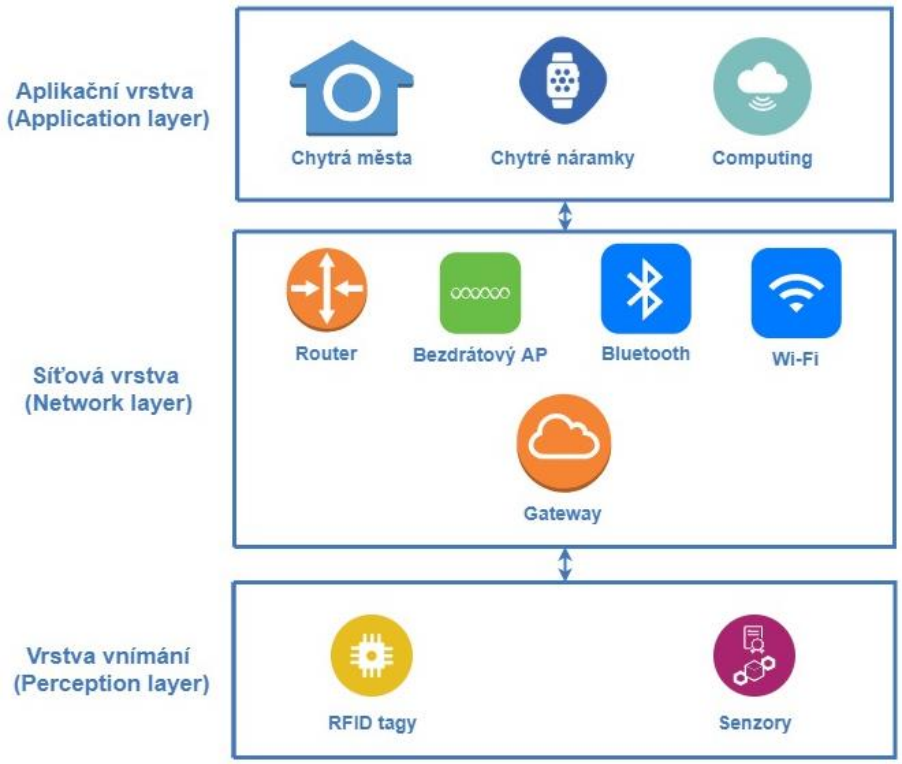
Aby bylo možné zajistit hard QoS v rámci IoT, je potřeba zajistit odpovídající mechanismy na každé vrstvě architektury IoT, od senzoru až k uživateli, neboť některé kvalitativní parametry jako je např. zpoždění se objevují v celém E2E (End-to-End) řešení.

Důsledkem zanedbání kvalitativního faktoru na některé z vrstev IoT architektury by mohla být snížena efektivita QoS na jiných vrstvách, a to by mohlo vyvrcholit až k nedodržení sjednaného SLA o kvalitě služby, či fatálním následkům u kritických aplikací, jako je automatické řízení vozidel, či zdravotní péče [48]. Existuje samozřejmě mnohem více kvalitativních faktorů, ke kterým je třeba přihlížet, např. bezpečnost, spolehlivost, či efektivita a další.

Autoři Manisha Singh a Gaurav Baranwal se v dokumentu „Quality of Service (QoS) in Internet of Things“ [50] z roku 2018 věnují definování běžných služeb v oblasti IoT a s nimi spojenými parametry QoS a v této kapitole bude z jejich práce čerpáno. Zjednodušeně lze říct, že existují dva druhy služeb/aplikací v oblasti IoT, které můžeme rozdělit podle požadavků na QoS parametry na [50]:

- Služby vyžadující propustnost, ale jsou tolerantní ke zpoždění.
- Služby citlivé na zpoždění vyžadující určitou šířku pásma s různými QoS požadavky.

Architektura IoT se neustále vyvíjí a nejčastěji se lze setkat s rozdělením na třívrstvý model, který je navržený po vzoru bezdrátových senzorových sítí (Wireless Sensor Network, zkratka WSN), Obr. 5.1. Nutno podotknout, že model architektury IoT neznamená jiný komunikační model, tedy např. referenční model TCP/IP.




Obr. 5.1 Třívrstvý model architektury IoT, vytvořeno na základě [50].

Nejnižší vrstvu představuje **vrstva vnímání (Perception layer)**, také někdy jako „vrstva věcí“. Ta představuje zařízení, které komunikují mezi sebou a dále pak pomocí síťového propojení, ať už drátového, nebo bezdrátového, předávají své data vyšším vrstvám [50]. Z dokumentu bych citoval poměrně jednoduchou a zcela vystihující větu, která popisuje „věci“ ve výrazu „Internet věcí“: *„Věci jsou inteligentní zařízení, senzory, lidské bytosti a avšechny objekty, které jsou si tohoto kontextu vědomy.“* [50].

**Síťová vrstva (Network layer)**, také jako „vrstva komunikace“ představuje komunikační spojení mezi vrstvou vnímání a aplikační vrstvou, ale obecně by šlo říct, že představuje spojení mezi vrstvou vnímání a okolním světem [50]. Věci mohou komunikovat mezi sebou a s okolním světem drátově, či bezdrátově a napřímo, tzn. přímo s určitým přístupovým bodem, nebo nepřímo pomocí brány, tj. gatewaye. Úkolem síťově vrstvy je kromě zajištění komunikace také zajištění bezpečnosti a neveřejnosti komunikace [50].

**Aplikační vrstva (Application layer)** představuje vrstvu, která definuje aplikace, které používají technologii IoT, nebo ty, ve kterých je nasazena. Probíhá zde také zpracování dat získaných od věcí (computing) a jejich patřičné uschování dle požadavků aplikace/uživatele [50].

Autoři v dokumentu [50] vytvořili rozsáhlý seznam QoS parametrů podle jednotlivých vrstev třívrstvého modelu IoT architektury, ke kterým je třeba přihlížet. V zájmu přehlednosti práce byly tyto parametry zpracovány do přehledné tabulky, Tab. 5.1. Jednotlivé limitní parametry QoS služeb nelze dobře, k množství služeb různorodého zaměření, specifikovat, avšak jako referenční rozdělení může v praktické části posloužit fakt, že služby mohou být přibližně rozděleny na ty s tolerancí zpoždění a ty bez tolerance zpoždění. V této práci je, vzhledem k využití IDS/IPS systému, zaměřeno především na část kvality služeb z pohledu komunikace (QoS Komunikace v Tab. 5.1).

Tab. 5.1 QoS parametry třívrstvého IoT modelu, vytvořeno na základě [50].

| QoS Věcí (QoS of Things) | QoS Komunikace (QoS of Communication) | QoS Výpočtů (QoS of Computing) |
|---|---|---|
| Hmotnost | Jitter | Škálovatelnost |
| Interoperabilita | Šířka pásma | Dynamická dostupnost |
| Flexibilita | Propustnost a efektivita | Spolehlivost |
| Dostupnost | Doba odezvy serveru | Finanční náklady |
| Spolehlivost | Finanční náklady | Doba odezvy |
| Celková přesnost | Dostupnost sítě | Kapacita |
| Dlouhodobá stabilita | Zabezpečení a soukromí | Zabezpečení a soukromí |
| Doba odezvy | Interoperabilita | Zákaznická podpora |
| Dosah | SLA | Zpětná vazba od uživatelů a recenze |
| Citlivost | Monitorování | |
| Přesnost | Spolehlivost | |
| Bezpečnost | | |
| Vzdálený update | | |
| Spotřeba energie | | |
| Drift (změna výsledků vlivem prostředí) | | |
| Podpora mobility | | |

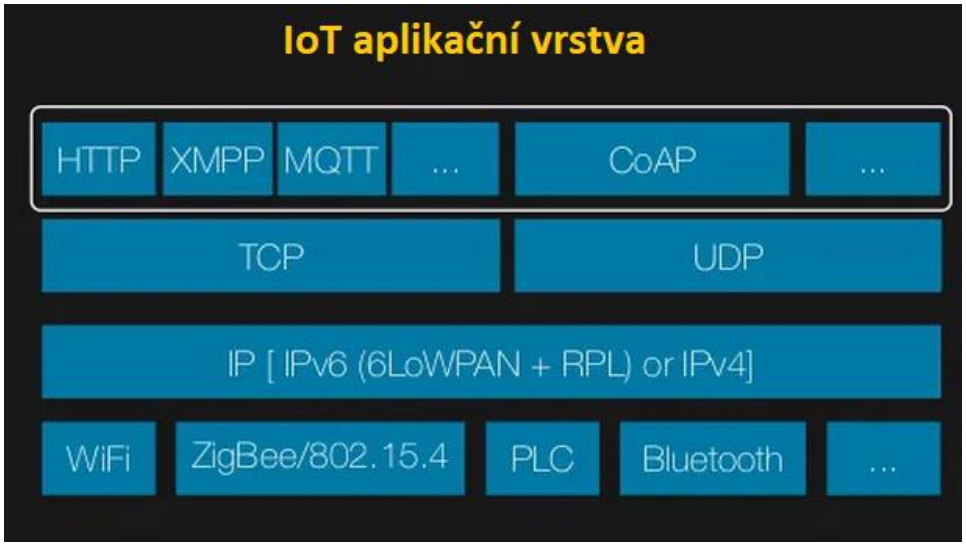
## 5.1 Datové protokoly aplikační vrstvy v IoT

Pro potřeby IoT, kde se předpokládá komunikace mezi zařízeními s omezeným zdrojem energie a výpočetní a přenosovou kapacitou, bylo třeba zavést datové protokoly nenáročné na komunikaci a jednoduché na implementaci do zařízení s omezenou pamětí a výpočetní kapacitou. V této práci je zaměřeno na datové protokoly využívající architekturu TCP/IP. Při pohledu na referenční ISO/OSI model představují datové protokoly vrstvy 5–7. Z pohledu modelu TCP/IP pak představují čtvrtou, aplikační vrstvu, Obr 5.2.

Dnes se v oblasti IoT vyskytují desítky datových protokolů. Mezi ty neznámější, které byly širokou veřejností přijaty jako komunikační standardy pro IoT, patří [51]:

- MQTT (Message Queuing Telemetry Transport)
- CoAP (Constrained Application Protocol)
- XMPP (Extensible Messaging and Presence Protocol)
- MQTT-SN (MQTT For Sensor Networks)

V následujících kapitolách jsou popsány jednotlivé výše zmíněné protokoly z obecného pohledu a blíže jsou pak popsány parametry QoS u jednotlivých protokolů na aplikační vrstvě. Pro QoS jednotlivých protokolů zde tedy nejsou řešeny parametry jako je propustnost, zpoždění paketů, jitter a jiné, ale řeší se doručitelnost jednotlivých zpráv mezi komunikujícími zařízeními.



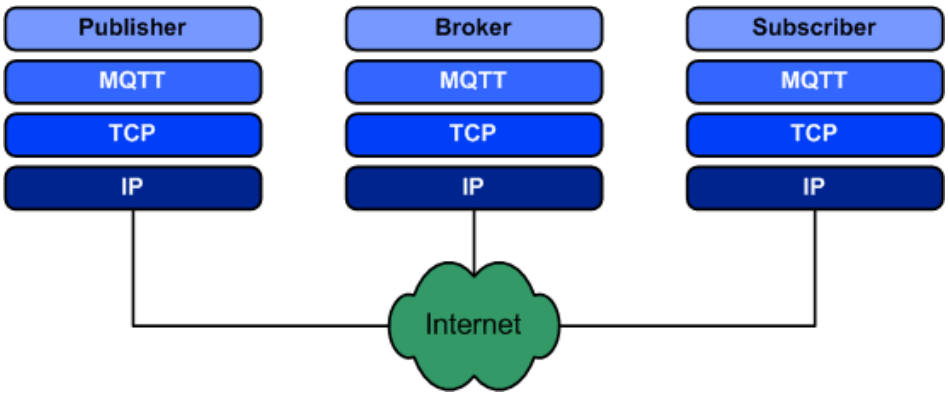

Obr. 5.2 Protokoly aplikační vrstvy, převzato a upraveno z [52].

## 5.2 Protokol MQTT

Protokol MQTT patří mezi komunikační standardy pro IoT. Jedná se o jednoduchý a nenáročný M2M (Machine to Machine) textový protokol pracující na principu předávání zpráv mezi klienty pomocí centrálního bodu, tzv. brokera [53].

Činnost brokera lze přiřadit k činnosti pošťáka. Přijímá zprávy od klientů, kteří se označují jako „publisher“ a předává je klientům s označením „subscriber“, Obr. 5.3. Zjednodušeně řečeno, publisher představuje určitý zdroj informací, o které se chce podělit s určitými subscribery. Jeden broker může mít teoreticky nekonečně velký počet publisherů a subscriberů. Reálně jsou klienti elektronické jednotky, např. publisher může být senzor teploty a subscriber může být aplikace v telefonu, která bude hodnotu teploty odebírat a zobrazovat. Broker obvykle bývá server implementovaný v lokální síti, nebo jako veřejný server dostupný na internetu. Díky své jednoduchosti je MQTT protokol lehce implementovatelný do zařízení s malou výpočetní kapacitou a zařízení, u kterých se dbá především na nízký datový tok a nízkou spotřebu energie pro komunikaci [53].

Tento protokol má velkou základnu podporovatelů pro využití jak v jednotlivých embedded zařízeních typu Arduino, Raspberry Pi apod., tak i v komplexních senzorových sítích. Pro snadnou implementaci do zařízení lze využít klientské knihovny pro nejrůznější programovací jazyky, jako jsou Python, Java, JavaScript, C a další. Existují také implementace brokerů pro MQTT, např. open-source Mosquitto MQTT broker.




Obr. 5.3 TCP/IP architektura MQTT komunikace, převzato z [53].

### 5.2.1 QoS pro MQTT protokol

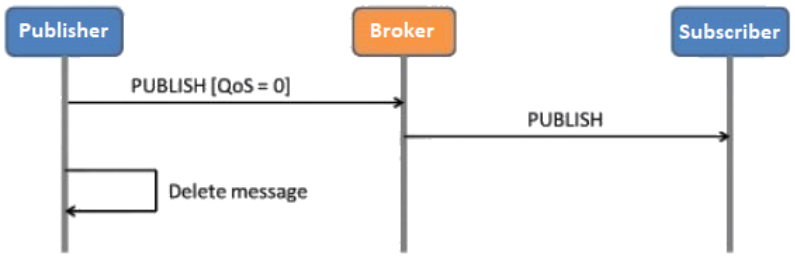
Pro protokol MQTT jsou definovány tři úrovně QoS. Tyto úrovně nabývají hodnot 0–2 a definují úroveň potvrzení zpráv v komunikaci mezi subscribery, brokerem a

Publishery [53]. Publisher i subscriber si volí sami, jakou úroveň QoS požadují. Každý subscriber oznamuje před samotným započetím komunikace brokerovi ve zprávě Subscribe jakou úroveň QoS požaduje. Broker vrací v potvrzovací zprávě SubAck potvrzení, že bude subscriberovi zasílat zprávy v takové úrovni QoS, kterou žádal.

Obecně platí, že broker přeposílá zprávu na takové úrovni QoS, se kterou zprávu přijal, avšak s možností úroveň snížit, pokud subscriber hlásí, že požaduje úroveň nižší [55].

**QoS úrovně 0** - Nejnižší úroveň QoS je označovaná jako „At most once“, tedy „nejvýše jednou“ nebo někdy také jako „fire and forget“ [55]. Znamená to, že zpráva je ze strany publishera odeslána k brokerovi bez zpětného potvrzení o doručení zprávy a poté je vymazána. Stejným způsobem je zpráva předána od brokera k subscriberům. Tato zpráva je označená jako Publish, Obr. 5.4.

Tato úroveň QoS má nejmenší datovou režii. Její užití se jeví jako vhodné, pokud máme stabilní a spolehlivé spojení mezi klienty a brokerem a zároveň požadujeme co nejjednodušší komunikaci, a s tím spojený nízký objem přenesených dat. Zároveň také nevyžadujeme potvrzení o doručení zprávy [55].



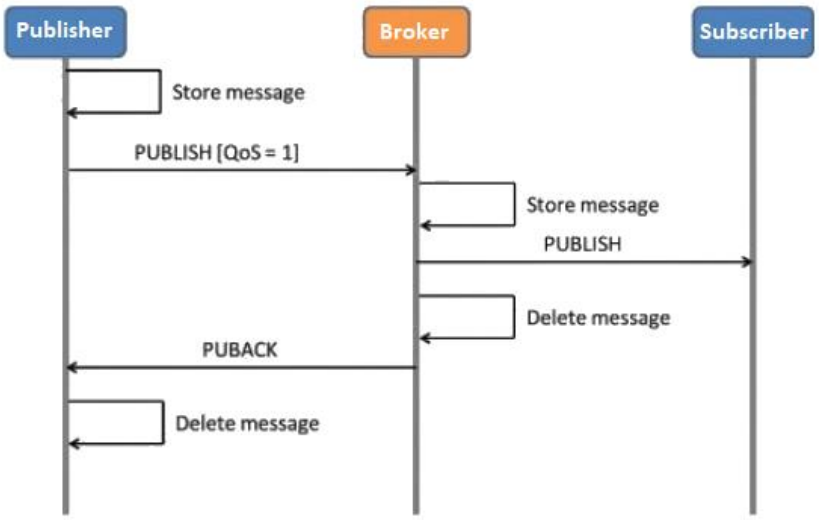

Obr. 5.4 QoS úrovně 0, převzato a upraveno z [52].

**QoS úrovně 1** - Prostřední úroveň QoS je označovaná jako „At least once“, tedy „alespoň jednou“. Znamená to, že je zpráva doručena alespoň jednou, ale může být doručena vícekrát [53]. Pokud v určitém časovém intervalu neobdrží publisher potvrzení o dodání, posílá zprávu znovu s hodnotou DUP = 1 (Duplicate delivery of a PUBLISH Control Packet), která značí, že se jedná o duplikát již zaslané zprávy.

Nejprve je zpráva uchována u publishera a zaslána na brokera jako Publish. Broker taktéž zprávu uchová a přeposílá jí ke všem subscriberům. Jakmile alespoň jeden ze subscriberů potvrdí doručení zprávy zprávou PubAck (Publish Acknowledgement), broker zprávu odstraní a jako PubAck posílá potvrzení publisherovi. Publisher ví, že zpráva prošla procesem definovaným QoS 1 a může zprávu zahodit. Přesné chování je závislé na implementaci, ale MQTT umožnuje oba

scénáře, tedy buď stačí k odeslání PubAck potvrzení pouze od jednoho subscribera nebo se čeká na potvrzení od všech nahlášených subscriberů [55], Obr. 5.5.

Úroveň QoS 1 je vhodné v implementaci využít, pokud chceme mít jistotu, že dostaneme každou zprávu, ale subsciber (aplikace, node apod.) se musí umět vypořádat s případnými duplikáty zprávy [56]. Pokud například přichází zprávy o teplotě během dne, duplikace stejné hodnoty v záznamu nám příliš nevadí, ale pokud například informujeme nějaký sčítač a počtu stlačení tlačítka, můžeme dostat ve výsledku velmi zkreslené hodnoty [56]. Jedna z možností potlačení takového chování je přidělení časové značky zprávám a pak duplikáty na úrovni aplikace potlačit [56].

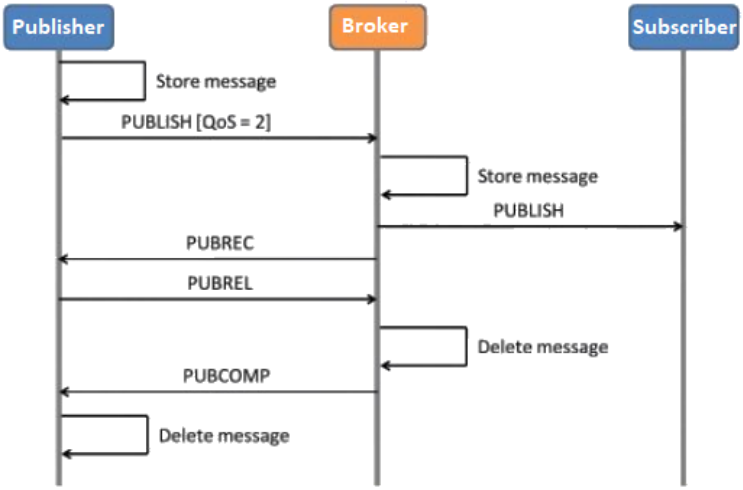



Obr. 5.5 QoS úrovně 1 (Sub. QoS 0), převzato a upraveno z [52].

**QoS úrovně 2** - Nejvyšší úroveň QoS je označovaná jako „Exactly once“, tedy „právě jednou“. Doslovný překlad odpovídá funkci QoS 2 a to tak, že je zajištěno, aby každá zpráva byla doručena právě jednou [53].

Komunikace probíhá tak, že prvně posílá publisher zprávu Publish k brokerovi. Ten jí vezme a stejně přeposílá na všechny subscribery. Broker ihned po odeslání zpráv subscriberům vrací publisherovi zprávu PubRec (Publish Receive) a tím hlásí, že zprávu přijal. Subscriber již zprávu znovu neposílá a vrací brokerovi zprávu PubRel (Publish Release). Broker přeposílá PubRel všem subscriberům a ti pak zpětně oznámí publisherovi přes brokera kompletní přenos zprávy zprávou PubComp (Publish Complete), Obr. 5.6.

S úrovní QoS 2 je spojená nejvyšší datová režie v porovnání s ostatními úrovněmi, proto je vhodné k ní přistoupit, pokud v našem řešení vyžadujeme, aby byla zpráva doručena právě jednou. Pseudo-úrovně QoS 2 lze dosáhnout také např. již zmíněným využitím úrovně QoS 1 s potlačením pomocí časové značky [56].




Obr. 5.6 QoS úrovně 2 (Sub. QoS 0), převzato a upraveno z [52].

## 5.3 Protokol CoAP

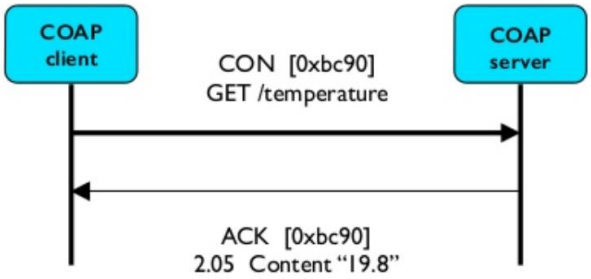
Protokol CoAP je jednoduchý M2M protokol z rodiny datových protokolů IoT. Vychází z protokolu HTTP, ale jedná se odlehčenou verzi, kde jsou textové hlavičky nahrazeny binárními hlavičkami s nižším počtem parametrů a jako transportní protokol využívá protokol UDP. Je vhodný pro implementaci do nízkoenergetických zařízení s omezenou výpočetní kapacitou a omezenou komunikací [57]. CoAP využívá model komunikace klient/server, tedy klient zasílá žádost směrem k serveru (request) a sever vrací odpověď (response). CoAP je stejně jako HTTP založen na architektuře REST (Representational State Transfer), která umožnuje klientovi přistupovat ke zdrojům na serveru užitím metod GET, PUT, POST a DELETE [57].

### 5.3.1 QoS pro CoAP protokol

Protokol CoAP umožňuje dvě úrovně QoS. Tyto dvě úrovně jsou definovány jako dva typy zpráv:

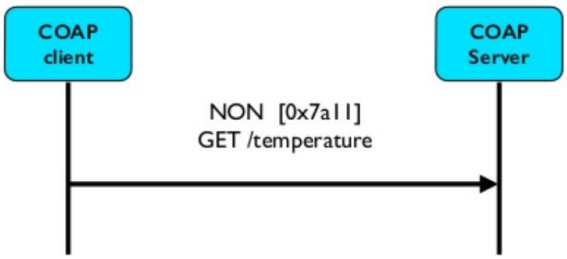
- Confirmable (CON)
- Nonconfirmable (NON)

V případě označení zprávy jako potvrditelné (zpráva CON) musí příjemce potvrdit doručení odesláním paketu ACK zpět odesílateli, Obr. 5.7. Zprávy v těle paketu obsahují Message ID, pomocí kterého lze poznat, na kterou zprávu je daný ACK odpověď, tedy Message ID CON zprávy a ACK zprávy musejí být shodné [57].




Obr. 5.7 QoS úrovně CON, převzato z [58].

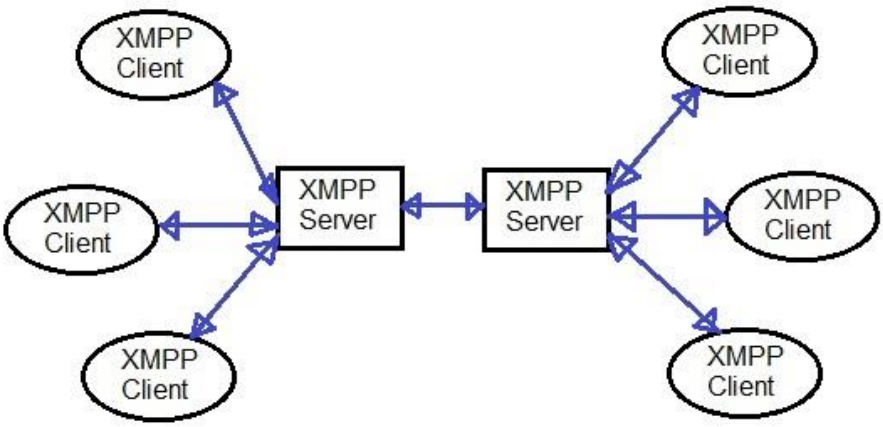
V případě označení zprávy jako nepotvrditelné (zpráva NON), se se odesílatel chová podobně jako v případě QoS úrovně 0 u protokolu MQTT, tedy scénář „fire and forget“, Obr. 5.8. Zpráva taktéž obsahuje Message ID [57].




Obr. 5.8 QoS úrovně NON, převzato z [58].

## 5.4. Protokol XMPP

Open source protokol XMPP, v anglickém jazyce „Extensible Messaging and Presence Protocol“, což lze dle [59] přeložit jako "*rozšiřitelný protokol pro posílání zpráv a zobrazení stavu*" a dříve známý pod názvem „Jabber“, je další z rodiny datových protokolů. Protokol je založený na XML (Extensible Markup Language) a je standardizován v několika RFC dokumentech, především v RFC 3920, RFC 3921 a RFC 6120. Architektura je založena na principu klient/server a síť pro komunikaci s využitím XMPP protokolu je rozdistribuována na mnoho serverů po celém světě, které spolu vzájemně komunikuji. Pro komunikaci mezi klientem a serverem je využit protokol TCP na portu 5222 a servery spolu komunikují skrze TCP na portu 5269 [60], Obr. 5.9.




Obr. 5.9 Model komunikace klient/server protokolu XMPP, převzato z [60].

Základním předpokladem komunikace je registrace klienta na jeden z mnoha světových serverů. Následně je mu umožněno přihlášení jako klienta pod uživatelským jménem (tvar: uživatel@server, také označováno jako JID (Jabber ID)) a heslem. Z toho plyne, že každý klient musí mít jednoznačné JID. Uživatel komunikuje vždy pouze se svým serverem a při potřebě mezi-serverové komunikace si komunikaci obstarává sám server [60].

### 5.4.1 QoS pro XMPP protokol

V tuto chvíli nejsou v žádném RFC dokumentu definované jakékoliv úrovně QoS na aplikační úrovni pro tento protokol, avšak v roce 2015 přišel autor Peter Waher s návrhem implementace tří úrovní QoS se stejnými vlastnostmi, jako u MQTT protokolu. Jsou to tyto úrovně [61]:

- Unacknowledged service – At most once (doručení zprávy nejvýše jednou)
- Acknowledged service – At least once (doručení zprávy alespoň jednou)
- Assured service – Exactly once (doručení zprávy právě jednou)

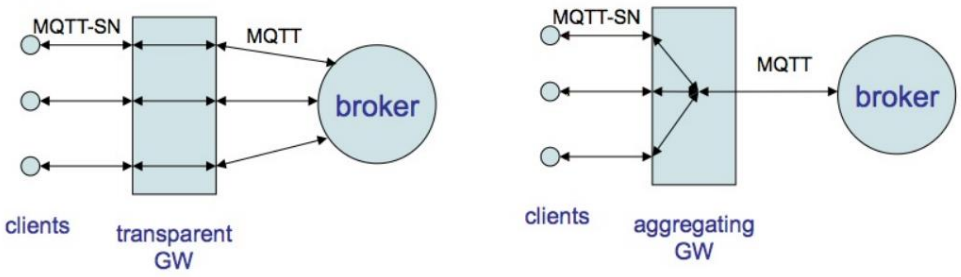
Ale toto řešení nebylo doposud přijato a schváleno oficiální cestou od nadace XMPP Standards Foundation, která se zabývá standardizací a správou rozšíření XMPP.

## 5.5 Protokol MQTT-SN

MQTT-SN, někdy také chybně označován jako MQTT-S, je zjednodušená verze MQTT protokolu určena pro komunikaci v senzorových sítích, kde je třeba maximálně zredukovat objem přenášených dat [62]. Jelikož neexistuje mnoho brokerů, kteří by

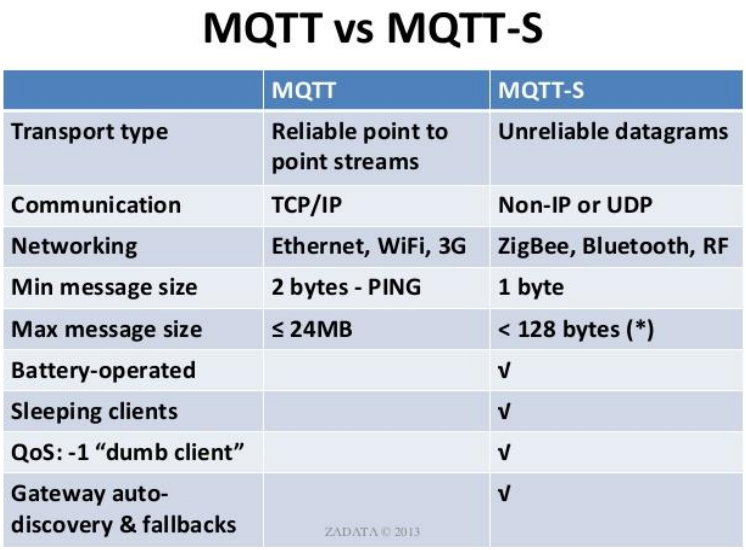
podporovali verzi MQTT-SN, lze mezi jednotlivé senzory a brokera zařadit tzv. gateway (dále jen GW). Tato GW zajistí převod protokolu MQTT-SN na MQTT protokol, a tím na brokera dorazí zpráva již v podobě, které rozumí běžný MQTT broker. Další možností je implementace GW přímo do brokera, tedy GW nemusí být stand-alone zařízení. Výhodou při užití převodu je, že přenos v podobě MQTT protokolu mezi GW a brokerem může být zabezpečen pomocí TLS (Transport Layer Security) [62]. Rozlišují se dva druhy GW, Obr. 5.10 [63]:

- Transparentní GW – GW udržuje přímé mapování mezi klienty (např. senzory) s MQTT-SN a MQTT připojením k brokerovi.
- Agregační GW – GW agreguje všechny data od senzorů do jednoho datového MQTT toku směrem k brokerovi.




Obr. 5.10 Dva druhy GW pro protokol MQTT-SN, převzato z [63].

Protokol MQTT-SN je navržen tak, aby měl stejné funkce jako protokol MQTT, avšak pro potřeby senzorových sítí se některé parametry protokolu liší a jsou v něm implementovány nové funkce. Mezi největší změny oproti klasickému MQTT patří využití přenosového transportního protokolu UDP namísto TCP, disponuje funkcí „Broker Discovery“, tedy funkcí pro objevení brokera a maximální velikost zprávy se snížila na pouhých 128 bajtů [62]. Detailnější porovnání protokolů MQTT a MQTT-S, resp. MQTT-SN lze vidět na Obr. 5.11.




Obr. 5.11 Porovnání MQTT a MQTT-S, resp. MQTT-SN, převzato z [64].

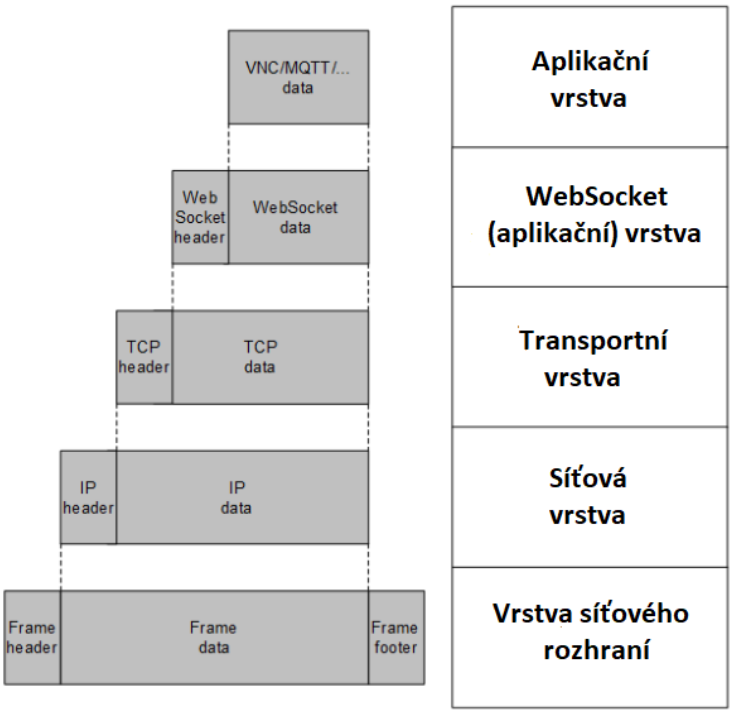
### 5.5.1 QoS pro MQTT-SN protokol

Odlehčený protokol MQTT-SN podporuje stejné úrovně QoS jako MQTT protokol, avšak umožnují navíc také čtvrtou úroveň označovanou jako QoS úrovně -1, resp. úrovně 3 (QoS flag = 3) [62]. Výhodou této úrovně QoS je, že publisher může své zprávy publikovat směrem k brokerovi bez předchozího navázání spojení, tedy bez zpráv CONNECT a CONNACK jako je tomu u běžného MQTT, avšak pro tuto úroveň QoS nelze zajistit ACK směrem zpět k senzoru [62].

Tento model publikování je ideální pro senzory napájené baterií, jelikož se zde ušetří část zpráv, které jsou pro běžné MQTT QoS úrovně potřebné, a to především na sestavení spojení, ACK potvrzeních a pravidelných dotazech mezi publisherem a brokerem na stav spojení v podobě Ping dotazů (request/response).

## 5.6 Protokol WebSocket

WebSocket je komunikační protokol na aplikační vrstvě TCP/IP modelu, Obr. 5.12. Pomocí WebSocketu lze navázat spojení pro oboustrannou komunikaci mezi klientem a serverem prostřednictvím jednoho TCP spojení, které je trvalé až do jeho ukončení jednou stranou [65]. Protokol je dnes standardně součástí většiny běžně užívaných prohlížečů jako je Google Chrome, Microsoft Edge, či Mozilla Firefox. Od klasického jednosměrného provozu jako je tomu u HTTP protokolu (dotaz/odpověď) se liší, mimo jiné, obousměrnou komunikací v reálném čase a nižším zatížením sítě.

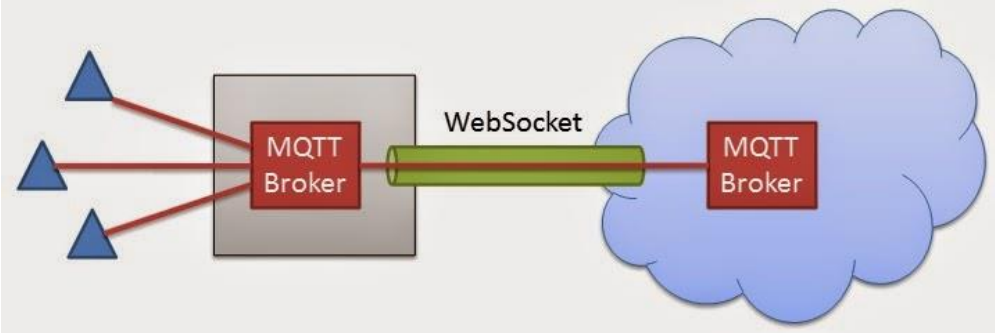



Obr. 5.12 Pozice WebSocketu v TCP/IP modelu, převzato a upraveno z [66].

Spojení klient-server začíná klient HTTP požadavkem, který obsahuje v hlavičce parametr HTTP upgrade [65]. Tato komunikace probíhá v případě HTTP na portu 80 a v případě HTTPS na portu 443. V hlavičce upgrade je definováno, že klient chce přejít na jiný protokol a port, v našem případě na WebSocket. Pokud server WebSocket podporuje, potvrzuje přechod na daný protokol a port v hlavičce zpětného paketu.

### 5.6.1 MQTT skrze WebSocket

Data mohou být přenášená přímo v těle WebSocketu, ale častěji se lze setkat s tím, že WebSocket slouží také jako tzv. „zábalová“ vrstva pro protokoly vyšší vrstvy, Obr. 5.13 [65]. Pole “Sec-WebSocket-Protokol”, které je obsaženo v těle HTTP paketu nese informaci směrem k serveru, že se bude skrze WebSocket přenášet podprotokol aplikační vrstvy, v našem případě: „Sec-WebSocket-Protocol: mqttvx.x“. Výhodou se tak stává, že skrze jedno TCP spojení spolu mohou obousměrně komunikovat klient a server. MQTT protokol pak není jediný protokol, který lze skrze WebSocket přenášet. Může se přenášet libovolný protokol, na kterém se klient a server shodnou [65].




Obr. 5.13 komunikace skrze WebSocket, převzato z [67].

### 5.6.2 Nevýhody WebSocketu v IoT

V některé literatuře se lze ještě s protokolem WebSocket jako IoT protokolem setkat, avšak již velmi okrajově. Důvodem je postupné upouštění od tohoto protokolu a jeho nahrazení novými, efektivnějšími protokoly v oblasti IoT.

Při pohledu na Obr. 5.12 je zřejmé, že zabalíme-li MQTT protokol do WebSocketu, zvětší se velikost celkové zprávy o hlavičku WebSocketu, což je pro zařízení, která vyžadují kvůli životnosti baterie a kapacitě přenosového média minimalizaci přenesených dat, nadbytečné. Další nevýhodou je, že musí server a všichni klienti podporovat jak daný podprotokol, tak i samotný WebSocket.

Obecně lze říct, že v tuto chvíli komunikuje převážná většina IoT periférií jednosměrně a komunikace opačným směrem tvoří procentuální zastoupení v jednotkách procent. Při takovém scénáři není nutné zajišťovat tunel pro obousměrnou komunikaci mezi komunikujícími zařízeními.

## 5.7 Tabulkové porovnání protokolů

O obecně uznávaných datových protokolech v IoT lze říct, že u většiny zmíněných je možné řídit v určité podobě kvalitu služeb na aplikační vrstvě, Tab. 5.2. Pouze u protokolu XMPP aktuálně není rozšíření o parametry QoS schváleno a implementováno, viz. kapitola 5.4.1.

Tab. 5.2 Protokoly s QoS.

| Protokol | Umožňuje řídit QoS? | Množství QoS úrovní |
|---|---|---|
| **MQTT** | Ano | 3 |
| **CoAP** | Ano | 2 |
| **XMPP** | Ne | - |
| **MQTT-SN** | Ano | 4 |

Obecné vlastnosti jednotlivých protokolů jsou shrnuty v Tab. 5.3.

Tab. 5.3 Shrnutí vlastností protokolů v IoT (* převzato z [68])

| Protokol | Transportní protokol | Messaging | Možné zabezpečení | Vhodné po 2G/3G/4G | Vhodnost pro LP síť | Adresování |
|---|---|---|---|---|---|---|
| **MQTT** | TCP/IP | Publish/Subscribe | TLS | Vynikající * | Přiměřená * | Topic |
| **CoAP** | UDP/IP | Request/Response | DTLS | Vynikající * | Přiměřená * | URL adresa |
| **XMPP** | TCP/IP | Request/Response | TLS | Vynikající * | Vynikající * | JID |
| **MQTT-SN** | Non-IP, nebo UDP/IP | Publish/Subscribe | TLS mezi GW a brokerem | Vynikající | Vynikající | Topic |

# 6 Testovací pracoviště

Cílem praktické části této diplomové práce je navrhnout testbed – testovací pracoviště, pomocí kterého bude testován vliv IDS/IPS sondy SonIoT na provozní parametry přenosového kanálu a následně bude testován vliv sondy na jednotlivé služby provozované v datových sítích se zaměřením na oblast služeb IoT, především pak budou sledovány parametry, které souvisí s QoS a QoE.

Testovací pracoviště bylo navrženo jako statické v laboratorních prostorách Fakulty elektrotechnické ČVUT v Praze. Vzhledem k testování sondy v obou verzích, tedy Home a Industry, a v režimech IDS i IPS, byl testbed navržen tak, aby byla minimalizovaná potřeba rekonfigurace pracoviště, a to jak po HW, tak i SW stránce. V následujících částech praktické části je kladen důraz na to, aby bylo srozumitelně popsáno fyzické spojení komponent a jednotlivá síťová nastavení tak, aby byl s pomocí této „dokumentace“ testbed opětovně sestavitelný. Důraz byl kladen také na minimalizaci blackbox úseků měření pro jednoznačnou průkaznost naměřených výsledků.

V následujících kapitolách je popsán způsob činnosti jednotlivých použitých zařízení v testbedu a v případě testovacího zařízení FT i popis jednotlivých testů (slovně i graficky) s vysvětlením jednotlivých nastavitelných parametrů a možných výsledků z daných měření. Při návrhu jednotlivých testů byla navrhnuta taková posloupnost, aby na sebe jednotlivé testy navazovaly a výsledky testu předchozího připravily podklady pro ty následující. Výsledkem testů jsou číselné hodnoty výsledků, slovní zhodnocení a grafické výstupy, které poukazují na maximální limity sondy a případné nedostatky, které by mohly vést k omezení či úplně nefunkčnosti sondy při nasazení v domácím, či firemním prostředí. Výsledky testů poukazují také na parametry QoS, na které má sonda vliv, a které by mohly omezovat služby nejen přístupu k internetu, ale především pak jednotlivé provozované služby/aplikace v síťové infrastruktuře domácnosti/firmy s bližším zaměřením na IoT komunikaci.

Měření na testovacím pracovišti byla v rámci práce rozdělena do těchto kategorií:

- Zjištění maximálních limitů síťové propustnosti sondy v režimu IPS na transportní vrstvě TCP/IP modelu pomocí nástroje FlowTester.
- Identifikace maximálního datového toku, který sonda v režimu IDS zpracuje.
- Zjištění omezujících faktorů sondy – vytížení CPU, pracovní a maximální teploty CPU, využití RAM.

- Testování vlivu sondy na parametry QoS komunikačního kanálu (propustnost, ztrátovost paketů, jitter, RTT a další).
- Testování vlivu sondy na datové IoT protokoly a jejich parametry QoS.
- Objektivní zhodnocení QoE na základě zjištěných parametrů QoS.

## 6.1. Využitý hardware

Každý HW nástroj, který je v testbedu využit, je charakterizován přiloženým obrázkem (pokud je k dispozici) a tabulkou, která obsahuje informace o názvu produktu a jeho úloze v testbedu. Síťové propojení jednotlivých komponent bude projednáváno a názorně zobrazeno v kapitole 6.3. O sondě SonIoT bylo detailněji pojednáváno v kapitole 3.3, Obr. 6.1, Obr. 6.2, Tab. 6.1 a Tab. 6.2, a detailní činnost FlowTesteru bude vysvětlena v následujícím textu, Obr. 6.3, Tab. 6.3. Při výběru vhodných síťových prvků byla zvolena, jako rozhodující parametr, síťová propustnost. Minimální propustnost každého síťového prvku (kromě sondy, která je měřena) musí dosahovat minimálně propustnosti 1Gbit/s (Gigabit Ethernet), jelikož sonda ve verzi Industry dosahuje dle specifikace propustnost 300 Mbit/s a proto by 100 Mbit/s porty (Fast Ethernet) představovaly výrazné omezení rychlosti, Obr. 6.4, Obr. 6.5, Tab. 6.4 a Tab. 6.5. Tomu také odpovídá volba propojovacích kabelů CAT5e, které jsou určené pro rychlosti až do 1 Gbit/s, Tab. 6.6.

**SonIoT, verze Home**

Tab. 6.1 Údaje o sondě ve verzi Home.

| | |
|---|---|
| **Název produktu** | Sonda SonIoT verze Home, v Raspbian-SonIot-201119 |
| **Funkce v testbedu** | IDS a IPS |

Obr. 6.1 SonIoT verze Home, převzato z [19].

**SonIoT, verze Industry**

Tab. 6.2 Údaje o sondě ve verzi Industry.

| | |
|---|---|
| **Název produktu** | Sonda SonIoT verze Industry, ver. 2019-12-03-00-SonIoT |
| **Funkce v testbedu** | IDS a IPS |

Obr. 6.2 SonIoT verze Industry, převzato z [19].

### FlowTester

Obr. 6.3 FlowTester, převzato a upraveno z neveřejné dokumentace.

Tab. 6.3 Údaje o FlowTesteru.

| | |
|---|---|
| **Název produktu** | FlowTester |
| **Funkce v testbedu** | generátor TCP a UDP provozu do 1Gbit/s + měření a vyhodnocování |

### Switch

Obr. 6.4 Switch Aruba 2930F 8G, převzato z [69].

Tab. 6.4 Údaje o switchi.

| | |
|---|---|
| **Název produktu** | Aruba 2930F 8G |
| **Funkce v testbedu** | propojení všech komponent na úrovni MAC vrstvy + port mirroring pro IDS |

### Router

Obr. 6.5 Router Huawei B315s-22, převzato z [69].

Tab. 6.5 Údaje o routeru.

| | |
|---|---|
| **Název produktu** | Huawei B315s-22 |
| **Funkce v testbedu** | přidělení IPv4 z DHCP poolu + přístup na internet |

### Další užité komponenty

Tab. 6.6 Další užité komponenty v testbedu.

| **Název** | Virtuální FT server – lokální | Univerzitní FT server | UTP CAT5e kabeláž | PC |
|---|---|---|---|---|
| **Funkce** | virtualizovaný FT server vůči kterému je testováno | záložní a ověřovací FT server | propojení komponent s propustností 1Gbit/s | správa sondy a vyhodnocování dat |

## 6.2 Využitý software

SW Iperf3 je v této práci využit v rámci nástroje FlowTester, Obr. 6.6 a Tab. 6.7. Představuje nejdůležitější nástroj, pomocí kterého je měřena reálná propustnost sondy SonIoT na základě obsahu (payloadu) TCP, či UDP transportního protokolu. Stejně tak je součástí FlowTesteru i SW FlowPing, pomocí kterého lze měřit propustnost sondy s využitím protokolu UDP, Tab. 6.9. Také lze s jeho pomocí měřit ztrátovost paketů při zahlcování úseku sítě generováním narůstajícího síťového provozu.

Pro měření hodnot RTT byla uvažována běžná funkce Ping z OS Windows, avšak ta je dostačující pouze v případě, že se měří síťové úseky, kde jsou milisekundy jako dostačující parametr. V případě měření narůstajícího RTT vlivem sondy je však nutné měřit i na mikrosekundy, aby byl vliv sondy viditelný. Z tohoto důvodu je využit freeware (SW distribuován bezplatně) hrPing, který toto přesné měření umožňuje, Tab. 6.10.

Node-RED je open-source programovací nástroj, který je založen na platformě Node.js, Obr. 6.7 a Tab. 6.8. Samotné programování je založeno na tzv. flow-based přístupu. Jedná se grafické programování, kterému je dodávána logika na základě propojování jednotlivých nodů s předdefinovanou funkcí. Pro tento testbed byl Node-RED zvolen kvůli jeho jednoduchému a přehlednému ovládání, nenáročné instalaci na různé distribuce OS Linux a jeho schopnosti propojovat „věci“ a služby. Pro lepší pochopení lze uvést příklad, kdy senzor teploty připojení k platformě Rapsberry Pi s OS Raspbian a aplikací v Node-RED, předává svá data a ta jsou aplikací zaznamenána, vyhodnocena, prezentována a případně předána dál. např. při využití MQTT protokolu může aplikace představovat klienta, ale i např. brokera. Aplikaci pro testbed vytvořené v Node-RED bude věnována kapitola 6.2.1.

**Iperf3**


Obr. 6.6 Iperf3, převzato z [70].

Tab. 6.7 Údaje o Iperf3.

| | |
|---|---|
| **Název produktu** | Iperf3 |
| **Funkce v testu** | SW nástroj pro generování datového TCP/UDP toku a následného vyhodnocování – součástí HW FlowTester |

**Node-RED**

Tab. 6.8 Údaje o Node-RED.

| | |
|---|---|
| **Název produktu** | Node-RED |
| **Funkce v testu** | webová aplikace pro monitorování, grafické vyhodnocování a ukládání dat parametrů sondy |

Obr. 6.7 Node-RED, převzato z [71].

**FlowPing**

Tab. 6.9 Údaje o FlowPingu

| | |
|---|---|
| **Název produktu** | FlowPing |
| **Funkce v testu** | SW nástroj pro generování datového UDP toku – součástí HW FlowTester |

**hrPing**

Tab. 6.10 Údaje o hrPingu

| | |
|---|---|
| **Název produktu** | HrPing |
| **Funkce v testu** | SW nástroj pro přesné měření RTT s využitím funkce Ping (ICMP protokol) |

### 6.2.1 Aplikace pro měření parametrů sondy

V rámci snazšího monitorování parametrů sondy jako je vytížení CPU, využití paměti RAM a síťové propustnosti, ale i základního ovládání sondy a monitorování hrozeb bez využití SSH spojení byla navržena a naprogramována aplikace založená na open-source programovacím nástroji Node-RED. Aplikace je navržena tak, aby mohla být s minimem změn aplikovatelná na obě verze sondy. Node-RED je v nejaktuálnější podobě ve verzi:

- Verze Node-RED - 1.0.3
- Verze Node.js - 12.13.1

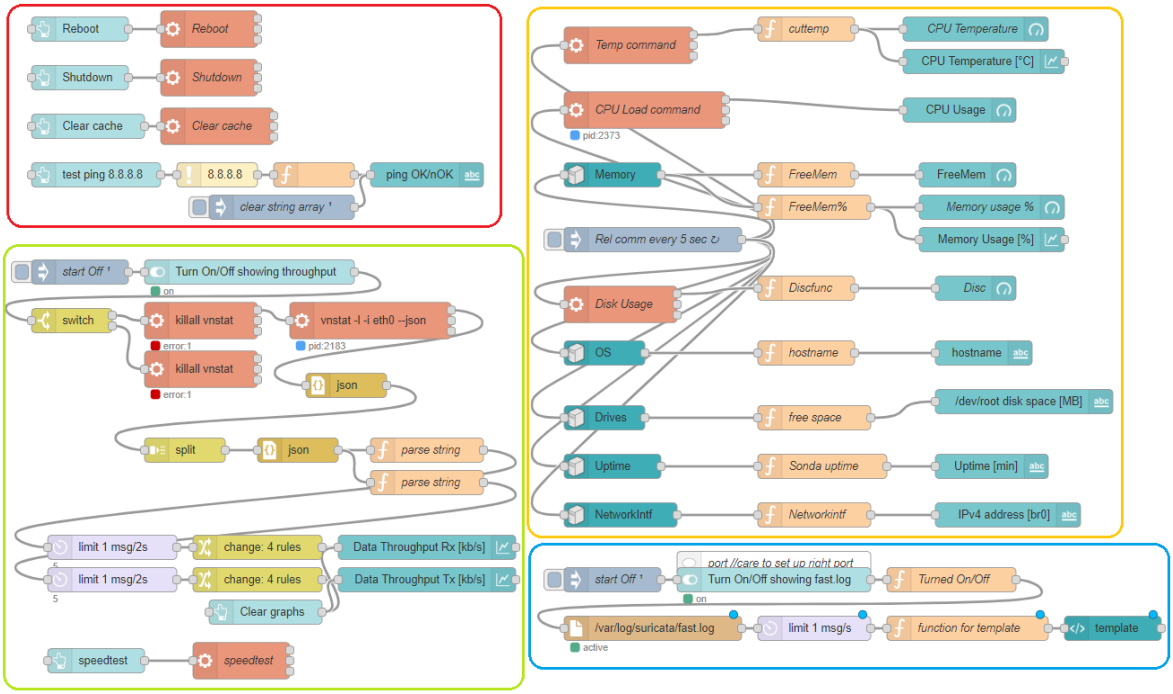
Při stávající konfiguraci monitorovací aplikace, Obr. 6.8, je využíváno přibližně 0–1 % výkonu procesoru u verze Home i Industry, což znamená, že aplikace nebude nikterak, či zcela minimálně omezovat činnost IDS/IPS sondy. Naprogramovanou aplikaci lze vidět na Obr. 6.8, 6.9 a 6.10. Mezi hlavní výhody aplikace patří:

- Měření a vyhodnocování hodnot z CPU a RAM v intervalu 5 sekund.
- Dlouhodobé statistiky hodnot z CPU a RAM zobrazené v grafech.

- Ukládání naměřených hodnot do textového souboru pro potřeby analýzy v jiných SW nástrojích.
- Průběžné informace o zaplněnosti úložiště sondy.
- Základní informace o sondě, jako je uptime, hostname a IPv4 adresa, na které je sonda dostupná.
- Kontrola přístupu sondy na internet pomocí funkce Ping na DNS server Googlu.
- Měření propustnosti Rx/Tx (přijaté/odeslané) sondy s ověřovacím speed testem.
- Real-time výpis logů zachycených hrozeb.

**Programovací prostředí**

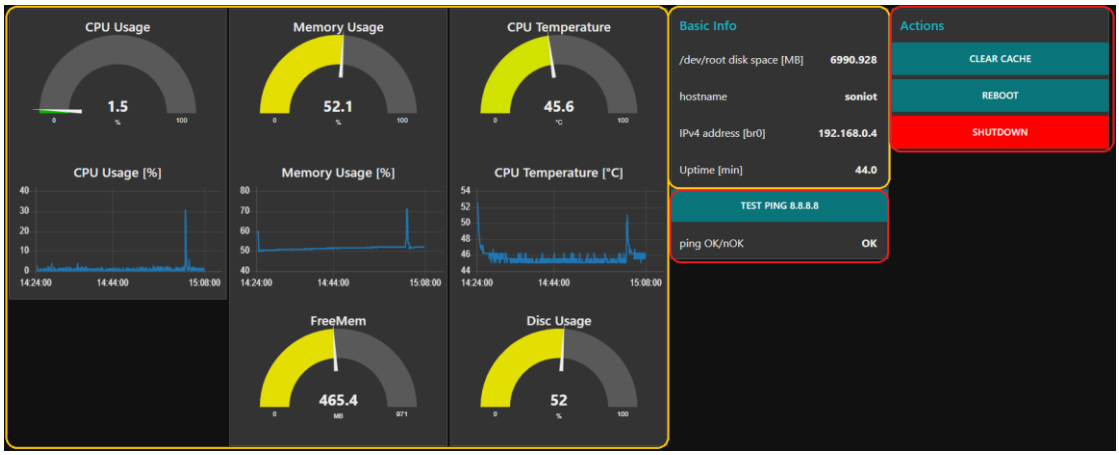
Aplikace je pomyslně rozdělena na 4 úseky, Obr. 6.8, které jsou rozděleny barevnými rámečky. V červeném rámečku jsou naprogramované funkce pro základní ovládání sondy – restart, vypnutí, uvolnění paměti RAM a ping na DNS server Googlu. Ve žlutém rámečku jsou naprogramované funkce, které monitorují parametry sondy, především pak ty nejdůležitější – teplota CPU, využití CPU a využití paměti RAM a ty jsou následně reprezentovány pomocí grafů. V zeleném rámečku je naprogramovaná funkce, která měří propustnost v každém směru (downstream a upstream) z požadovaného síťového rozhraní sondy a vynáší měřené hodnoty do grafů. V modrém rámečku lze nalézt funkci pro výpis zachycených hrozeb.




Obr. 6.8 Aplikace pro monitorování v programovacím prostředí Node-RED.

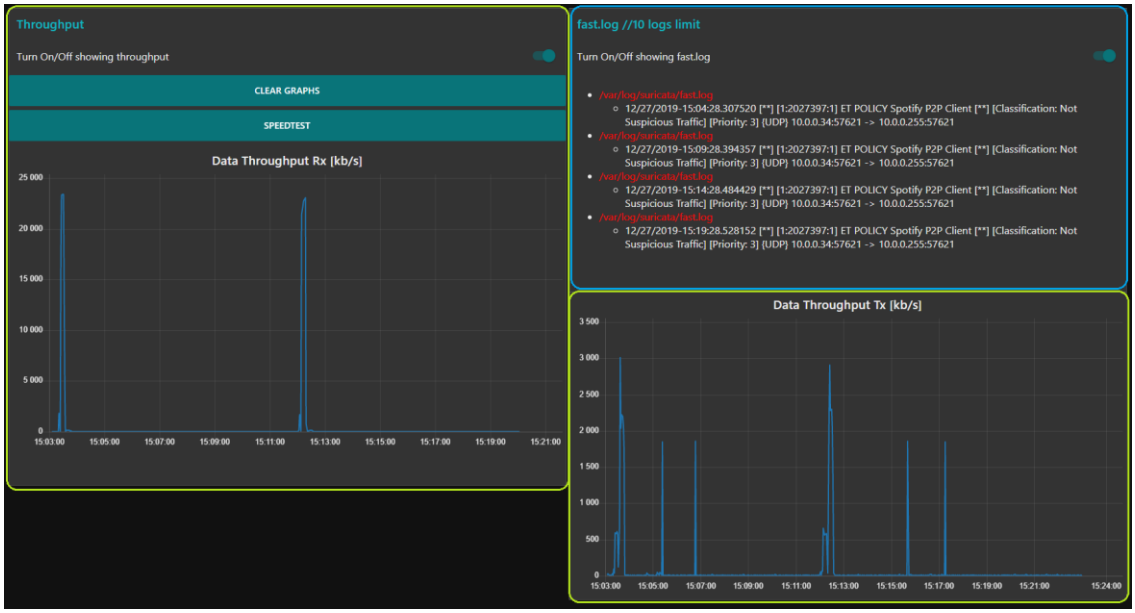
**Grafické prostředí pro správu a vyhodnocení dat**

První řada ukazatelů v levém žlutém rámečku ukazuje aktuální hodnotu využití měřených parametrů, která je aktualizována 1x za 5 vteřin. Jednotlivé testy jsou dlouhé 1,5 minuty a více, není proto třeba zatěžovat CPU častějšími odečty hodnot. Změny odečtených hodnot v čase lze sledovat v grafech v druhém řádku žlutého rámečku. Maximální zobrazovaný časový interval na ose x je nastaven na 1 hodinu tak, aby mohly být viditelné i vlivy na opakující se série testů. Ve třetím řádku žlutého rámečku lze sledovat přesné hodnoty volné paměti RAM. Procentuální využití paměti RAM je zobrazeno na prvním řádku. Také zde lze sledovat zaplněnost úložiště sondy, Obr. 6.9.




Obr. 6.9 Grafické prostředí aplikace pro správu a vyhodnocení dat – základní informace a monitorování CPU, RAM a uložiště.

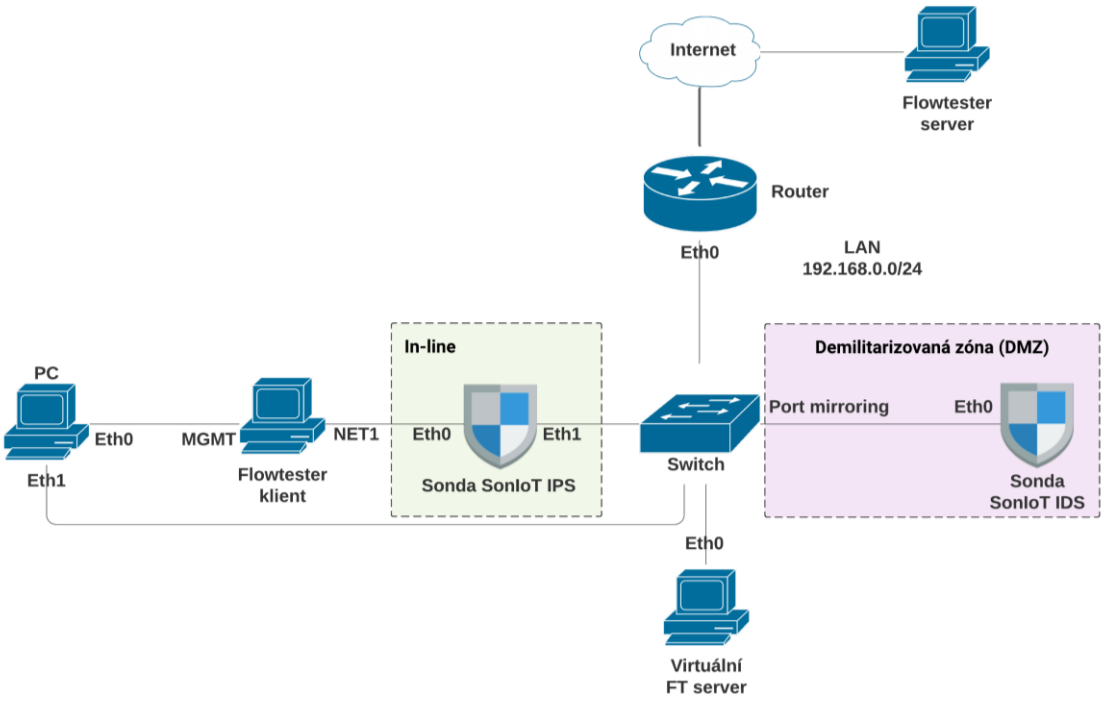
Ve dvou zelených rámečcích lze vidět v grafech naměřenou propustnost v obou směrech, Obr. 6.10. Pro ověření funkčnosti těchto grafů byla vytvořena jednoduchá funkce ověření rychlosti připojení (speedtest) vůči externímu serveru. V modrém rámečku se zobrazují aktuálně sondou zachycené hrozby, které jsou zobrazovány ihned při jejich zachycení. Pro přehlednost bylo okno v aplikaci omezeno na posledních 10 zachycených hrozeb.




Obr. 6.10 Grafické prostředí aplikace pro správu a vyhodnocení dat – propustnost a logy.

## 6.3 Schéma zapojení testbedu

Pro testbed bylo navrženo takové schéma zapojení, Obr. 6.11, které je platné jak pro verzi Home, tak i Industry. Testy spojené s vlivy sondy na provozní parametry přenosového kanálu jsou primárně vedeny mezi FT klientem a virtuálním FT serverem, aby byla zajištěna kompletní znalost celého úseku měřené trasy a vlastností jednotlivých komponent. V případě potřeby ověření, či porovnání naměřených výsledků lze následně využít testování vůči univerzitnímu FT serveru. V případě testování IoT služeb, které vyžadují komunikaci s externími klienty a servery přes internet, je router připojen do univerzitní sítě. Pod schématem na Obr. 6.11 se nachází tabulka, Tab. 6.11, která specifikuje využitá rozhraní zařízení a přiřazené IPv4 adresy. Adresy, které jsou označeny žlutě, se mohou reálně lišit v závislosti na nastavení DHCP serveru routeru. Sondy jsou dostupné i bez znalosti IP adresy na soniotips.local, resp. soniotids.local. Vzhledem k rozdílnosti IP rozsahů pro správu FT klienta a sond bylo využito druhé síťové rozhraní PC k propojení přímo do switche.




Obr. 6.11 Schéma zapojení s přístupem na internet.

Tab. 6.11 Rozhraní a IP adresy pro zapojení s přístupem na internet.

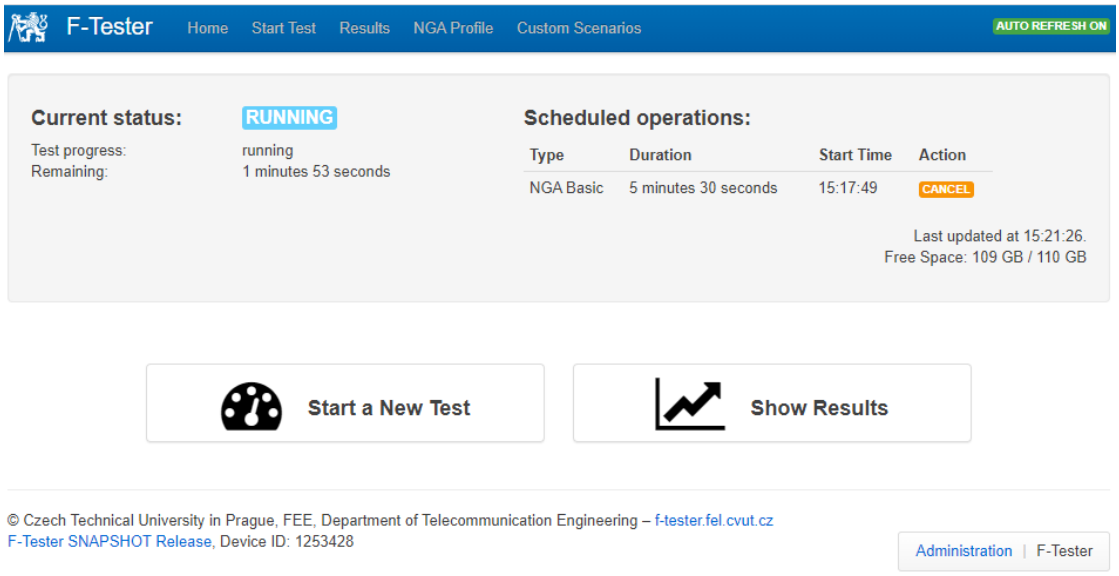
| Zařízení | Rozhraní | IP adresa | Síťová maska | Brána |
|---|---|---|---|---|
| PC | Eth0 | 172.16.1.2 | 255.255.255.0 | 172.16.1.0 |
| | Eth1 | 192.168.0.2 | 255.255.255.0 | 192.168.0.1 |
| Flowtester klient | MGMT | 172.16.1.1 | 255.255.255.0 | N/A |
| | NET1 | 192.168.0.3 | 255.255.255.0 | 192.168.0.1 |
| Sonda IPS | Eth0 | z DHCP – bridge mezi Eth0 a Eth1 dostupné na soniotips.local | | |
| | Eth1 | | | |
| Router | Eth0 | 192.168.0.1 | 255.255.255.0 | - |
| Flowtester server | N/A | 147.32.211.37 | N/A | N/A |
| Sonda IDS | Eth0 | Z DHCP – dostupné na soniotids.local | | |

## 6.4 Testovací rozhraní FlowTesteru

V této kapitole je uveden popis ovládání FT z grafického prostředí a možnosti nastavení jednotlivých testů, které jsou stěžejní pro měření vlivu sondy na parametry přenosového kanálu. Pro každý test jsou vysvětleny významy parametrů, které lze nastavit. Také jsou vypsány možné výstupy jednotlivých testů. Pro zachování přehlednosti jsou uvedeny pouze relevantní části k měření. Veškerá měření pomocí FT probíhají na 4. vrstvě referenčního modelu ISO/OSI a taktéž jsou na této vrstvě vykresleny průběhy propustnosti a RTT.

### Úvodní obrazovka

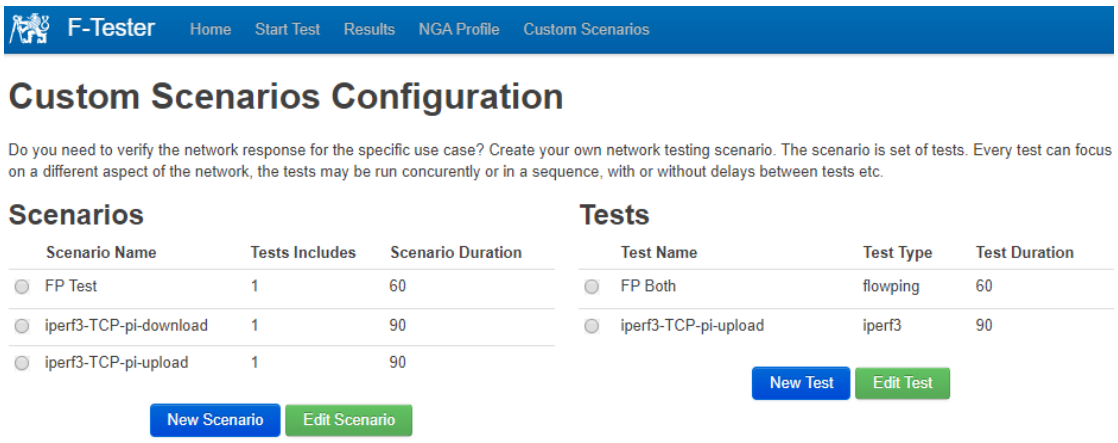
Úvodní stránka má informační charakter o stavu FT a probíhajících testů. Také představuje prvotní rozcestník, Obr. 6.12.




Obr. 6.12 FT úvodní stránka.

### Custom Scenarios (Vlastní scénáře)

V nastavení vlastních scénářů lze v sekci „Tests“ vytvářet testy se zvolenými parametry. V sekci „Scenarios“ pak lze vytvářet scénáře testování, tzn. že lze pouštet jednotlivé testy, nebo kombinace více testů, Obr. 6.13.

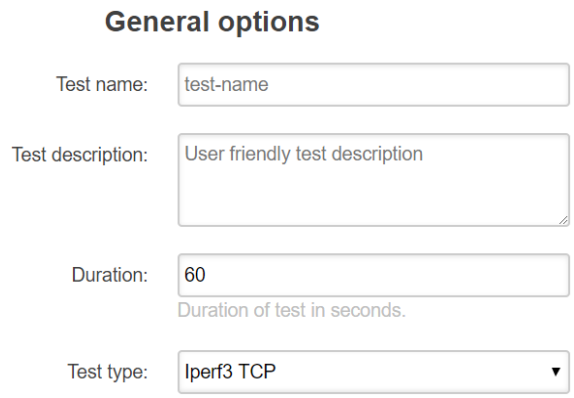



Obr. 6.13 FT vlastní scénáře.

### Custom Scenarios – Tests – New Test (Vlastní scénáře – testy – nový test)

V sekci vytváření nového testu je pro vytvoření testu jedna společná část a to „General options“, Obr. 6.14, kde lze definovat jméno testu (Test name), popis testu

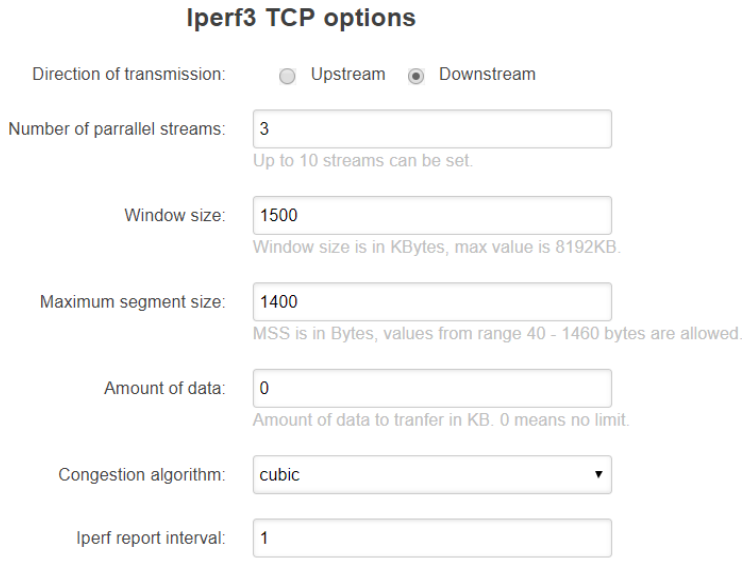
(Test description), dobu trvání testu v sekundách (Duration) a typ testu (Test type), který umožňuje 3 typy testů, jejichž rozdíly byly popsány v kapitole 4.2.1. Jsou to Iperf3 TCP, Iperf3 UDP a FlowPing.




Obr. 6.14 New Test - General options.

Následně se po vybrání testu objeví okno s nastavitelnými parametry, Obr. 6.15. Pro naše měření jsou relevantní především tyto parametry:

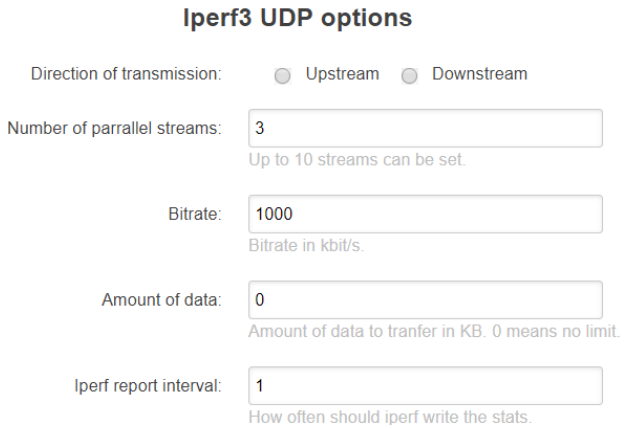
- Směr toku dat (Direction of Transmission) – upstream/downstream
- Počet paralelních streamů (Num. of parallel streams)
- Velikost TCP okna (Window size)
- MSS (Maximum segment size)
- Algoritmus kontroly zahlcení (Congestion algorithm) – cubic/reno/bbr



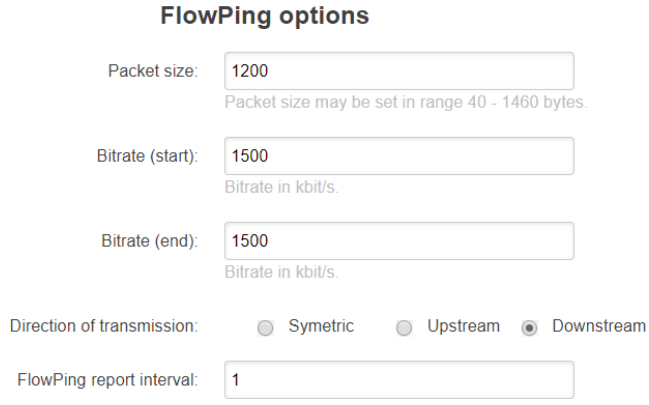

Obr. 6.15 New Test - Iperf3 TCP options.

Typy testů Iperf3 UDP a FlowPing lze popsat společně, jelikož oba pracují s protokolem UDP, Obr. 16 a Obr. 17. Mezi relevantní parametry pro naše měření patří:

- Směr toku dat (Direction of Transmission) – upstream/downstream a v případě FP i obousměrně (symetric)
- Počet paralelních streamů (Num. of parallel streams)
- Velikost paketu (Packet size)
- Vyžadovaná přenosová rychlost (bitrate), v případě FP i počáteční a koncová hodnota



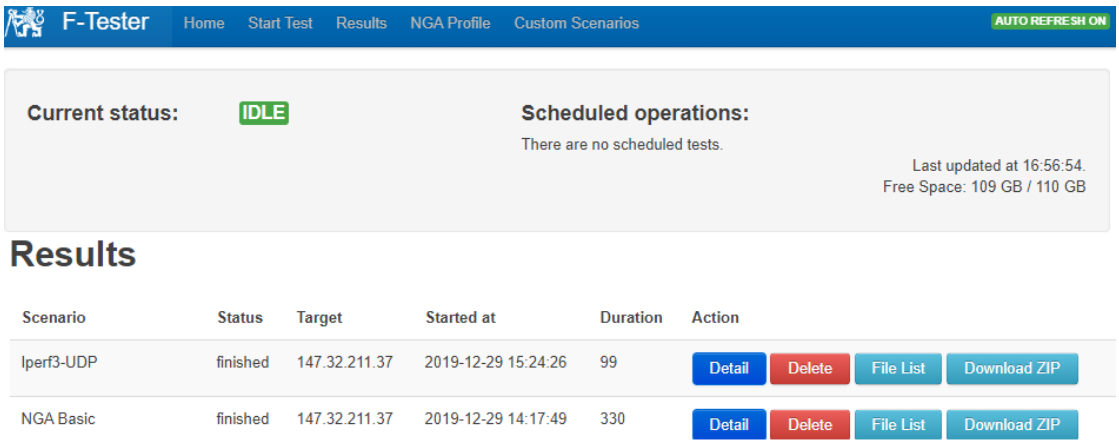

Obr. 6.16 New Test – Iperf3 UDP options.




Obr.6.17 New Test – FlowPing options.

**Results (Výsledky)**

V sekci výsledků lze nalézt jednotlivá měření, Obr. 6.18, a po rozkliknutí tlačítka „Detail“ se zobrazí požadované výsledky měření.



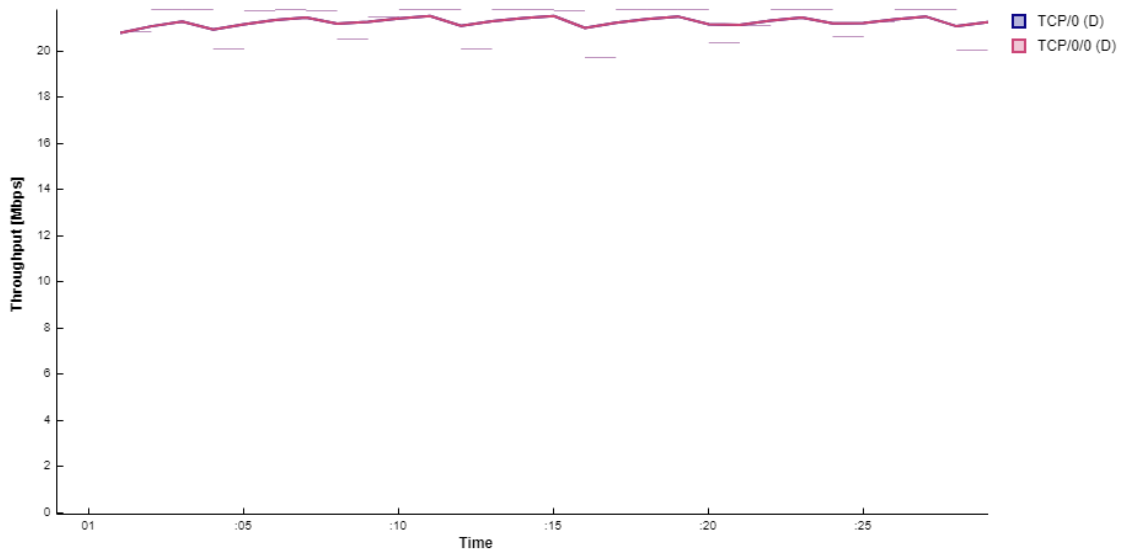

Obr. 6.18 FT výsledky měření.

### Výstupy jednotlivých testů

Pro zachování přehlednosti jsou jednotlivé možné výstupy z FT pouze tabulkově shrnuty, Tab. 6.12. Všechny výstupy jsou ve FT vyneseny do grafů. Ukázku grafu lze vidět na Obr. 6.19, kde je zobrazen časový průběh měřené propustosti přenosového kanálu ve směru downstream. Pro další zpracování lze naměřená data stáhnout jako soubor ZIP.

Tab. 6.12 Výstupy z jednotlivých FT testů.

| Iperf3 TCP | Iperf3 UDP | FlowPing |
|---|---|---|
| propustnost ve zvoleném směru [Mbit/s] | propustnost ve zvoleném směru [Mbit/s] | propustnost ve zvoleném směru [Mbit/s] |
| RTT [ms] | ztrátovost paketů [%] | RTT [ms] |
| CWND – škálováni TCP okénka [kB] | | ztrátovost paketů [%] |
| retransmise – znovu zaslané pakety v TCP přenosu [pcs] | | |




Obr. 6.19 Názorná ukázka FT výstupu – časový průběh měření propustnosti ve směru downstream s měřenými hodnotami mezi 21 – 22 Mbit/s.

# 7 Vliv sond v režimu IPS na provozní parametry přenosového kanálu

Tato kapitola je zaměřená především na prozkoumání vlivu sond ve verzích Home a Industry a v režimech IDS/IPS na parametry přenosového kanálu. Zajímavé jsou především ty parametry, které souvisí s kvalitou služeb. Vzhledem k tomu, že sonda je připojitelná do sítě skrze přenosovou technologii Ethernet, je celý přenosový kanál čistě fyzický a vedený po kroucené dvojlince s využitím UTP kabelů CAT5e. Aby bylo možné určit, jaké služby lze reálně provozovat na přípojce, kde je využita sonda SonIoT, je potřeba určit maximální limity parametrů kvality, tedy např. reálná propustnost sondy v obou směrech a vliv sondy na parametry jako RTT a ztrátovost paketů. Obecně lze říct, že je potřeba zjistit, jaké představuje sonda omezení při jejím nasazení do LAN sítě, a to především v režimu IPS. Z dokumentace víme, že verze Home a Industry mají určenou propustnost 100 Mbit/s a 300 Mbit/s v režimu IPS a v režimu IDS tato hodnota není známa. Vzhledem k různým principům vyhodnocování u systémů IDS a IPS můžeme předpokládat před samotným měřením, že spíše než propustnost sondy jako síťového elementu, nás u IDS systému bude zajímat, jaký objem dat je schopna sonda přijmout a zpracovat.

K samotnému testování je použito schéma zapojení z Obr. 6.11 a testy jsou spouštěny z grafického rozhraní FT.

**Měření testem Iperf3 TCP**

Před samotným měřením s využitím TCP protokolu je třeba identifikovat parametry, které k provedení testu vyžaduje FlowTester a jsou pro měření relevantní:

- Doba trvání testu
- Typ testu – Iperf3 TCP
- Směr toku dat – upstream/downstream
- Počet paralelních streamů
- Velikost TCP okna
- MSS

**Doba trvání testu** byla stanovena na 1,5 minuty ve směru downstream i upstream s přihlédnutím k požadovanému minimu 30 vteřin, dle ČTÚ, a předpokladu určitého časového intervalu k náběhu na požadovanou rychlost. Jako **typ testu** byl zvolen Iperf3 TCP, jelikož je to jediný typ testu v FT, který umožnuje měření pomocí

protokolu TCP. Vzhledem k požadavku ČTÚ na **měření ve směru downstream i upstream** byly vytvořeny 2 navazující testy, každý pro jeden směr. Bylo zvoleno 1, 3 a 6 **paralelních TCP streamů** (datových toků), tedy 3 měření pro každý směr, aby byla ověřena vyplněnost kapacity kanálu. **Velikost TCP okna** byla zvolena v závislosti na předchozích měřeních, které prokázali, že velikost okna 1500 kB je dostačující a vyšší hodnota již nevede ke zlepšení propustnosti. Pro ověření maximální možné hodnoty TCP MSS byla uvažována hodnota MTU 1500 B, což je typická hodnota pro Ethernet. Pomocí funkce Ping byla ověřena průchodnost ICMP payloadu o velikosti 1472 B bez nutnosti fragmentace (1472 B payload + 8 B ICMP hlavička + 20 B IPv4 hlavička = 1500 B velké MTU) a pro větší MTU již byla vyžadována fragmentace. Jako hodnota **TCP MSS** (TCP payload) byla tedy zvolena maximální možná hodnota 1460 B (1500 B MTU – 20 B TCP hlavička – 20 B IPv4 hlavička). Takovéto nastavení parametrů je platné pro všechny následující měření, pokud nebude stanoveno v textu jinak.

Následně byl proveden před samotným měřením ověřující test, zda je možné dosáhnout mezi FT klientem a FT virtuálním serverem přenosové rychlosti v jednom směru bez zapojené sondy požadované teoretické rychlosti 1 Gbit/s s využitím TCP protokolu, aby bylo možné předem vyloučit jiná omezení kapacity přenosového kanálu než ta vlivem sondy. Omezení by mohlo být např. v nedostatečném výkonu zařízení, na kterém běží virtuální FT server, poškozených kabelech, nebo nevhodným nastavením na switchi. Omezení také představuje fakt, že propustnost 1 Gbit/s je myšlena na fyzické vrstvě, proto na vrstvě transportní lze očekávat nižší rychlosti. Naměřené hodnoty lze vidět v Tab. 7.1

Tab. 7.1 Naměřené hodnoty propustnosti mezi FT klientem a FT serverem pro TCP.

| | Počet streamů | Velikost okna [kB] | MSS [B] | Min [Mbit/s] | Průměr [Mbit/s] | Max [Mbit/s] | Rozdíl min/max [Mbit/s] | Trvání testu [s] |
|---|---|---|---|---|---|---|---|---|
| **downstream** | 3 | 1500 | 1460 | 940,1 | 941,3 | 941.5 | 1,4 | 90 |
| **upstream** | 3 | 1500 | 1460 | 940,4 | 941,4 | 942,7 | 2,3 | 90 |

Hodnota okolo 940 Mbit/s (efektivita cca 94 %) je očekávaná hodnota propustnosti při výchozí velikosti Ethernet rámce 1518 B, resp. MTU 1500 B bez použití jumbo rámců, které zvyšují efektivitu až na cca 99 % [72]. Nejvyšší hodnoty byly naměřeny při třech datových tocích (streamech).

**Měření testem Iperf3 UDP a FlowPing**

Stejně jako v případě měření testem Iperf 3 TCP i zde je před samotným testováním sondy potřeba identifikovat relevantní parametry pro měření pomocí FlowTesteru:

- Doba trvání testu
- Typ testu – Iperf3 UDP/FlowPing
- Směr toku dat – upstream/downstream/symetric
- Počet paralelních streamů (Iperf3 UDP)
- Velikost paketu (FlowPing)
- Vyžadovaná přenosová rychlost – v případě FlowPingu i počáteční a koncová hodnota

**Doba trvání testu** byla pro oba typy testů stanovena na 1,5 minuty s přihlédnutím k potřebnému času pro náběh na požadovanou rychlost, především pak u FP testu, který generuje provoz rostoucí v čase. Jako **typ testu** byly zvoleny oba zmiňované, tedy Iperf3 UDP a FlowPing, jelikož s pomocí Iperf3 UDP lze měřit maximální propustnost UDP datového toku a s pomocí FP lze měřit ztrátovost paketů s rostoucím provozem. Propustnost bude měřena pro každý **směr toku dat** a v případě FP i obousměrně. Pro lepší vyplnění kapacity kanálu při měření maximální propustnosti pomocí Iperf3 UDP byl zvolen **počet paralelních streamů** 3. **Velikost paketu** v případě měření pomocí FP byla zvolena maximální možná hodnota, tedy 1460 B. Tato hodnota je stejně jako v případě měření pomocí TCP omezena velikostní MTU = 1500 B. Na rozdíl od měření TCP si při měření pomocí UDP lze vyžádat množství odesílaných/přijímaných dat a generujícímu zařízení je lhostejné, jestli dojde k zahlcení sítě. Proto bude pro zjištění maximální propustnosti generováno pomocí Iperf3 UDP **maximální množství dat** bez ohledu na ztrátovost a pomocí FP bude generován **narůstající provoz**, při kterém bude sledována ztrátovost paketů.

Před samotným měřením byl proveden ověřující test, jakých hodnot můžeme na kanálu s propustností 1 Gbit/s dosáhnout. Naměřené hodnoty lze vidět v Tab. 7.2.

Tab. 7.2 Naměřené hodnoty propustnosti mezi FT klientem a FT serverem pro UDP.

| | Počet streamů | Iperf3 UDP (průměr) [Mbit/s] | FlowPing (Max.) [Mbit/s] | Trvání testu [s] |
|---|---|---|---|---|
| **Downstream** | 3 | 624 | 170 | 90 |
| **Upstream** | 3 | 602 | 176 | 90 |
| **symetric** | - | - | 153/140 | 90 |

Naměřené hodnoty pomocí UDP, Tab. 7.2, se značně liší od těch naměřených pomocí TCP, Tab. 7.1. V obou případech měření, tedy Iperf3 UDP i FlowPing, je maximální dosahovaná rychlost omezena výkonem HW, na kterém běží virtuální FT server. V případě testu Iperf3 je dosažena průměrná rychlost cca 600 Mbit/s v jednom směru, což je 2x více, než je avizovaná rychlost sondy ve verzi Industry, takže je tato rychlost pro měření zcela dostačující. V případě testu FlowPing je tato rychlost výrazně nižší, což je částečně způsobeno výkonem HW a také verzí FP, která má omezenou rychlost v jednom směru do 250 Mbit/s za ideálních podmínek (dostatečně výkonný HW), avšak tímto testem je ověřována ztrátovost paketů, a proto lze tuto hodnotu považovat za dostačující.

## 7.1 Home v režimu IPS

Měření kapacity přenosového kanálu pro měření sondy ve verzi Home v režimu IPS probíhalo v zapojení dle schématu dle Obr. 6.11. V takovémto zapojení představuje sonda omezení kapacity kanálu (bottleneck) mezi měřícím zařízením (FT klient) a měřícím serverem (virtuální FT server). V běžném pojetí IPS systému bychom se spíše setkali se zapojením sondy mezi router a switch, tak, aby byla chráněná celá vnitřní síť, avšak pro měření, které je zaměřené na testování propustnosti sondy lze využít zapojení dle tohoto schéma.

Jak již bylo zmíněno, úkolem IPS systému je kontrola a případná eliminace nežádoucího síťového provozu, což vyžaduje určitou výpočetní kapacitu zařízení, které poskytuje tuto kapacitu systému. Lze tak předem identifikovat, které faktory mohou ovlivnit výslednou propustnost přenosového kanálu. Mezi těmi hlavními faktory lze identifikovat tyto:

- Výkonnost CPU
- Paměť RAM
- Síťová propustnost zařízení

Proto je v této práci zkoumán, s pomocí aplikace vytvořené v Node-RED, vliv sondy nejen na přenosový kanál, ale také vliv datového toku na sondu.

Byla provedena série měření s využitím testu Iperf 3 TCP – downstream a upstream, Tab 7.3. Vzhledem k tomu, že bylo během měření zjištěno, že propustnost sondy je s narůstajícím časem testu různá, byla testovací doba zvýšena na 3 minuty. Tento jev může být spojen s přehlcením sondy, resp. s obrannými mechanismy proti

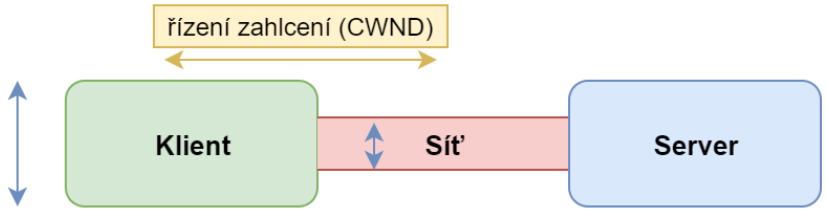
zahlcení sítě a procesy kontroly datového streamu sondy. Výsledky z měření lze vidět v Tab. 7.3.

Tab. 7.3 Naměřené hodnoty propustnosti komunikačního kanálu s propustností 1 Gbit/s (verze Home).

| Směr toku dat | Počet streamů | Velikost okna [kB] | MSS [B] | Min. [Mbit/s] | Průměr [Mbit/s] | Max. [Mbit/s] | Rozdíl min/max [Mbit/s] | Trvání testu [s] |
|---|---|---|---|---|---|---|---|---|
| **Down** | 1 | 1500 | 1460 | 75,7 | 82,1 | 93 | 17,3 | 180 |
| **Up** | 1 | 1500 | 1460 | 68,2 | 72,2 | 75,9 | 7,7 | 180 |
| **Down** | 3 | 1500 | 1460 | 110,3 | 131,2 | 145,2 | 34,9 | 180 |
| **Up** | 3 | 1500 | 1460 | 110 | 144,6 | 161,9 | 51,9 | 180 |
| **Down** | 6 | 1500 | 1460 | 77,9 | 114,4 | 150,1 | 72,2 | 180 |
| **Up** | 6 | 1500 | 1460 | 69,2 | 110,9 | 157,0 | 87,8 | 180 |

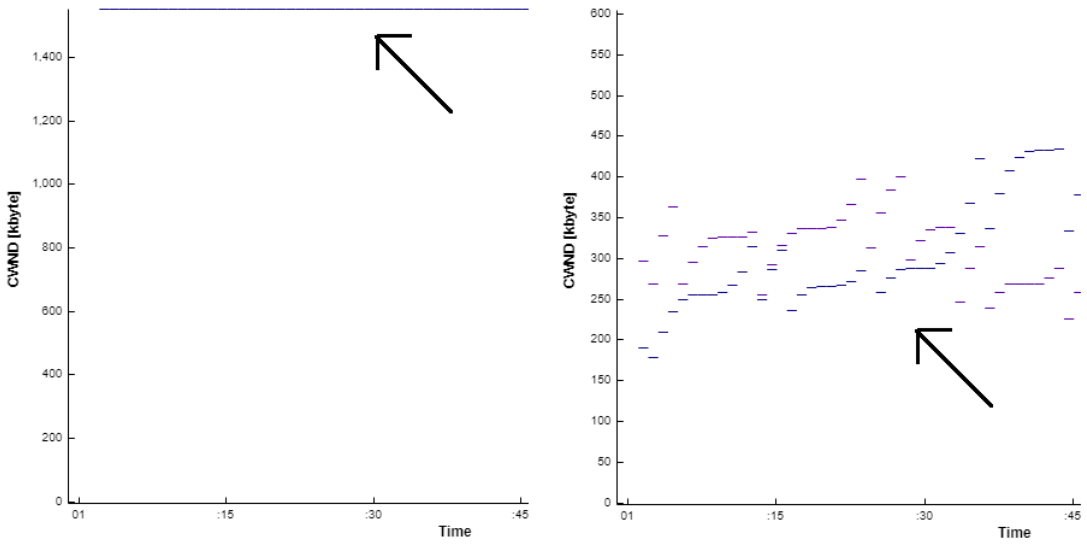
Z výsledků měření je patrné, že sonda ve verzi Home představuje pro 1 Gbit/s přípojku zpomalení na cca desetinu kapacity. To je samozřejmě očekávaný stav vzhledem k avizované rychlosti 100 Mbit/s pro tuto verzi, avšak zajímavý a nepřehlédnutelný je rozdíl minimálních a maximálních dosahovaných hodnot. Během testování byla naměřena maximální hodnota přenosové rychlosti na transportní vrstvě 161,9 Mbit/s pro 3 datové streamy. 6 datových streamů již nepřineslo lepší vyplnění kapacity kanálu, avšak pro 1 stream byla dosahovaná maximální rychlost 93 Mbit/s, a to i v případě opakovaných testů, což bude nejspíše způsobeno algoritmem proti přehlcení sítě.

Jednou z vlastností TCP protokolu je, že je umožněno řídit datový tok na vysílací (regulací vysílacího okna podle provozu v síti), či přijímací straně (inzerování velikosti přijímacího okna podle místa v přijímací vyrovnávací paměti) [73]. V případě, že by došlo ke skokovému zahlcení sítě, mohlo by dojít k zneprůchodnění všech již realizovaných spojení. Tato situace nastává např. v případech, kdy se posílají data z rychlejší sítě na pomalejší. Proti tomu bojuje mechanismus řízení proti zahlcení, který pracuje na principu úpravy velikosti vysílacího okna (CWND, Congestion Window) na vysílací straně a udržuje velikost CWND takovou, aby byla pod hranicí, nad kterou začne docházet k zahlcení [73], Obr. 7.1.




Obr. 7.1 Řízení zahlcení, vytvořeno na základe [73].

V našem případě představuje sonda omezující síťový prvek, který způsobuje proměnlivou kapacitu kanálu v závislosti na výkonu a vyrovnávací paměti sondy. Tím lze vysvětlit výrazné rozdíly v minimálních a maximálních dosahovaných hodnotách přenosové rychlosti, Tab. 7.3. Na Obr. 7.2 lze vidět rozdílné chování TCP okna – vlevo při měření úseku pouze mezi FT klientem a FT serverem a vpravo při měření stejného úseku se zařazenou sondou ve verzi Home.




Obr. 7.2 Ukázka činnosti proměnné hodnoty CWND. Bez zahlcení sítě (vlevo). Funkční mechanismus ochrany proti zahlcení sítě laděním velikosti CWND (vpravo).

V rámci porovnání byla provedena další série testů, kdy byla veškerá ethernetová rozhraní na switchi přepnuta do rychlostního režimu 10/100, tzn. že došlo k omezení maximální kapacity kanálu na 100 Mbit/s. Výsledky lze vidět v Tab. 7.4. Pokud porovnáme Tab. 7.3 a Tab. 7.4, tak lze vidět, že na komunikačním kanálu s propustností 100 Mbit/s se zvýšily minimální dosahované rychlosti a to především pro 1 a 6 datových streamů. To bude nejspíše způsobeno tím, že pro stabilní 100 Mbit/s propustnost kanálu po celé délce úseku není třeba časté skokové změny velikosti CWND vlivem obraného mechanismu proti přehlcení sítě, a proto i

přenosová rychlost je méně kolísavá. Průměrné přenosové rychlosti zhruba odpovídají očekávanému stavu pro 100 Mbit/s přípojku.

Tab. 7.4 Naměřené hodnoty propustnosti komunikačního kanálu s kapacitou 100 Mbit/s (verze Home).

| Směr toku dat | Počet streamů | Velikost okna [kB] | MSS [B] | Min. [Mbit/s] | Průměr [Mbit/s] | Max. [Mbit/s] | Rozdíl min/max [Mbit/s] | Trvání testu [s] |
|---|---|---|---|---|---|---|---|---|
| **Down** | 1 | 1500 | 1460 | 88,74 | 94,1 | 95,1 | 6,7 | 180 |
| **Up** | 1 | 1500 | 1460 | 79,4 | 81,8 | 83,37 | 4 | 180 |
| **Down** | 3 | 1500 | 1460 | 80 | 93,8 | 95,3 | 15,3 | 180 |
| **Up** | 3 | 1500 | 1460 | 90,1 | 94,1 | 95,2 | 5,1 | 180 |
| **Down** | 6 | 1500 | 1460 | 87,3 | 94,1 | 95,6 | 8,3 | 180 |
| **Up** | 6 | 1500 | 1460 | 85,2 | 94,3 | 95,2 | 10 | 180 |

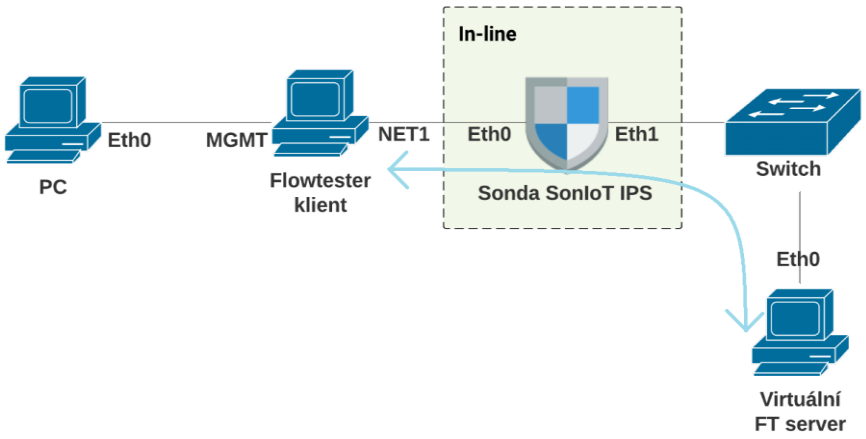
Dalším z měřitelných parametrů a ukazatelem kvality je RTT. RTT je proměnlivé v čase především v závislosti na faktorech jako jsou:

- Použité komponenty na dané trase
- Síťový provoz v LAN
- Vzdálenost mezi zdrojem a cílem
- Doba odezvy serveru a další

Výše zmíněné faktory jsou důvodem, proč jsou výsledky RTT, které poskytuje FT při měření maximální propustnosti, vhodné spíše pro měření úseků reálné sítě, např. z LAN k cíli mimo tuto LAN, kde nás zajímá výsledné RTT jako ukazatel kvality dané služby, jelikož RTT je jako parametr jeden hlavních parametrů ovlivňující uživatelský prožitek (QoE). V tomto měření je však důležité, jaký má vliv sonda jako síťový prvek na výslednou hodnotu RTT – jaký časový přírůstek přidá do celkového obousměrného zpoždění. RTT graf, který je generován při měření maximální propustnosti s FT, představuje takové hodnoty RTT, které odpovídají průběhu při dosahování limitních hodnot propustnosti. Tyto hodnoty jsou však výrazně vyšší než v případě, že není síť maximálně vytížená, proto nevypovídají o časovém přírůstku RTT vlivem sondy pro běžný provoz v síti. Z tohoto důvodu lze pro měření využít funkci Ping, kterou taktéž k měření RTT doporučuje ČTÚ [27].

Aby nebyla třeba změna v konfiguraci testbedu, byl jako zdroj Pingu využit PC, na kterém běží virtuální FT server a jako protistrana posloužil FT klient, Obr. 7.3. Jako

nástroj pro přesné měření byl využit freeware hrPing, který oproti běžnému Pingu měří RTT i na mikrosekundy, vypočítá min/avg/max RTT a v případě potřeby umožňuje vynést výsledky měření do grafu. Před zahájením měření bylo zajištěno, aby v síti neprobíhal žádný jiný datový tok a nedošlo tak ke zkreslení výsledků měření, např. prioritizací fronty na switchi.




Obr. 7.3 Schéma úseku měření pomocí funkce Ping.

Jelikož ČTÚ, ani jiná metoda nespecifikují přesný způsob měření pomocí funkce Ping, tak bylo měření provedeno následovně:

1. Měření bez zapojené sondy
2. Měření se zapojenou sondou

Naměřené výsledky lze vidět v Tab. 7.5. Z naměřených hodnot je patrné, že sonda ve verzi Home prodlužuje dobu RTT o více jak 1 milisekundu, což bude nejpravděpodobněji způsobeno procesem kontroly ICMP paketu při průchodu sondou tam (Echo request) a zpět (Echo reply). Také lze pozorovat rozdíl ve zpoždění u paketů s malou velikostí, tj. MTU = 60 B, a velikostí, která je rovna MTU = 1500 B, tedy maximální přenositelné jednotce technologie Ethernet. Zatímco u měření bez sondy lze označit průměrné dosahované hodnoty po zaokrouhlení za stejné, tak v případě měření se sondou dělal rozdíl ještě o cca 0,4 ms navíc.

Tab. 7.5 Naměřené hodnoty RTT se sondou a bez ní (verze Home).

| | IP MTU=60 B; ICMP payload = 32 B | | | IP MTU = 1500 B; ICMP payload = 1472 B | | |
|---|---|---|---|---|---|---|
| **Odesláno** | 100 paketů | | | 100 paketů | | |
| **RTT** | min [ms] | avg [ms] | max [ms] | min [ms] | avg [ms] | max [ms] |
| **Bez sondy** | 0,517 | 0,672 | 1,085 | 0,542 | 0,7 | 1,02 |
| **Se sondou** | 1,552 | 1,705 | 1,987 | 1,887 | 2,148 | 2,602 |
| **Rozdíl** | 1,035 | 1,033 | 0,902 | 1,345 | 1,448 | 1,582 |

Stejně tak dobře může být RTT naměřeno s pomocí nástroje FlowPing, pomocí kterého lze zároveň generovat dynamicky se zvyšující provoz, takže lze sledovat průběh změn hodnot RTT při narůstajícím provozu a zároveň s tím i spojenou ztrátovost paketů. Pro měření běžného RTT (jako v Tab 7.5) je však třeba generovat malý provoz, který se neblíží hranici propustnosti sondy.

Pro real-time služby využívající UDP protokol je krom parametru propustnosti sítě také velmi důležitý parametr ztrátovosti paketů, tedy rozdíl mezi množstvím odeslaných a přijatých paketů. Pro real-time služby je stěžejní, aby ztrátovost paketů byla co možná nejmenší, jelikož ztracené pakety již nejsou znovu odeslány. Obecně to znamená, že pokud budou UDP datagramy zasílány, nebo naopak přijímat skrze síť, která na to není dimenzovaná, dochází k zahazování paketů s těmito UDP datagramy, které se již nevejdou do vyrovnávací paměti omezujícího zařízení na trase, a tím i ke ztrátě např. kvality hovoru, nebo videostreamu, a tím i zhoršení uživatelského prožitku.

Pro běžně provozované real-time služby, Tab. 4.3, byly stanoveny mezní hranice do 1 % a 5 % ztrátovosti. V Tab. 7.6 lze vidět naměřené hodnoty s využitím testu FlowPing, při kterých byly tyto hranice překročeny. Také lze vidět maximální naměřenou rychlost v jednom směru s využitím testu Iperf3 UDP.

Tab. 7.6 Maximální hodnoty UDP provozu při ztrátovosti 1 % a 5 % (verze Home).

| Směr toku dat | Překročení limitu ztrátovosti paketů 1 % [Mbit/s] | Překročení lim. Ztrátovosti paketů 5 % [Mbit/s] | Max dosahovaná rychlost UDP [Mbit/s] | UDP payload [B] |
|---|---|---|---|---|
| **Down** | 65 | 80 | 170 Mbit/s | 1460 B |
| **Up** | 75 | 85 | 182 Mbit/s | 1460 B |
| **Down/Up (symetric)** | 50/50 | 48/66 | - | 1460 B |

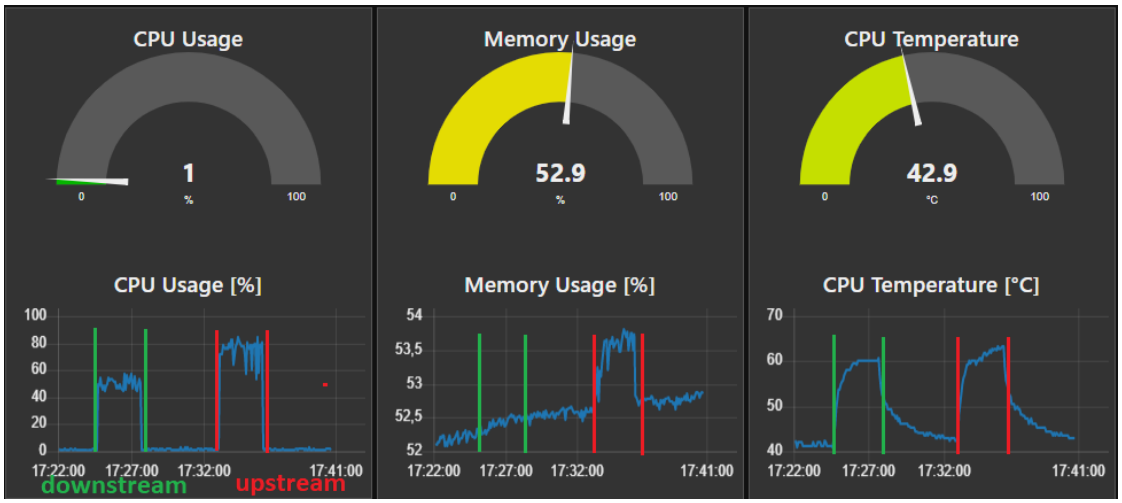
Z tab. 7.6 je patrné, že sonda dosahuje ztrátovosti paketů nad 5% už při cca při rychlosti 80 Mbit/s ve směru downstream a 85 Mbit/s ve směru upstream, což je při porovnání s Tab. 4.3 maximální hodnota ztrátovosti real-time služby streamování videa, avšak většina real-time služeb vyžaduje hodnotu ztrátovosti pod 1%. Tomu odpovídá přenosová rychlost menší než 65 Mbit/s ve směru downstream a menší než 75 Mbit/s ve směru upstream.

Z aplikace běžící na sondě lze během měření zjistit, že největší podíl na rychlosti zpracování a vyhodnocení dat a zároveň na celkovou propustnost sondy má výkon CPU, Obr. 7.4 a Tab. 7.7. Naopak paměť RAM není v rámci procesu IPS systému

téměř využita. Zajímavý jev lze pozorovat v případě dvou testů – Iperf3 TCP downstream a Iperf3 TCP upstream. I přesto, že oba testy dosahovaly přibližně stejných maximálních hodnot propustnosti, tak v případě upstreamu dosahoval využitý výkon CPU až 82 %, zatímco v případě downstreamu to bylo přibližně 60 %. Také paměť RAM byla v případě upstreamu využita o cca 1 % více než v případě downstreamu.

Tab. 7.7 Provozní parametry sondy (verze Home).

| Max. využití CPU [%] | | Využití RAM [%] | | Klidová teplota CPU [°C] | Max. teplota CPU [°C] | |
|---|---|---|---|---|---|---|
| **Downstream** | **Upstream** | **Downstream** | **Upstream** | | **Downstream** | **Upstream** |
| 59 | 82 | 0,2 | 1,2 | 41 | 60 | 64 |




Obr. 7.4 Graficky zpracované provozní parametry sondy (verze Home).

## 7.2 Industry v režimu IPS

Měření kapacity přenosového kanálu pro měření sondy ve verzi Industry v režimu IPS probíhalo v zapojení dle schématu dle Obr. 6.11. Měření probíhalo zcela identicky jako v případě verze Home, avšak pro zachování přehlednosti byly tato dvě měření rozdělena do dvou kapitol.

Byla provedena série měření s využitím testu Iperf 3 TCP – downstream a upstream, Tab. 7.8. Vzhledem ke kolísavosti přenosových rychlostí byla doba trvání testu nastavena na 3 minuty.

Tab.7.8 Naměřené hodnoty propustnosti komunikačního kanálu s propustností 1 Gbit/s (verze Industry).

| Směr toku dat | Počet streamů | Velikost okna [kB] | MSS [B] | Min. [Mbit/s] | průměr [Mbit/s] | Max. [Mbit/s] | Rozdíl min/max [Mbit/s] | Trvání testu [s] |
|---|---|---|---|---|---|---|---|---|
| **Down** | 1 | 1500 | 1460 | 158,7 | 303,5 | 414,2 | 255,5 | 180 |
| **Up** | 1 | 1500 | 1460 | 157,5 | 209,8 | 309,3 | 151,8 | 180 |
| **Down** | 3 | 1500 | 1460 | 200 | 323,4 | 426,6 | 226,6 | 180 |
| **Up** | 3 | 1500 | 1460 | 155,7 | 230,4 | 250,5 | 94,8 | 180 |
| **Down** | 6 | 1500 | 1460 | 190,7 | 345 | 426,7 | 236 | 180 |
| **Up** | 6 | 1500 | 1460 | 165 | 225,7 | 280 | 115 | 180 |

Při zprůměrování průměrných dosahovaných rychlosti je dosaženo rychlosti 324 Mbit/s ve směru downlink a 222 Mbit/s ve směru uplink. Tento rozdíl téměř 100 Mbit/s bude pravděpodobně způsoben nedostatečným výkonem jedné z měřících stran při ladění velikosti CWND a RWND. Vzhledem k výrazným rozdílům maximálních a minimálních dosahovaných hodnot je směrodatné uvažovat především průměrnou naměřenou rychlost, jelikož naměřené maximální hodnoty se během měření objevily pouze v podobě „špiček“, tzn. jednorázově ve chvílích, kdy síť byla nezahlcená. Minimální naměřené hodnoty se pak objevily pouze v počátku testu.

Druhá série testů, Tab. 7.9, byla provedena na komunikačním kanálu, který byl omezen na propustnost 100 Mbit/s. Z měření je patrné, že sonda dosahovala stabilně průměrné rychlosti 94 Mbit/s v obou směrech a pro různé počty datových streamů. Také rozdíly minimálních a maximálních hodnot jsou velmi nízké, což svědčí o stabilitě IPS systému na takto omezeném komunikačním kanálu.

Tab. 7.9 Naměřené hodnoty propustnosti komunikačního kanálu s kapacitou 100 Mbit/s (verze Industry).

| Směr toku dat | Počet streamů | Velikost okna [kB] | MSS [B] | Min. [Mbit/s] | Průměr [Mbit/s] | Max. [Mbit/s] | Rozdíl min/max [Mbit/s] | Trvání testu [s] |
|---|---|---|---|---|---|---|---|---|
| **Down** | 1 | 1500 | 1460 | 93,2 | 94,1 | 95 | 1,8 | 180 |
| **Up** | 1 | 1500 | 1460 | 91,4 | 93,8 | 95 | 3,6 | 180 |
| **Down** | 3 | 1500 | 1460 | 92,9 | 94,1 | 95,4 | 2,5 | 180 |
| **Up** | 3 | 1500 | 1460 | 93,1 | 94,2 | 95,2 | 2,1 | 180 |
| **Down** | 6 | 1500 | 1460 | 93,2 | 94,1 | 95 | 1,8 | 180 |
| **Up** | 6 | 1500 | 1460 | 87,6 | 94 | 95,1 | 7,5 | 180 |

Následně byla provedena série testů v zapojení dle Obr. 7.3, pro ověření časového přírůstku sondy k výsledné hodnotě RTT s pomocí hrPingu. Naměřené hodnoty lze vidět v Tab. 7.10.

7.10 Naměřené hodnoty RTT se sondou a bez ní (verze Industry).

| | IP MTU=60 B; ICMP payload = 32 B | | | IP MTU = 1500 B; ICMP payload = 1472 B | | |
|---|---|---|---|---|---|---|
| **Odesláno** | 100 paketů | | | 100 paketů | | |
| **RTT** | min [ms] | avg [ms] | max [ms] | min [ms] | avg [ms] | max [ms] |
| **Bez sondy** | 0,423 | 0,633 | 1,337 | 0,545 | 0,69 | 1,07 |
| **Se sondou** | 0,833 | 1,544 | 3,426 | 1,266 | 2,18 | 3,578 |
| **Rozdíl** | 0,41 | 0,911 | 2,089 | 0,721 | 1,49 | 2,508 |

Z naměřených hodnot je patrné, že sonda představuje časový přírůstek přibližně 1 ms, což je stejné jako v případě měření ve verzi Home, Tab. 7.5. To lze přisoudit kontrolním procesům sondy. Rozdíl měření se sondou při různých velikostech paketů, tedy MTU = 60 B a MTU = 1500 B, činil ještě navíc přibližně 0,6 milisekundy. Z toho plyne, že sonda potřebuje o 0,6 milisekundy více času na kontrolu jednoho paketu o maximální velikosti přenositelné na technologii Ethernet než v případě malého paketu.

V případě měření ztrátovosti paketů, Tab. 7.11, je nutné brát v potaz, že maximální propustnost v jednom směru pomocí testu FlowPing byla naměřena do hodnot 170 Mbit/s ve směru downstream a 176 Mbit/s ve směru upstream, Tab.7.2. Maximální dosahovaná rychlost byla měřena pomocí testu Iperf3 UDP.

7.11 Maximální hodnoty UDP provozu při ztrátovosti 1 % a 5 % (verze Industry).

| Směr toku dat | Překročení limitu ztrátovosti paketů 1 % [Mbit/s] | Překročení lim. Ztrátovosti paketů 5 % [Mbit/s] | Max dosahovaná rychlost UDP [Mbit/s] | UDP payload [B] |
|---|---|---|---|---|
| **Down** | 150 | 160 | 452 | 1460 |
| **Up** | 140 | 160 | 439,6 | 1460 |
| **Down/Up (symetric)** | 70/80 | 100/ 120 | x | 1460 |

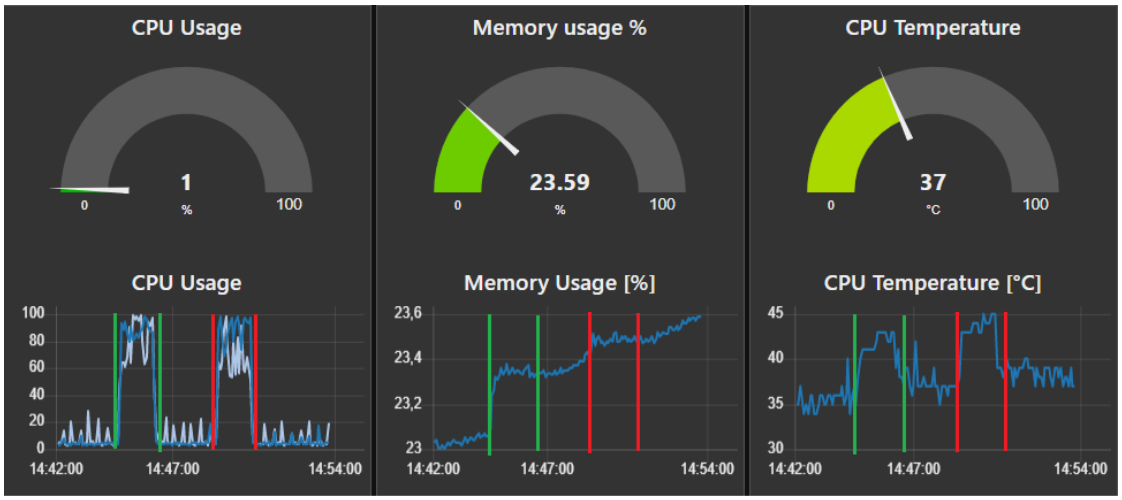
Dle naměřených hodnot, Tab. 7.11, je dosažen limit ztrátovosti paketů 1 % ve verzi Industry při rychlostech 150 Mbit/s ve směru downlink a 140 Mbit/s ve směru uplink. Při symetrickém měření to pak jsou hodnoty při rychlostech 70/80 Mbit/s. Při porovnání s maximálními naměřenými hodnotami to je přibližně třetinová rychlost

v jednom směru. Toto měření však může být zkreslené, jelikož je během testování dosahováno maximálních limitů nástroje FlowPing. Pro přesné měření by bylo třeba, aby FlowPing mohl dosahovat hodnot alespoň srovnatelných s maximální naměřenou hodnotou UDP provozu.

Pomocí naprogramované aplikace bylo možné během měření ověřit provozní parametry sondy, Tab. 7.12 a Obr. 7.5. Byly provedeny 2 testy – Iperf3 TCP ve směru downlink a ve směru uplink. Během testování bylo zjištěno, že je při maximálním TCP datovém toku je využití CPU přibližně 90 – 100 %. To znamená, že hlavním omezujícím faktorem sondy je výkonnost CPU a lze předpokládat, že v případě výkonnějšího CPU by sonda mohla dosahovat výrazně vyšší propustnosti. Těchto hodnot využití CPU bylo dosaženo jak v případě měření ve směru downlink, tak i uplink. Stejně jako v případě verze Home je i zde paměť RAM během měření využita zcela minimálně, a proto se lze domnívat, že nepředstavuje omezující faktor, který by měl vliv na činnost sondy.

7.12 Provozní parametry sondy (verze Industry).

| Max. využití CPU při testu [%] | | Využití RAM při testu [%] | | Klidová teplota CPU [°C] | Max. teplota CPU při testu [°C] | |
|---|---|---|---|---|---|---|
| **Downstream** | **Upstream** | **Downstream** | **Upstream** | | **Downstream** | **Upstream** |
| 100 | 100 | 0,1 | 0,1 | 35 | 44 | 45 |




Obr. 7.5 Graficky zpracované provozní parametry sondy (verze Industry).

## 7.3 Vliv sond v režimu IPS na provozní parametry přenosového kanálu

Z naměřených výsledků z Tab. 7.3 a Tab. 7.4. lze říct, že sonda ve verzi Home je pro použitelná pro přípojku do 100 Mbit/s a to jak ve směru downlink, tak i uplink a pro takovou přípojku nepředstavuje bottleneck. Jelikož v Tab. 7.3 byly naměřeny hodnoty vyšší, a to až 160 Mbit/s v jednom směru, lze tento stav označit za výhodu, jelikož reálně ISP většinou neposkytují přenosové rychlosti stejné pro směr downlink a uplink, ale většinou je downlink výrazně vyšší, např. 100/10 Mbit/s. I v takovém případě by sonda zvládala obousměrný provoz pro více datových toků.

Při měření verze Industry byly naměřeny průměrné dosahované hodnoty přenosové rychlosti 324 Mbit/s ve směru downlink a 222 Mbit/s ve směru uplink. Z těchto výsledků lze potvrdit, že sonda je schopna dosahovat avizované přenosové rychlosti 300 Mbit/s. Při úvaze obousměrného provozu lze říct, že sonda by mohla monitorovat síť, která je k poskytovateli připojena běžně nabízenou rychlostí 250/25 Mbit/s, což je např. rychlost nabízená společností O2 Czech Republic a.s. s označením tarifu „Internet HD Platinový“.

Vliv sondy na celkové obousměrné zpoždění, tedy RTT, byl v případě obou verzí sond stejný. Průměrně se tato hodnota pohybovala okolo 1ms, což nepředstavuje při běžném užívání non real-time služeb jako je „surfování“ na internetu a stahování online obsahu postřehnutelné zpoždění, a tedy i zhoršení uživatelského prožitku (QoE). Pro srovnání lze uvést např. přípojku přes technologii LTE, u které je běžná hodnota RTT 15 – 30 ms vůči externímu serveru.

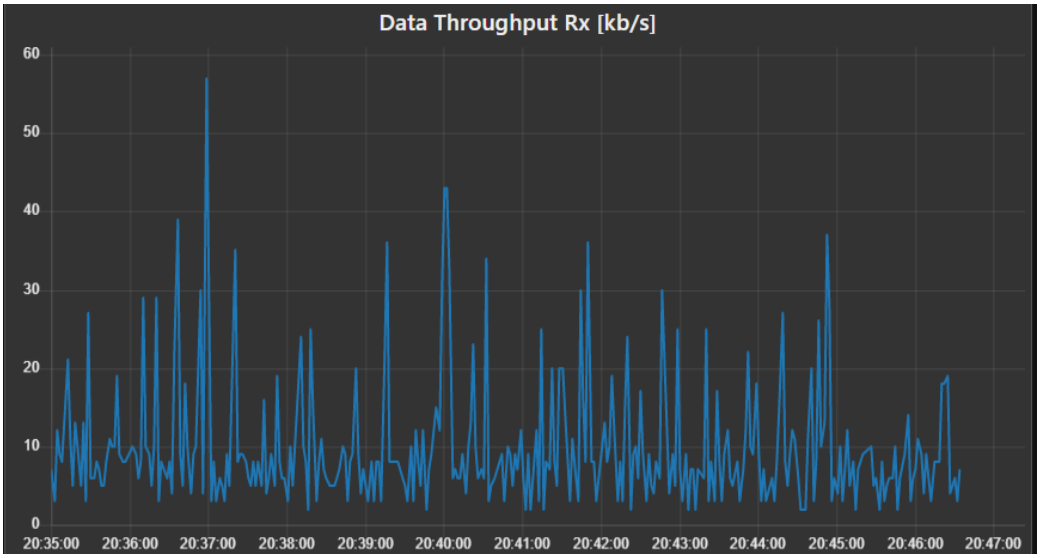
Z pohledu real-time služeb bylo během měření přihlíženo především ke ztrátovosti paketů. U verze Home byla hranice 1 % překročena při rychlosti 65 Mbit/s ve směru downstream a 75 Mbit/s ve směru upstream. U verze Industry byla překročena hranice 1 % při rychlostech 150 Mbit/s ve směru downlink a 140 Mbit/s ve směru uplink, avšak toto měření nelze, vzhledem k limitům měřícího nástroje, považovat za věrohodné. Z těchto výsledků lze říct, že pokud budou zařízení v dané síti generovat takový UDP provoz, který bude větší, než jsou limitní hodnoty ztrátovosti, bude to mít výrazný vliv na kvalitu prožitku z dané služby.

S pomocí aplikace v Node-RED bylo možné zjistit, že hlavním omezujícím faktorem propustnosti sondy je výkonnost CPU, které zpracovává veškeré požadavky na kontrolu dat, zatímco paměť RAM je během procesu kontroly téměř nevyužita.

## 7.4 Výkonnost zpracování dat sondy SonloT v režimu IDS

Vzhledem k principu činnosti IDS systémů a jejich umístění v demilitarizované zóně sonda nikterak neovlivní celkovou kapacitu kanálu, ať už v LAN, nebo směrem do WAN. Měření probíhá dle schématu Obr. 6.11, avšak s odpojenou sondou IPS, která by kapacitu kanálu značně zredukovala, a tím i zkreslila výsledná měření. Bez připojené IPS sondy je kapacita kanálu až k 1 Gbit/s. Na Switchi byl nastaven port mirroring tak, aby byl veškerý provoz z portu, který představuje propojení mezi FT klientem a FT virtuálním serverem, zrcadlen na port sondy.

Pro ověření množství přijatých dat sondou byla využita část aplikace v Node-RED, která vynáší do grafu množství dat, resp. množství dat za sekundu, přijaté na ethernetovém rozhraní. Díky tomu lze monitorovat, jaký objem dat sonda přijme. Vzhledem k tomu, že sonda v režimu IDS data hlavně přijímá, je stěžejní sledovat data přijímaná (Rx), Obr. 7.6.

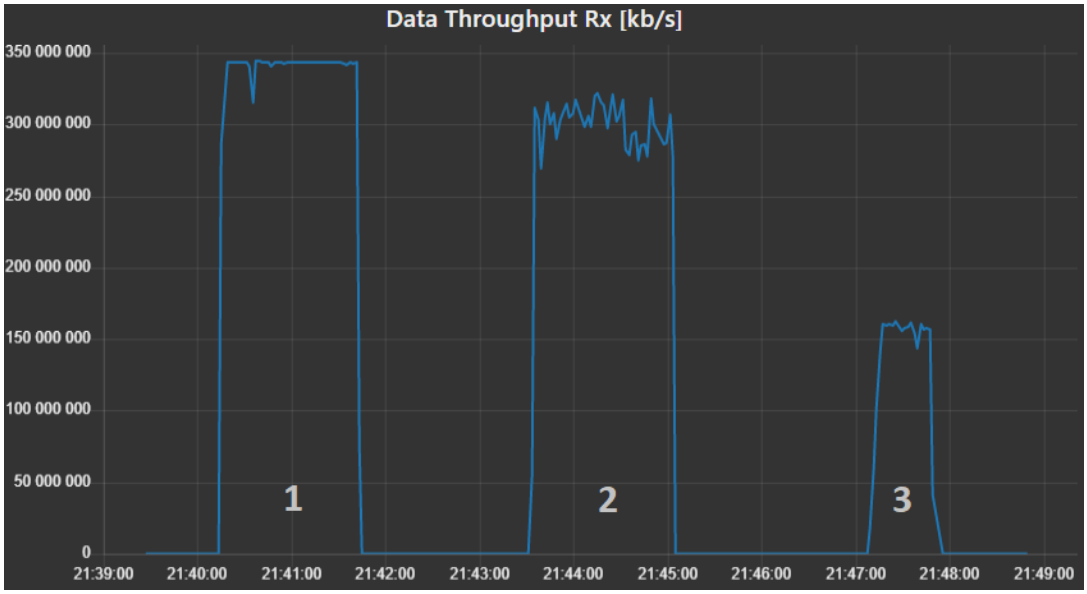



Obr. 7.6 Graf pro monitorování přijímaných dat sondou (Rx).

### 7.4.1 Verze Home v režimu IDS

V rámci měření, byly provedeny 3 testy, Obr. 7.7. Lze je rozlišit dle číselného označení v obrázku:

1. Iperf3 TCP downstream
2. Iperf3 UDP downstream
3. FlowPing downstream



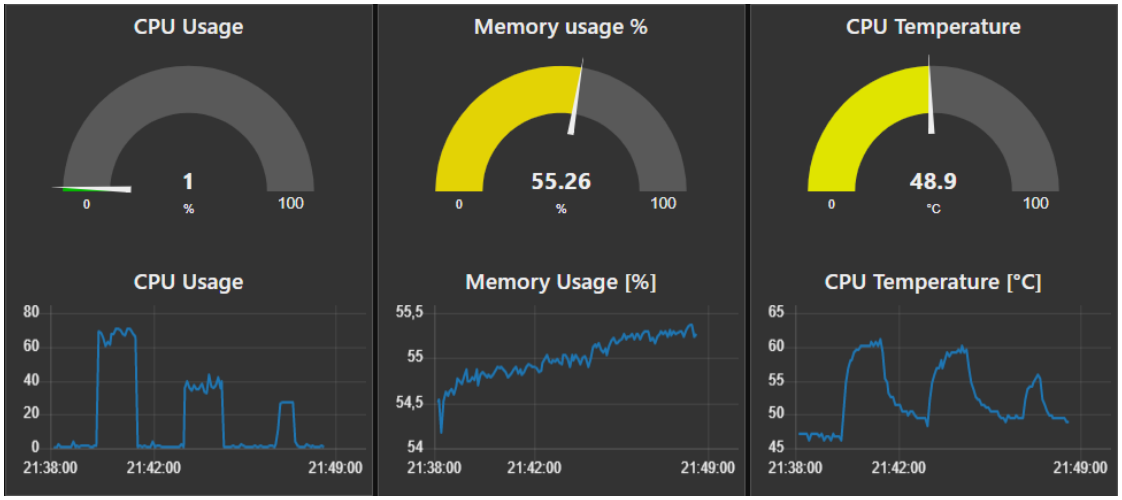

Obr. 7.7 Měření přijatých dat (Rx) sondy ve verzi Home.

Při porovnání hodnot z Tab. 7.13 je patrné, že nejvíce dat sonda přijala při měření pomocí TCP rychlostí 343 Mbit/s, což je o cca 600 Mbit/s méně, než bylo reálně generovaného provozu a zbytek paketů byl zahazován. Můžeme tak říct, že aby sonda zvládala zpracovávat veškerý provoz dané LAN, musel by být maximální provoz kontrolovaného úseku do přenosové rychlosti cca 340 Mbit/s. Jednotlivé špičky na Obr. 7.7, resp. kolísavost špiček přibližně odpovídají průběhům grafů z FT, i přesto, že Rx sondy je výrazně nižší. U aplikace FlowPing naměřená rychlost z FT odpovídá té z Node-RED aplikace, jelikož ve stávající verzi FP nelze vyšších hodnot dosahovat.

Tab. 7.13 Porovnání reálně generovaných dat s přijatými daty sondy ve verzi Home

| Typ testu | Směr toku dat | Rychlosti naměřené v FT [Mbit/s] | Rychlosti Rx sondy [Mbit/s] |
|---|---|---|---|
| **Iperf TCP** | downstream | 940 | 343 |
| **Iperf UDP** | downstream | 650 - 730 | 280 - 320 |
| **FlowPing** | downstream | 140 - 160 | 147 - 158 |

Z měřených parametrů sondy, Obr. 7.8, jsou pro jednotlivá měření z Tab. 7.13 vidět průběhy využití CPU, paměti RAM a změn teploty CPU. Nejvíce dosahovaly hodnoty využití a teploty CPU při prvním měření, kdy bylo také sondou přijato nejvíce dat. V prostředním grafu na Obr. 7.8 si lze všimnout, že paměť RAM není při procesu kontroly dat IDS systémem nikterak, nebo zcela minimálně, využita.

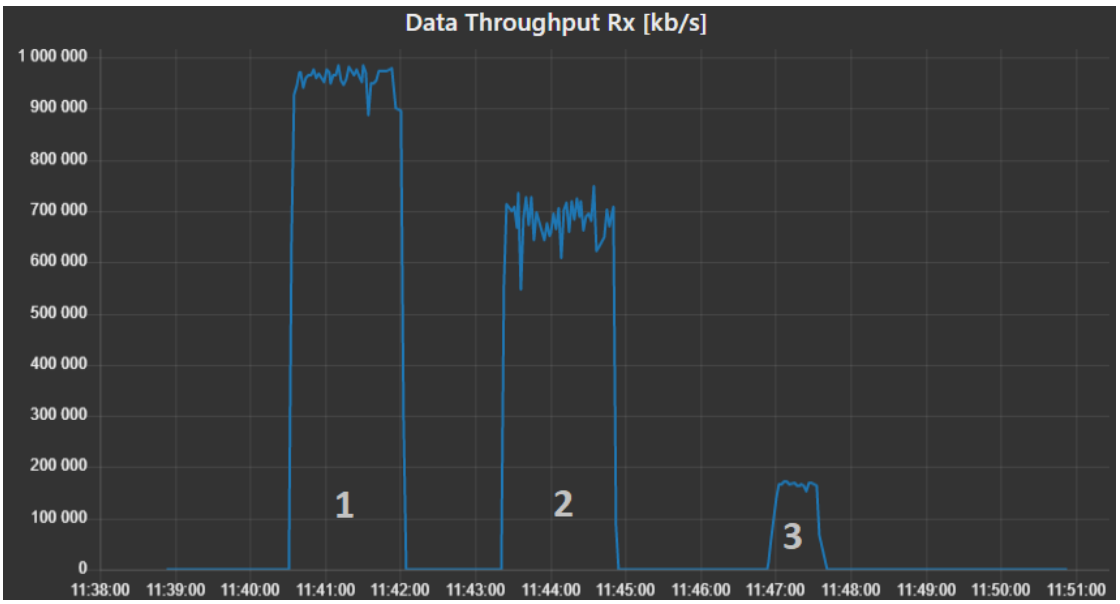



Obr. 7.8 Průběhy využití CPU, RAM a teploty CPU během tří měření u verze Home.

## 7.4.2 Verze Industry v režimu IDS

V rámci měření, byly provedeny 3 identické testy, jako v případě verze Home, Obr. 7.9. Lze je rozlišit dle číselného označení v obrázku:

1. Iperf3 TCP downstream
2. Iperf3 UDP downstream
3. FlowPing downstream




Obr. 7.9 Měření přijatých dat (Rx) sondy ve verzi Industry.

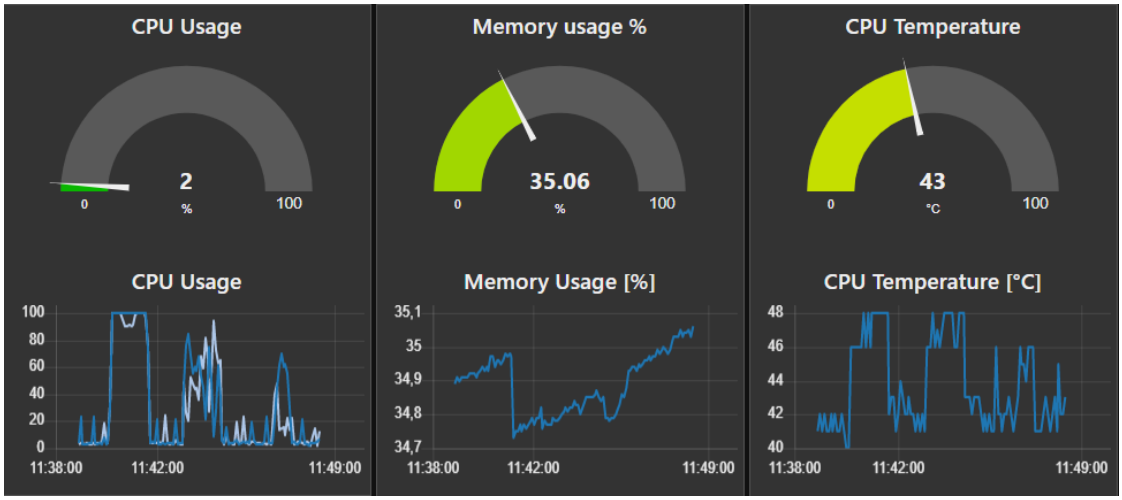
Při porovnání hodnot z Tab. 7.14 je patrné, že data zachycená sondou přibližně odpovídají hodnotám přenosu mezi FT klientem a FT virtuálním serverem. To znamená, že sonda je dostatečně výkonná, aby zpracovala veškerá data na 1 Gbit/s

přenosovém kanálu. V případě měření pomocí TCP protokolu dokonce hodnota Rx sondy přesahovala maximální naměřenou hodnotu v FT. To může být způsobeno chybou měření, nebo také tím, že při měření pomocí TCP se počítá s „payloadem“ TCP vrstvy, zatímco v případě měření v aplikaci Node-RED je počítán objem na nižší vrstvě, takže vzniká rozdíl především v nezapočítaných hlavičkách přenášené jednotky.

Tab. 7.14 Porovnání reálně generovaných dat s přijatými daty sondy ve verzi Industry.

| Typ testu | Směr toku dat | Rychlosti naměřené v FT [Mbit/s] | Rychlosti Rx sondy [Mbit/s] |
|---|---|---|---|
| **Iperf TCP** | downstream | 890 - 940 | 946 - 970 |
| **Iperf UDP** | downstream | 620 - 730 | 630 - 730 |
| **FlowPing** | downstream | 133 - 158 | 152 - 172 |

Na Obr. 7.10 lze vidět, že v případě prvního měření, dosahovalo využití CPU stabilně 100 % a to na obou jádrech CPU, což znamená, že větší Rx by sonda již pravděpodobně zahazovala. V případě druhého měření již bylo CPU využito přibližně na 50 – 60 % a v třetím měření to bylo průměrně 30%. Hodnoty využití RAM na Obr. 7.10, jsou poměrně neprůkazné, jelikož časové změny v RAM neodpovídají časovým průběhům měření, avšak lze říct, že během zpracovávání dat sondou je paměť RAM využívána zcela minimálně.




Obr. 7.10 Průběhy využití CPU, RAM a teploty CPU během tří měření u verze Industry.

### 7.4.3 Vliv sond v režimu IDS na provozní parametry přenosového kanálu

Z měření v kapitolách 7.4.1 a 7.4.2 bylo zjištěno, že sonda v režimu IDS nepředstavuje omezení kapacity kanálu, ani jiných provozních parametrů, jako je zpoždění, ztrátovost paketů apod., a tím pádem ani žádných služeb provozovaných v dané síti. V našem případě nekomunikuje sonda ani s jiným prvkem v síti, resp. nekomunikuje v rámci snahy IDS systému, proto nepředstavuje ani omezení v podobě impulzu na změnu na switchi, nebo routeru.

Pokud bychom porovnali verzi Home a Industry, tak můžeme říct, že verze Home umožňuje zpracovávat Rx do hodnoty TCP provozu až 340 Mbit/s a UDP až 320 Mbit/s, pak jsou pakety zahazovány a nedochází ke kompletní kontrole provozu. Tato hodnota je více jak trojnásobek inzerované propustnosti sondy v režimu IPS.

Verze Industry, dle výsledků měření, umožňuje přijmout veškerý provoz na kanálu s propustností 1 Gbit/s pro TCP provoz. V případě UDP provozu bylo měření omezeno výkonem měřícího HW. Byly naměřeny průměrné hodnoty okolo 660 Mbit/s se skokovými změnami až k 730 Mbit/s v jednom směru. V případě měření pomocí TCP bylo z grafu na obr. 7.10 vidět, že CPU sondy bylo vytíženo na 100 % a proto zde vzniká možnost, že při maximálním provozu na 1 Gbit/s kanálu by mohla sonda nezpracovat všechna data.

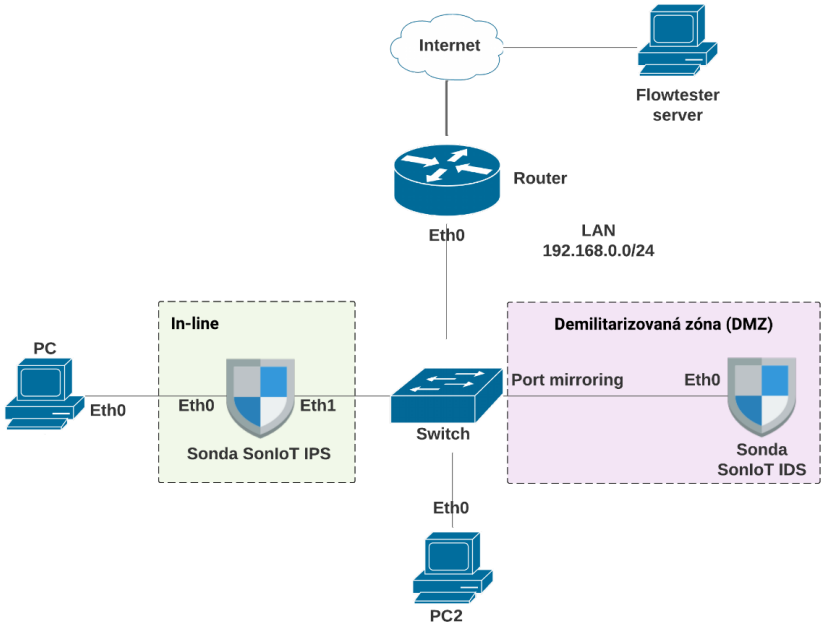
Z výsledků těchto měření, ale i obecných faktů o principu činnosti IDS lze říct, že sonda nepředstavuje žádné zhoršení, ale ani zlepšení kvalitativních parametrů služby (QoS) a ani uživatelského prožitku (QoE).

# 8 Vliv sondy na datové protokoly IoT

Tato kapitola je zaměřena na prozkoumání vlivu aplikace sondy v LAN síti na služby v oblasti IoT. Zaměřeno je především na provoz s využitím datových IoT protokolů MQTT, CoAP a jejich QoS parametry na aplikační vrstvě. Pro tyto účely byl testbed rozšířen o další funkci, a to o simulaci senzorů, které komunikují pomocí jednoho ze zmíněných protokolů, a tento provoz je veden skrze sondu v režimu IPS. Režim IPS byl zvolen, jelikož výsledky měření uvedené v kapitole 7 prokazují, že v režimu IDS sonda nikterak provoz neomezuje a nemá tak vliv na QoS a QoE parametry přenosového kanálu a na něm provozovaných služeb, avšak může nám posloužit pro ověření a porovnání zachycených hrozeb s IPS v případě potřeby. Také z kapitoly 7 je známa maximální propustnost sond v režimu IPS a jejich vliv na parametry přenosového kanálu při vysokých i nízkých přenosových rychlostech.

## 8.1. Schéma zapojení

Aby došlo jen k minimální rekonfiguraci testbedu, byl z něj odstraněn FT klient a PC, který FT klienta obsluhoval, se připojil přímo do IPS sondy. Také PC, na kterém běží virtuální FT server byl pro přehlednost přejmenován na PC2 a využit jako serverová strana pro virtuální senzory (broker pro MQTT). Pod Obr. 8.1 se nachází Tab. 8.1, která specifikuje využitá rozhraní zařízení a přiřazené IPv4 adresy. Adresy, které jsou označeny žlutě, se mohou reálně lišit v závislosti na nastavení DHCP serveru routeru.




Obr. 8.1 schéma zapojení pro měření vlivu sondy na datové protokoly IoT.

Tab. 8.1 Konfigurační tabulka pro měření vlivu sondy na datové protokoly IoT.

| Zařízení | Rozhraní | IP adresa | Síťová maska | Brána |
|---|---|---|---|---|
| PC | Eth0 | 192.168.0.2 | 255.255.255.0 | 192.168.0.1 |
| Sonda IPS | Eth0 | z DHCP – bridge mezi Eth0 a Eth1 dostupné na soniotips.local | | |
| | Eth1 | | | |
| Router | Eth0 | 192.168.0.1 | 255.255.255.0 | - |
| Sonda IDS | Eth0 | Z DHCP – dostupné na soniotids.local | | |
| PC2 | Eth0 | 192.168.0.3 | 255.255.255.0 | 192.168.0.1 |

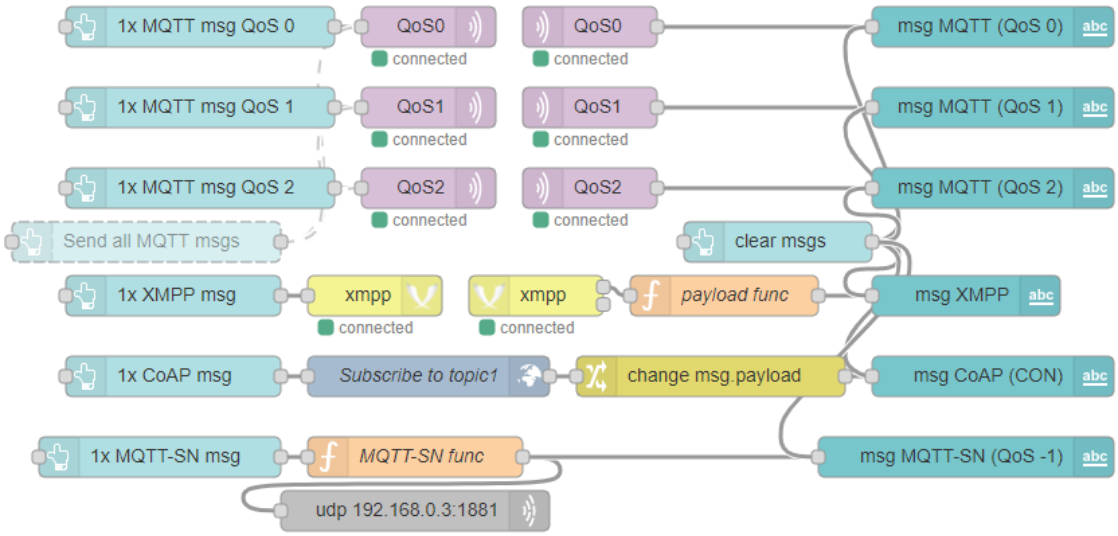
## 8.2 Vliv provozu sondy na IoT protokoly z pohledu bezpečnosti

Vzhledem k faktu, že sonda SonIoT využívá k detekování hrozeb detekční mechanismus na základě pravidel (signatur) a je dodávána s balíčkem pravidel „Emerging Threats“, které čítají více jak 20 tisíc pravidel, může nastat situace, že některé z těchto pravidel mohou vyhodnotit a případně zahodit pakety, obsahující IoT datové protokoly a tím znehodnotit, nebo zcela znemožnit např. sběr dat ze senzorové sítě, či chytrého IoT zařízení v domácnosti, a tím výrazně snížit kvalitu prožitku z dané služby.

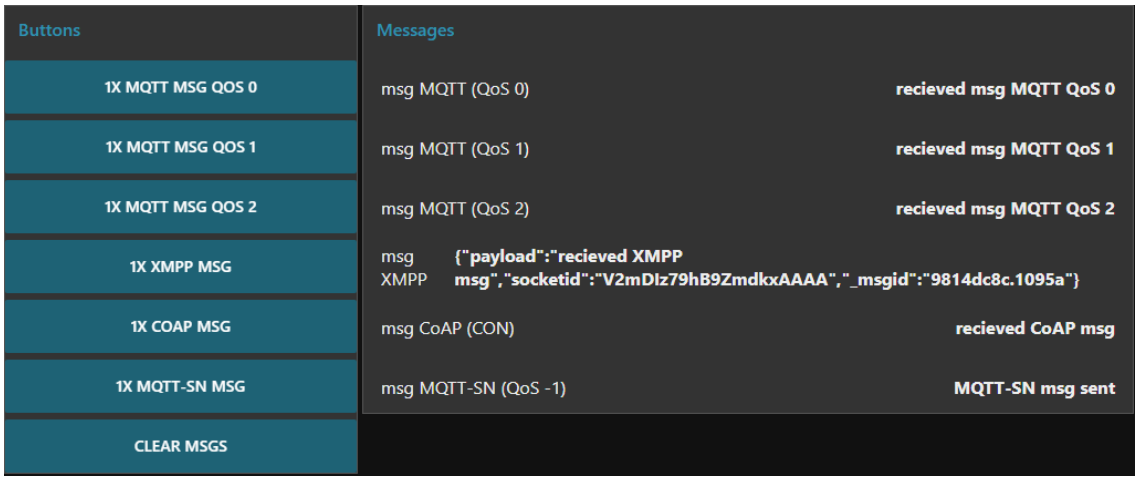
V rámci testbedu tak byla aplikace v Node-RED obohacena a funkci zasílání zpráv pomocí těchto IoT protokolů, Obr. 8.2 a Obr. 8.3:

- MQTT (vůči online MQTT brokerovi, QoS 0,1 a 2)
- XMPP (vůči online serveru, bez QoS)
- CoAP (vůči online serveru, QoS CON)
- MQTT-SN (vůči PC2, QoS úrovně -1)

V textu práce následuje série testů, které mají prokázat, zda IPS, případně i IDS, sonda nevyhodnotí některou z komunikací jako bezpečnostní hrozbu.




Obr. 8.2 Aplikace pro testování vlivu sondy na IoT protokoly v programovacím prostředí Node-RED.




Obr. 8.3 Grafické prostředí aplikace pro testování vlivu sondy na IoT protokoly.

Během měření bylo zjištěno, že kromě XMPP protokolu nebyla IPS systémem a ani IDS detekována žádná hrozba, Tab. 8.2. Pro XMPP komunikaci vyhodnotila sonda hrozbu „GPL CHAT Jabber/Google Talk Outgoing Traffic“ s prioritou 3, která má spíše informativní charakter. Paket tedy nebyl vyhodnocen jako nebezpečný a dle pravidel mohl skrze sondu projít.

Tab. 8.2 Vyhodnocení generování alarmů sondou.

| Protokol | Vygenerování alarmu |
|---|---|
| **MQTT** | Ne |
| **XMPP** | Ano |
| **CoAP** | Ne |
| **MQTT-SN** | Ne |

Z výsledků tohoto měření, Tab. 8.2, lze soudit, že v sítích, kde je aplikovaná sonda s defaultními pravidly, ať už v režimu IPS, či IDS, nebudou služby využívající tyto protokoly nikterak omezovány, a to ani pro různé úrovně QoS datových protokolů.

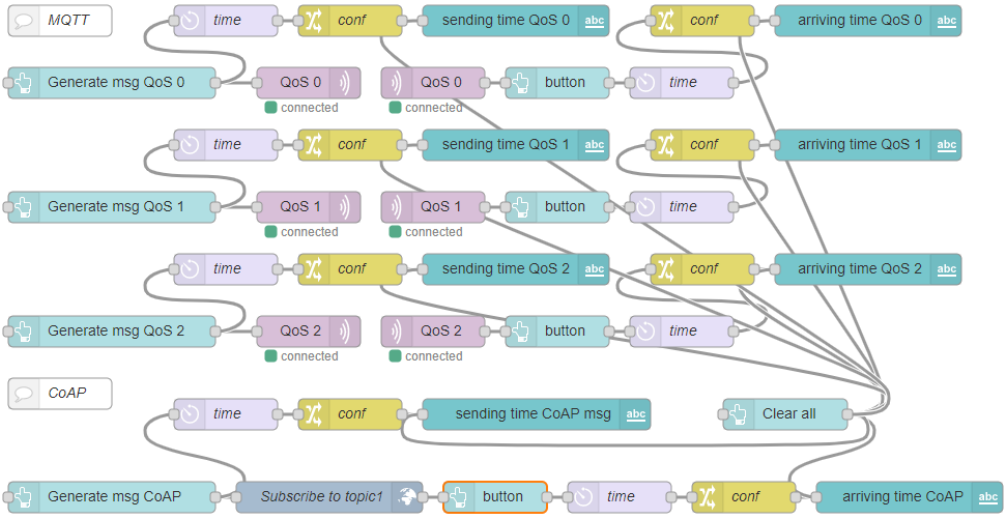
## 8.3 Vliv provozu sondy na QoS parametry IoT protokolů

V této kapitole je zkoumán vliv sondy na datové toky s využitím IoT protokolů. Pro testování byl zvolen jeden aplikační IoT protokol od každého transportního protokolu, MQTT (TCP) a CoAP (UDP). Tyto protokoly byly také zvoleny proto, že se s nimi v literatuře nejčastěji setkáváme ve spojitosti s využitím v oblasti IoT.

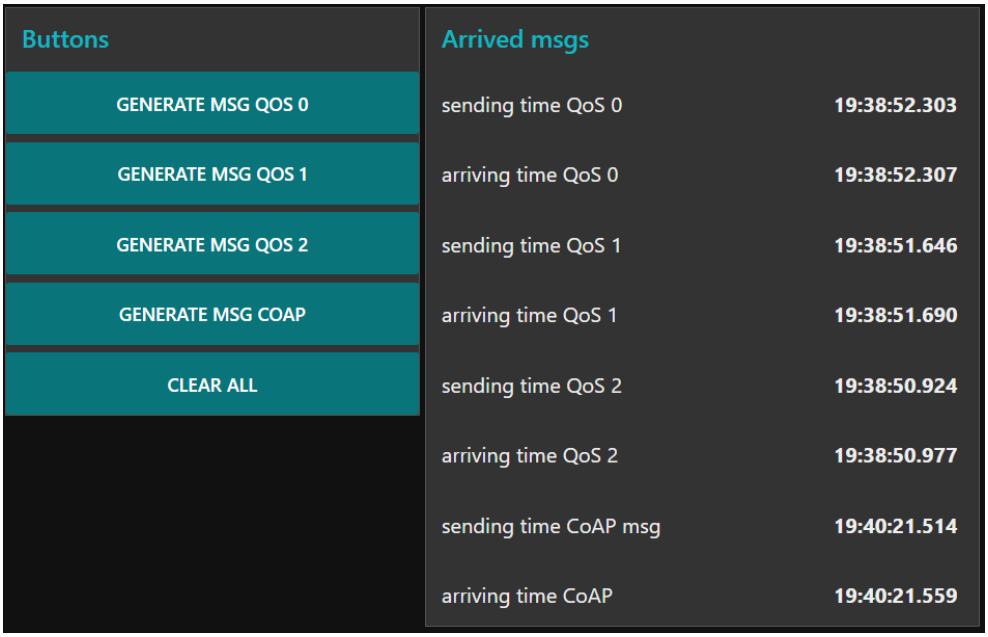
Pokud bychom se zaměřili na trh s IoT službami, tak převážná většina těchto služeb využívá non real-time komunikaci, ve které se klade důraz především na propustnost a zpoždění. Vzhledem k faktu, že maximální propustnost obou variant sond v režimu IPS je vysoká, v desítkách až stovkách Mbit/s, v jednom směru, což bylo naměřeno v kapitole 7, tak již v tuto chvílí můžeme předpokládat, že pro běžně využívané IoT služby (chytrá domácnost, senzorové sítě apod.) nebude ve většině případů dosaženo těchto limitů, a to jak pro TCP, tak UDP provoz. To samotné potvrzuje fakt, že tyto protokoly jsou navržené pro komunikaci do zařízení s omezenou výpočetní kapacitou, kde je vyžadován minimalistický datový provoz. Při využití IoT protokolů využívajících UDP transportní protokol je však nutné při využití sondy v síti, zvážit, při jakých hodnotách propustnosti dochází vlivem sondy ke ztrátovosti paketů. Vzhledem k principu činnosti UDP by při ztrátě paketu nedošlo k jeho opětovnému zaslání.

### 8.2.1 Vliv provozu sondy na zpoždění zpráv IoT protokolů

Aby mohl být ověřen vliv sondy na zpoždění doručování zpráv od zdroje k cíli, byla v rámci testbedu navržena jednoduchá rozšiřující Node-RED aplikace, která pracuje na principu časových značek (timestamps), Obr. 8.4 a Obr. 8.5.




Obr. 8.4 Aplikace pro testování vlivu sondy na zpoždění zpráv IoT protokolů v programovacím prostředí Node-RED




Obr. 8.5 Grafické prostředí aplikace pro testování vlivu sondy na zpoždění zpráv IoT protokolů.

Odesílatel zprávy směrem k serveru (k brokerovi v případě MQTT protokolu) do zprávy vloží časovou značku a ve chvíli, kdy dorazí zpráva k cíli, který je v našem případě stejný jako zdroj, se vytvoří nová časová značka. Tyto dvě časové značky se porovnají a jejich rozdíl nám udává hodnotu rozdílu času mezi odesláním a přijetím zprávy. Porovnáním naměřených hodnot při měření bez sondy a s ní pak můžeme zjistit, jaký časový přírůstek má sonda jako síťový element k celkové době šíření zprávy od zdroje k cíli.

**MQTT**

Při měření s protokolem MQTT měly všechny zaslané zprávy stejnou velikost paketů (TCP payload = 21 B), aby bylo měření co nejpřesnější. Jednotlivá měření pak probíhala se sondou a bez sondy a pro různé hodnoty QoS. Pro zpřesnění výsledků byl každý test opakován 10x a výsledná hodnota byla zprůměrovaná. Test byl prováděn pouze v rámci LAN, Obr. 8.1, kdy subscriber a publisher byli umístěni v PC a broker v PC2. Výsledky z měření lze nalézt v Tab. 8.3.

Tab. 8.3 Měření zpoždění MQTT zpráv s QoS 0-2.

| **Zpoždění zpráv – bez sondy [ms]** | | | | **Zpoždění zpráv – se sondou [ms]** | | | |
|---|---|---|---|---|---|---|---|
| **Měření** | QoS 0 | QoS 1 | QoS 2 | **Měření** | QoS 0 | QoS 1 | QoS 2 |
| **1** | 3 | 44 | 46 | **1** | 3 | 45 | 49 |
| **2** | 4 | 44 | 47 | **2** | 4 | 48 | 50 |
| **3** | 3 | 45 | 46 | **3** | 3 | 47 | 51 |
| **4** | 3 | 43 | 48 | **4** | 3 | 46 | 50 |
| **5** | 4 | 42 | 48 | **5** | 3 | 45 | 50 |
| **6** | 3 | 43 | 46 | **6** | 3 | 46 | 52 |
| **7** | 4 | 43 | 46 | **7** | 3 | 46 | 50 |
| **8** | 4 | 43 | 46 | **8** | 3 | 45 | 49 |
| **9** | 3 | 44 | 45 | **9** | 3 | 45 | 50 |
| **10** | 3 | 44 | 50 | **10** | 4 | 45 | 49 |
| **průměr** | 3.4 | 43.5 | 46.8 | **průměr** | 3.2 | 45.8 | 50 |

Z měření je patrné, že pro **QoS 0** je čas doručení zprávy v případě měření se sondou i bez ní přibližně stejný. Rozdíl dvou desetin by se dal označit jako neprůkazný, jelikož se od sebe naměřené hodnoty z každého měření téměř neliší. Pro zpřesnění výsledků by bylo třeba měřit i na mikrosekundy. Výrazný rozdíl lze pozorovat mezi QoS 0 a QoS 1, kde průměrná doba na doručení zprávy k subscriberovi vzrostla o cca 40 ms. To bude pravděpodobně způsobeno na straně brokera, který si u QoS 1 zprávu nejprve ukládá a pak až jí odesílá. Rozdíl u **QoS 1** při měření bez sondy a se sondu je již výraznější a to 2,3 ms. Pro **QoS 2** pak rozdíl v čase doručení při měření se sondou vrostl na 3,2 ms oproti měření bez sondy.

Z těchto výsledků lze usoudit, že sonda má vliv na zpoždění paketů nesoucích MQTT protokol. Zpoždění se se zvyšuje se zvyšující se úrovní QoS, což bude pravděpodobně způsobeno rozdílnými mechanismy ověřování doručení dle jednotlivých QoS mechanismů MQTT protokolu.

Vzhledem k faktu, že MQTT využívá jako transportní protokol TCP, tak lze z naměřených výsledků v rámci kapitol 7 a 8 říct, že pokud by hodnota RTT mezi subscriberem a publisherem byla konstantní, tak by zpoždění doručení zpráv jako je

uvedené v Tab. 8.3 bylo pro jednotlivé úrovně QoS stejné. Toto však platí pro nízké datové, resp. takové datové toky, při kterých ještě nedochází k zahlcení sítě. Při zahlcení výrazně roste hodnota RTT, tedy se zvýší interval mezi odesláním TCP zprávy nesoucí MQTT protokol a ACK zprávy potvrzující doručení (myšleno potvrzení TCP zprávy, ne MQTT). Na QoS 0 nebude mít zvýšená hodnota RTT takový dopad, jelikož subscriber nepožaduje potvrzení o doručení. V případě QoS 1 a QoS 2 by se však mohl, vzhledem k většímu počtu zpráv v procesu QoS, čas doručení subscriberovi výrazně zvýšit, a to samé v případě publishera, který očekává potvrzení. Aby byla zajištěna doručitelnost v nejkratším možném čase, je třeba uvažovat maximální využívanou přenosovou rychlost v jednom směru.

Vzhledem k velmi proměnlivým hodnotám naměřených propustnosti ve verzi Home, Tab. 7.3, měla by být přenosová rychlost v dané síti s využitím verze Home nižší než nejnižší naměřená hodnota, tedy přibližně 68 Mbit/s ve směru uplink anebo 75 Mbit/s ve směru downlink.

V případě verze Industry by měl být datový tok s MQTT protokoly do maximální rychlosti ve směru downlink do rychlosti 158 Mbit/s anebo ve směru uplink do 155 Mbit/s.

### CoAP

Při měření s protokolem CoAP měly zprávy příchozí (UDP payload = 43 B) i odchozí (UDP payload = 49 B) vždy stejnou velikost. Ve stávající verzi modulu v Node-RED nelze definovat QoS pro CoAP protokol a standardně je úroveň QoS nastavena na „confirmable“ (CON). Testování bylo prováděno vůči serveru coap.me na portu 5683, což je testovací server. Výsledky z měření lze vidět v Tab. 8.3.

Tab. 8.3 Měření zpoždění CoAP zpráv s QoS CON.

| CoAP QoS CON | | |
|---|---|---|
| **Měření** | **Bez sondy [ms]** | **Se sondou [ms]** |
| **1** | 41 | 42 |
| **2** | 43 | 43 |
| **3** | 43 | 43 |
| **4** | 42 | 43 |
| **5** | 43 | 42 |
| **6** | 40 | 43 |
| **7** | 40 | 43 |
| **8** | 43 | 45 |
| **9** | 43 | 43 |
| **10** | 43 | 44 |
| **Průměr** | 42.1 | 43.1 |

Z výsledků měření, Tab. 8.3, je patrné, že obousměrné zpoždění doručení zprávy narůstá při využití sondy o cca 1ms, což bude s největší pravděpodobností způsobeno právě sondou a jejím kontrolním mechanismem.

Jelikož CoAP využívá protokol UDP, který je bez záruky doručení, je pro zajištění doručitelnosti zprávy nutné při užití sondy přihlédnout k limitním hodnotám ztrátovosti paketů. V případě, že není CoAP zpráva doručena požadované B straně, tak nezáleží na tom, jakou úroveň QoS uživatel požaduje, jelikož není na co odpovědět. Z Tab. 7.6 je zřejmé, že sonda ve verzi Home dosahuje hranice ztrátovosti paketů 1 % při přenosové rychlosti přibližně 65 Mbit/s ve směru downstream a 75 Mbit/s ve směru upstream pro UDP provoz. Pro verzi Industry, Tab. 7.11, to pak je 150 Mbit/s ve směru downstream a 140 Mbit/s ve směru upstream.

Aby mohla být zaručena doručitelnost zpráv s využitím IoT protokolů využívajících jako transportní protokol UDP, měla by být hodnota přenosové rychlosti v jednom směru do takové hodnoty, která odpovídá požadované limitní hodnotě ztrátovosti paketů.

# 9. Závěr

Tvorba této diplomové práci mi přinesla mnoho nových poznatků z oblasti datových sítí, především z pohledu bezpečnosti a způsobů měření síťových parametrů. Také rozšířila mé znalosti z oblasti IoT, které jsem se věnoval v bakalářské práci. Během psaní teoretické části jsem se nesetkal s problémy spojenými s nedostatkem literatury, jelikož byly popisována témata, která nejsou nikterak nová a věnovalo se jim již mnoho knižních autorů a publicistů technicky zaměřených článků.

Druhá kapitola je zaměřena na popis významu slova testbed, který není příliš rozšířený.

Třetí kapitola je věnována tématu síťové bezpečnosti v oblasti datových sítí, kde jsou popsány jednotlivé možnosti síťové obrany/ochrany a následně jsou detailně popsány systémy pro detekci a prevenci průniku (IDS/IPS). Důraz je kladen především na popsání mechanismů detekce a možností umístění systémů v síti. Také je v této kapitole popsán IDS/IPS nástroj Suricata a bezpečnostní sonda SonIoT a její dvě verze – Home a Industry.

Ve čtvrté kapitole jsou projednávány způsoby měření parametrů datových sítí. jsou popsány tři známe metody, RFC 2544, ITU-T Y.1564 a RFC 6349, které se liší různými způsoby měření na různých vrstvách TCP/IP modelu. Na základě zjištěných poznatků z jednotlivých metod je vycházeno v praktické části, především pak z metody RFC 6349, která definuje způsob měření na třetí, transportní vrstvě TCP/IP modelu. Měřící nástroj využívající měření na této vrstvě je FlowTester, který představuje hlavní měřící nástroj pro praktickou část a je v této kapitole popsán z pohledu teorie. Ke konci kapitoly je popsána oficiální metodika ČTÚ pro měření QoS parametrů datových sítí a následuje popis QoE a jeho provázanosti s QoS.

Pátá, rozsáhlá kapitola pojednává o QoS v oblasti IoT. Popsány jsou tu především protokoly aplikační vrstvy TCP/IP modelu a jejich možnosti řízení QoS, jelikož QoS parametry na nižších vrstvách jsou již probírány v kapitole 4. Služby v oblasti IoT nelze nikterak přehlížet, jelikož dnes a denně statisíce senzorů generují enormní množství dat, jejichž bezpečnost je taktéž třeba zajistit, k čemuž může posloužit právě sonda SonIoT.

Šestou kapitolou začíná praktická část diplomové práce. Představuje komplexní popis sestaveného pracoviště. Lze tu nalézt např. použitý software, hardware, schéma zapojení jednotlivých komponent, popis způsobů měření pomocí FlowTesteru a popis činnosti aplikace pro měření parametrů sondy.

Sedmá kapitola zahrnuje samotné měření vlivu sondy na parametry přenosového kanálu. Jsou měřeny obě varianty sondy, a to v režimech IDS a IPS. Pro sondy v režimu IPS jsou měřeny především parametry jako je propustnost ve směru downlink a uplink, obousměrné zpoždění (RTT) a ztrátovost paketů. V případě měření sond v režimu IDS je sledováno množství dat, které je sonda schopna zpracovat, k čemuž je využita aplikace v Node-RED, resp. část pro měření přijatých dat na rozhraní sondy. V rámci každého měření jsou slovně zhodnoceny naměřené výsledky a důsledky plynoucí z výsledků. Také jsou pro každý režim měření porovnány naměřené výsledky pro verze Home a Industry.

V závěrečné, osmé kapitole praktické části jsou prováděny testy pro ověření vlivu sondy na QoS parametry datových IoT protokolů. Nejprve jsou provedeny testy pomocí aplikace v Node-RED, pomocí které jsou generovány zprávy s využitím protokolů MQTT, CoAP, XMPP a MQTT-SN a pomocí aplikace na sondě je ověřováno, zda sonda vyhodnotí protokoly jako rizikové. V následující podkapitole je provedena série testů s využitím MQTT (TCP) a CoAP (UDP) a je měřeno s pomocí naprogramované aplikace, jaké zpoždění přidává sonda do celkové doby šíření zprávy od zdroje k cíli. Toto je ověřováno pro tři úrovně QoS protokolu MQTT a jedné úrovně QoS pro protokol CoAP.

Hlavním přínosem této práce je realizované testovací pracoviště, které lze využít pro testování nejen bezpečnostní sondy SonIoT, ale obecně síťových prvků a jejich vlivu na kvalitativní parametry přenosového kanálu, jako je propustnost, obousměrné zpoždění, či ztrátovost paketů. Také naprogramované aplikace mohou posloužit k lepšímu pochopení procesů spojených s kontrolou datového toku uvnitř sondy a pomoci s dalšími testy v rámci testbedu. Naprogramované aplikace využívají především vnitřních funkcí OS Linux, a proto je lze využít také pro monitorování provozních parametrů síťových prvků založených na různých distribucích Linuxu.

Výsledky z jednotlivých měření mohou případným zájemcům o využití sondy pro jejich síť poskytnout přehled o reálných možnostech sondy a porovnání jednotlivých verzí jim může ulehčit výběr při rozhodování.

# Seznam zkratek

| | |
|---|---|
| AP | Access Point |
| BB | Bottleneck Bandwitdh |
| BDP | Bandwidth Delay Product |
| BEREC | Body of European Regulators for Electronic Communications |
| CoAP | Constrained Application Protocol |
| CPU | Central Processing Unit |
| CWND | Congestion Window |
| ČTÚ | Český telekomunikační úřad |
| DDoS | Distributed Denial of Service |
| DHCP | Dynamic Host Configuration Protocol |
| DIDS | Distributed Intrusion Detection Systems |
| DMZ | Demilitarized Zone |
| E2E | End-to-End |
| FP | FlowPing |
| FT | FlowTester |
| gMOS | Game Mean Opinion Score |
| HIDS | Host-based Intrusion Detection Systems |
| HIPS | Host-based Intrusion Prevention Systems |
| HTTP | Hypertext Transfer Protocol |
| HTTPS | Hypertext Transfer Protocol Secure |
| HW | Hardware |
| ICMP | Internet Control Message Protocol |
| IDS | Intrusion Detection Systems |
| IoT | Internet of Things |
| IPS | Intrusion Prevention Systems |
| IPv4 | Internet Protocol version 4 |
| ISP | Internet Service Provider |
| LAN | Local Area Network |
| M2M | Machine to Machine |
| MOS | Mean Opinion Score |
| MQTT | Message Queuing Telemetry Transport |
| MQTT-SN | Message Queuing Telemetry Transport – Sensor Network |
| MSS | Maximum Segment Size |
| MTU | Maximum Transmission Unit |

| | |
|---|---|
| NBA | Network Behavior Analysis |
| NGA | Next Generation Access Network |
| NIDS | Network-based Intrusion Detection Systems |
| NIPS | Network-based Intrusion Prevention Systems |
| NSM | Network Security Monitoring engine |
| OISF | Open Information Serucity Foundation |
| QoE | Quality of Experience |
| QoS | Quality of Service |
| RAM | Random Access Memory |
| REST | Representational State Transfer |
| RTT | Round-Trip-Time |
| RWND | Receive Window |
| SLA | Service Level Agreement |
| SSH | Secure Shell |
| SW | Software |
| TCP | Transmission Control Protocol |
| TLS | Transport Layer Security |
| UDP | User Datagram Protocol |
| VoIP | Voice over Internet Protocol |
| WAN | Wide Area Network |
| WIDS | Wireless Intrusion Detection Systems |
| WIPS | Wireless Intrusion Prevention Systems |
| WSN | Wireless Sensor Network |
| XMPP | Extensible Messaging and Presence Protocol |

# Literatura a zdroje

[1] SPACEY, John. 5 Examples of a Testbed. In: Simplicable [online]. Simplicable, 2017. [cit. 12.12.2019]. Dostupné z https://simplicable.com/new/testbed

[2] Test Bed [online]. tutorialspoint.com, Copyright 2019. [cit. 12.12.2019]. Dostupné z https://www.tutorialspoint.com/software_testing_dictionary/test_bed.htm

[3] SORIANO, Miguel. Moderní systémy zabezpečení. In: Techpedia [online]. Czech Technical University in Prague - Faculty of Electrical Engineering, 2017. [cit. 12.12.2019]. Dostupné z http://techpedia.fel.cvut.cz/download/?fileId=801&objectId=76

[4] KOVÁŘOVÁ, Jiřina. Síťová bezpečnost I. In: DocPlayer [online]. DocPlayer.cz, 2017. [cit. 12.12.2019]. Dostupné z https://docplayer.cz/19144946-Sitova-bezpecnost-i-zakladni-popis-zabezpeceni-1-uvod.html

[5] Intrusion Prevention Systems [online]. Valency Networks LLP/ Copyright © 2008. [cit. 12.12.2019]. Dostupné z https://www.valencynetworks.com/articles/intrusion-prevention-systems.html

[6] COOPER, Stephen. Intrusion Detection Systems Explained: 11 Best IDS Tools Reviewed. In: Comparitech [online]. Comparitech Limited, 2019. [cit. 13.12.2019]. Dostupné z https://www.comparitech.com/net-admin/network-intrusion-detection-tools/

[7] DI PIETRO, Roberto a Luigi V. MANCINI. Intrusion detection systems. New York: Springer, c2008. Advances in information security. ISBN 9780387772653.

[8] SAXENA, Manish. Next Generation Intelligent Network Intrusion Prevention System. Research maGma, 2017. ISBN 9781387015795.

[9] R. GOODALL, John, CONTI, Gregory, MA, Kwan-Liu. Visualization for Computer Security: 5th International Workshop, VizSec 2008, Cambridge, MA, USA, September 15, 2008, Proceedings. Cambridge, Massachusetts, USA: Springer, 2008. ISBN 978-3540859314.

[10] ARYA, Karm Veer, BHADORIA Robin Singh, S. CHAUDHARI, Narendra. Emerging wireless communication and network technologies. New York, NY: Springer Berlin Heidelberg, 2018. ISBN 9789811303951.

[11] CryptoCypher. Open Source IDS Tools: Comparing Suricata, Snort, Bro (Zeek), Linux. In: AT&T Cybersecurity [online]. AT&T Cybersecurity, 2018. [cit. 13.12.2019] Dostupné z https://cybersecurity.att.com/blogs/security-essentials/open-source-intrusion-detection-tools-a-quick-overview

[12] NBA [online]. VUMS DataCom, spol. s r.o., 2014. [cit. 12.12.2019]. Dostupné z https://www.datacom.cz/piktogramy/nba/

[13] PALA, Zdeněk. Zaútočte si na WiFi, je to jednoduché aneb Kategorie útoků na bezdrátové sítě a možnosti obrany. In: SystemOnLine [online]. CCB, spol. s r. o., 2012. [cit. 13.12.2019]. Dostupné z https://www.systemonline.cz/it-security/kategorie-utoku-na-bezdratove-site.htm

[14] COOPER, Stephen. 7 best intrusion prevention systems (IPSs). In: Comparitech [online]. Comparitech Limited, 2019.[cit. 13.12.2019]. Dostupné z https://www.comparitech.com/net-admin/ips-tools-software/

[15] Suricata | Open Source IDS / IPS / NSM engine [online]. OISF. [cit 14.12.2019]. dostupné z https://suricata-ids.org/

[16] PETTERS, Jeff. IDS vs. IPS: What is the Difference?. In: Varonis [online]. Varonis, 2018. [cit. 16.12.2019]. Dostupné z https://www.varonis.com/blog/ids-vs-ips/

[17] 9.1. Suricata.yaml — Suricata unknown documentation [online]. OISF, Copyright 2016. [cit. 14.12.2019]. Dostupné z https://suricata.readthedocs.io/en/suricata-5.0.0/configuration/suricata-yaml.html

[18] DNSstuff. What Is an Intrusion Detection System? Latest Types and Tools. In: DNSstuff [online]. SolarWinds Worldwide LLC, 2019. [cit. 14.12.2019]. Dostupné z https://www.dnsstuff.com/intrusion-detection-system

[19] SonIoT – bezpečnostní sonda [online]. ČVUT, Copyright © 2019. [cit. 14.12.2019]. Dostupné z: https://soniot.fel.cvut.cz/

[20] Český telekomunikační úřad. Zpráva o vývoji trhu elektronických komunikací 2012 – 2017 se zaměřením na rok 2017 [online]. Český telekomunikační úřad, 2018. [cit. 14.4.2019]. Dostupné z https://www.ctu.cz/sites/default/files/obsah/stranky/8179/soubory/zovt-finalniverze-opendata.pdf

[21] Úvod – NetMetr. [online]. CZ.NIC. [cit. 14.4.2019]. Dostupné z: https://www.netmetr.cz/cs/

[22] CLEMENT, J. Worldwide digital population as of October 2019. In: Statista [online]. Statista, Inc., 2019. [cit 16.12.2019]. Dostupné z https://www.statista.com/statistics/617136/digital-population-worldwide/

[23] VODRÁŽKA, Jiří, KOCUR, Zbyněk, VONDROUŠ, Ondřej, VOTAVA, Ondřej. Použití otevřených nástrojů pro testování přístupových sítí nové generace. In: elektrorevue [online]. ISES (International Science and Engineering Society, o.s.), 2019. [cit. 16.12.2019]. Dostupné z http://www.elektrorevue.cz/cz/download/pouziti-otevrenych-nastroju-pro-testovani-pristupovych-siti-nove-generace/

[24] ZAHRADNÍK, Pavel, ČTÚ. Základní principy: „Metodiky pro měření a vyhodnocení datových parametrů pevných komunikačních sítí“. In: Český telekomunikační úřad [online]. ČTÚ, 3. srpna 2016, Praha [cit. 16.12.2019]. Dostupné z https://www.ctu.cz/sites/default/files/obsah/stranky/97338/soubory/ctuprezentace-zakladniprincipymetodiky.pdf

[25] Český telekomunikační úřad. Stanovení základních parametrů a měření kvality služby přístupu k sítí internet. In: Český telekomunikační úřad [online]. Český telekomunikační úřad, 2014. [cit. 16.12.2019]. Dostupné z https://www.ctu.cz/cs/download/datovy_provoz/rizeni_datoveho_provozu_stanoveni_zakladnich_parametru_18_12_2014.pdf

[26] BRADNER, S.; McQuaid, J. RFC 2544. Benchmarking Methodology for Network Interconnect Devices [Online]. Březen 1999. [cit. 16.12.2019]. Dostupné z https://tools.ietf.org/html/rfc2544

[27] Český telekomunikační úřad. Metodika pro měření a vyhodnocení datových parametrů pevných komunikačních sítí. In: Český telekomunikační úřad [online]. Český telekomunikační úřad, 2016. [cit. 21.12.2019]. Dostupné z https://www.ctu.cz/sites/default/files/obsah/stranky/97338/soubory/metodika-pevne-site-v100.pdf

[28] ITU-T Recommendation Y.1564. Ethernet service activation test methodology [online]. ITU, únor 2016. [cit. 21.12.2019]. Dostupné z https://www.itu.int/rec/T-REC-Y.1564/en

[29] CONSTANTINE, B., FORGET, G., GEIB, R., SCHRAGE, R. RFC 6349. Framework for TCP Throughput Testing [online]. Srpen 2011. [cit. 21.12.2019]. Dostupné z https://tools.ietf.org/html/rfc6349

[30] BEREC. BEREC Guidelines on the Implementation by National Regulators of European Net Neutrality Rules. In: BEREC [online]. BEREC, 2016. [cit. 22.12.2019]. Dostupné z: https://berec.europa.eu/eng/document_register/subject_matter/berec/regulatory_best_practices/guidelines/6160-berec-guidelines-on-the-implementation-by-national-regulators-of-european-net-neutrality-rules

[31] Český telekomunikační úřad. Měření datových parametrů sítí pomocí TCP protokolu. In: Český telekomunikační úřad [online]. Český telekomunikační úřad, 2014. [cit. 21.12.2019]. Dostupné z https://www.ctu.cz/cs/download/datovy_provoz/rizeni_datoveho_provozu_metodika_mereni_17_12_2014_v0_4_5.pdf

[32] ČVUT v Praze, Fakulta elektrotechnická. Tisková zpráva. Unikátní FlowTester z Fakulty elektrotechnické ČVUT umožňuje lepší diagnostiku a monitorování telekomunikačních sítí. In: ČVUT – Fakulta elektrotechnická [online]. ČVUT – Fakulta elektrotechnická, 2016. [cit. 21.12.2019]. Dostupné z https://www.fel.cvut.cz/cz/vz/tz/2016-0329-TZ-FELCVUT-FlowTester_fin.pdf

[33] flowtester [online] ČVUT – Fakulta elektrotechnická. [cit 21.12.2019]. Dostupné z https://flowtester.fel.cvut.cz/

[34] SEDLÁK, Jan. Na ČVUT vyvinuli systém sledování sítí, zvládá i smart gridy a IoT. In: lupa [online]. Lupa.cz, 2016. [cit. 21.12.2019]. Dostupné z: https://www.lupa.cz/clanky/na-cvut-vyvinuli-system-sledovani-siti-zvlada-i-smart-gridy-a-iot/

[35] KOCUR, Zbyněk, VONDROUŠ, Ondřej, VOTAVA, Ondřej. F-Tester - TCP/IP testing platform [přednáška]. Praha: Fakulta elektrotechnická, ČVUT v Praze, In: indico.csnog.eu [PDF], [21. 12. 2019]. PDF dostupné na https://indico.csnog.eu/event/6/contributions/55/attachments/36/102/F-Tester_ENG_SNOG2019.pdf?fbclid=IwAR2CTnBQeIx_vz03Wygxqn0l0mw8W5YerFxZXLJ6y32bc_4OlxoG1AEATR8

[36] COHEN, Dor. MTU and MSS: What You Need to Know. In: IP MTU and TCP MSS Missmatch - an evil for network performance | APNIC Blog [online]. Imperva, 2017. [cit. 22.12.2019]. Dostupné z https://www.imperva.com/blog/mtu-mss-explained/

[37] iPerf - The TCP, UDP and SCTP network bandwidth measurement tool [online]. iPerf.fr. [cit. 21.12.2019]. Dostupné z: https://iperf.fr/

[38] FlowPing [online] Department of Telecommunication Engineering, FEE, CTU in Prague, 2016 Copyright ©.  [cit. 21.12.2019]. Dostupné z https://flowping.fel.cvut.cz/

[39] VONDROUŠ, Ondřej, MACEJKO, Peter, KOCUR, Zbyněk. FlowPing - The New Tool for Throughput and Stress Testing. In: ResearchGate [online]. ResearchGate, 2015. [cit. 21.12.2019]. Dostupné z https://www.researchgate.net/publication/289546111_FlowPing_-_The_New_Tool_for_Throughput_and_Stress_Testing

[40] OSIPOV, Evgeny. Wired/wireless internet communications: 8th international conference, WWIC 2010, Luleaa, Sweden, June 1-3, 2010 : proceedings. New York: Springer, c2010. ISBN 9783642133145.

[41] KUIPERS, Fernando, Robert KOOIJ, Danny DE VLEESCHAUWER a Kjell BRUNNSTRÖM. Techniques for Measuring Quality of Experience. OSIPOV, Evgeny, Andreas KASSLER, Thomas Michael BOHNERT a Xavier MASIP-BRUIN, ed. Wired/Wireless Internet Communications [online]. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, 2010, s. 216-227 [cit. 23.12.2019]. Lecture Notes in Computer Science. DOI: 10.1007/978-3-642-13315-2_18. ISBN 978-3-642-13314-5. Dostupné z: http://link.springer.com/10.1007/978-3-642-13315-2_18

[42] BRADA, Miloslav, ZELENKA, Jan. Posuzování kvality hlasu [online]. listopad 2008. [cit. 22.12.2019]. Dostupné z http://www.ip-telefon.cz/data/downld/40.pdf

[43] ZAHRADNÍK, Pavel, VIK, Tomáš, ČTÚ. Parametry služby z pohledu zákazníka – QoE (Quality of Experience). In: DocPlayer [online]. DocPlayer, 12. března 2015, Brno. [cit. 23.12.2019]. Dostupné z https://docplayer.cz/6264459-Parametry-sluzby-z-pohledu-zakaznika-qoe-quality-of-experience-metodika-ctu-pro-mereni-kvality-sluzby-pristupu-k-siti-internet.html

[44] SZIGETI, Tim, Christina HATTINGH, Robert BARTON a Kenneth BRILEY. End-to-end QoS network design. 2nd edition. Indianapolis, IN: Cisco Press, [2014]. Cisco Press networking technology series. ISBN 15-871-4369-0.

[45] ITU-T Recommendation Y.1541. Network performance objectives for IP-based services [online]. ITU, 2011. [cit. 24.12.2019]. Dostupné z https://www.itu.int/rec/T-REC-Y.1541-201112-I/en

[46] ITU-T Recommendation E.800. Definitions of terms related to quality of service [online]. ITU, 2008. [cit. 25.12.2019]. Dostupné z https://www.itu.int/rec/T-REC-E.800-200809-I/en

[47] HON, Petr. Jak funguje řízení datových toků s QoS. In: Jak funguje řízení datových toků s QoS – Connect.cz [online]. CZECH NEWS CENTER a.s., 2012. [cit. 25.12.2019]. Dostupné z https://connect.zive.cz/clanky/jak-funguje-rizeni-datovych-toku-s-qos/sc-320-a-161738/

[48] WHITE, Gary, Vivek NALLUR a Siobhán CLARKE. Quality of service approaches in IoT: A systematic mapping. Journal of Systems and Software [online]. 2017, 132, 186-203 [cit. 25.12.2019]. DOI: 10.1016/j.jss.2017.05.125. ISSN 01641212. Dostupné z: https://linkinghub.elsevier.com/retrieve/pii/S016412121730105X

[49] ZHOU, Hong a Zhongwei ZHANG. Differentiated Statistical QoS Guarantees for Real-Time CBR Services in Broadband Wireless Access Networks. In: 2010 International Conference on Computational Intelligence and Software Engineering [online]. IEEE, 2010, 2010, s. 1-4 [cit. 2020-01-01]. DOI: 10.1109/WICOM.2010.5601439. ISBN 978-1-4244-3708-5. Dostupné z: http://ieeexplore.ieee.org/document/5601439/

[50] SINGH, Manisha a Gaurav BARANWAL. Quality of Service (QoS) in Internet of Things. In: 2018 3rd International Conference On Internet of Things: Smart Innovation and Usages (IoT-SIU) [online]. IEEE, 2018, 2018, s. 1-6 [cit. 27.12.2019]. DOI: 10.1109/IoT-SIU.2018.8519862. ISBN 978-1-5090-6785-5. Dostupné z: https://ieeexplore.ieee.org/document/8519862/

[51] IoT Standards & Protocols Guide | 2019 Comparisons on Network, Wireless Comms, Security, Industrial. Postscapes | Internet of Things (IoT) & Connected Systems Trend Research [online]. Copyright © Postcapes .[cit. 27.12.2019]. Dostupné z: https://www.postscapes.com/internet-of-things-protocols/

[52] SATYAVRAT. IoT Protocols : An Overview. In: element14 [online]. Premier Farnell Ltd., 2017. [cit. 27.12.2019]. Dostupné z https://www.element14.com/community/groups/internet-of-things/blog/2017/02/05/iot-protocols-an-overview

[53] VOJÁČEK, Antonín. IoT MQTT prakticky v automatizaci - 1.díl – úvod. In: Automatizace.HW.cz | Elektronika v automatizaci [online]. HW server s.r.o., 2017. [cit. 27.12.2019]. Dostupné z https://automatizace.hw.cz/iot-mqtt-prakticky-v-automatizaci-1dil-uvod.html

[54] BALAJI, Abhinaya. Dissecting MQTT Using Wireshark. In: DZone [online]. Dzone, 2017. [cit. 27.12.2019]. Dostupné z https://dzone.com/articles/dissecting-mqtt-using-wireshark

[55] MALÝ, Martin. Protokol MQTT: komunikační standard pro IoT. In: Protokol MQTT: komunikační standard pro IoT - Root.cz [online]. Internet Info, s.r.o., 2016. [cit. 27.12.2019]. Dostupné z https://www.root.cz/clanky/protokol-mqtt-komunikacni-standard-pro-iot/

[56] MQTT QoS: Understanding Quality of Service. The purpose-built IoT portal for facilities management [online]. Copyright © Tribal Ltd. 1997-2020 .[cit. 28.12.2019]. Dostupné z: https://assetwolf.com/learn/mqtt-qos-understanding-quality-of-service

[57] JAFFEY, Toby. MQTT and CoAP, IoT Protocols. In: Eclipse [online]. Eclipse Foundation, Inc., 2014. [cit. 28.12.2019]. Dostupné z: https://www.eclipse.org/community/eclipse_newsletter/2014/february/article2.php

[58] GHOLKAR, Vidhya. Internet of Things (IoT) protocols COAP MQTT. In: SlideShare [prezentace]. LinkedIn Corporation © 2020, 2014. [cit. 28.12.2019]. Dostupné z https://www.slideshare.net/vgholkar/io-t-protocolsoscon2014

[59] Jabber – Jabber.cz Wiki [online].Jabber.cz. [cit. 28.12.2019]. Dostupné z: https://www.jabber.cz/wiki/Jabber

[60] What is XMPP Protocol in IoT | XMPP Server | XMPP Client. RF Wireless Vendors and Resources | RF Wireless World [online]. Copyright ©RF Wireless World 2012. [cit. 28.12.2019]. Dostupné z: http://www.rfwireless-world.com/IoT/XMPP-protocol.html

[61] XEP-xxxx: Quality of Service. XMPP | XMPP Main [online]. xmpp.org, Copyright © 1999. [cit. 29.12.2019]. Dostupné z: https://xmpp.org/extensions/inbox/qos.html

[62] COPE, Steve. Introduction to MQTT-SN (MQTT for Sensor Networks). In: steves-internet-guide [online]. Copyright © 2011-2020 Steve's internet Guide, 2019. [cit. 29.12.2019]. Dostupné z http://www.steves-internet-guide.com/mqtt-sn/

[63] STANFORD-CLARK, Andy, LING TRUONG, Hong. MQTT For Sensor Networks (MQTT-SN) Protocol Specification Version 1.2. In: mqtt.org [online]. 1999 – 2013 International Business Machines Corporation (IBM), 2013. [cit. 29.12.2019]. Dostupné z http://www.mqtt.org/new/wp-content/uploads/2009/06/MQTT-SN_spec_v1.2.pdf

[64] AVRAHAM, Zvi, Building Wireless Sensor Networks with MQTT-SN, RaspberryPi and Erlang. In: SlideShare [online]. 11. listopadu. 2013. [cit. 29.12.2019]. Dostupné z https://www.slideshare.net/nivertech/zvi-mqtts-foreuc2013

[65] FETTE, I., MELNIKOV, A. RFC 6455. The WebSocket Protocol [online]. prosinec 2011. [cit. 29.12.2019]. Dostupné z https://tools.ietf.org/html/rfc6455

[66] emNet WebSocket | SEGGER - The Embedded Experts [online]. SEGGER, Copyright © 2020. [cit. 29.12.2019]. Dostupné z: https://www.segger.com/products/connectivity/emnet/add-ons/related-products/websocket/

[67] Embedded Experience: MQTT over WebSocket. Embedded Experience [online]. [cit. 30.12.2019]. Dostupné z: http://embeddedexperience.blogspot.com/2014/09/mqtt-over-websocket.html

[68] The IOT Protocols The Base of Internet of Things Ecosystem | 14core.com. 14core.com | ideas comes reality [online]. 14CORE, Copyright © 2020. [cit. 30.12.2019]. Dostupné z: https://www.14core.com/the-iot-protocols-the-base-of-internet-of-things-ecosystem/

[69] Elektronika, IT, spotřební materiál a hračky | AB-COM.cz. Elektronika, IT, spotřební materiál a hračky | AB-COM.cz [online]. Copyright © 2003. [cit. 2.1.2020]. Dostupné z: https://www.ab-com.cz/

[70] Ubikode / SysUtils / iperf · GitLab. [online]. GitLab. [cit 2.1.2020]. Dostupné z: https://gitlab.com/UbikBSD/SysUtils/iperf

[71] Resources : Node-RED. Node-RED [online]. [cit. 2.1.2020]. Dostupné z: https://nodered.org/about/resources/

[72] SCHULZ, Greg. Software-defined data infrastructure essentials: cloud, converged, and virtual fundamental server storage I/O tradecraft. Boca Raton: CRC Press, Taylor & Francis Group, [2017]. ISBN 9781498738156.

[73] DOVROLIS, Constantinos. Passive and active network measurement: 6th International Workshop, PAM 2005, Boston, MA, USA, March 31-April 1, 2005 : proceedings. New York: Springer, c2005. ISBN 9783540255208