VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ
DEPARTMENT OF COMPUTER SYSTEMS

PROFILOVANIE SIEŤOVEJ PREVÁDZKY PRE MITIGÁCIU DDOS

PROFILING OF NETWORK TRAFFIC FOR DDOS MITIGATION

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE **ALEXANDRA LIGOCKÁ**
AUTHOR

VEDOUCÍ PRÁCE **Ing. MARTIN ŽÁDNÍK, Ph.D.**
SUPERVISOR

BRNO 2021

Vysoké učení technické v Brně
Fakulta informačních technologií

Ústav počítačových systémů (UPSY) Akademický rok 2020/2021

Zadání bakalářské práce

Studentka: **Ligocká Alexandra**
Program: Informační technologie
Název: **Profilování síťového provozu pro mitigaci DDoS**
Profiling of Network Traffic for DDoS Mitigation
Kategorie: Počítačové sítě

Zadání:

1. Seznamte se s měřením síťového provozu a motivací pro vytváření profilů služeb, IP adres a podsítí v síťovém provozu (obecně síťových entit).
2. Nastudujte relevantní literaturu z pohledu vytváření a uchování profilů, zaměřte se na použité příznaky v profilu a jejich aplikaci pro mitigaci DDoS.
3. Navrhněte metodu a příznaky pro profilování síťového provozu na různé úrovni detailu (např. zařízení, segment, podsíť, organizace).
4. Implementujte navrženou metodu formou prototypu.
5. Proveďte analýzu metody na reálných síťových datech.
6. Zhodnoťte výsledky a diskutujte možnosti dalšího pokračování.

Literatura:

- Dle pokynů vedoucího.

Pro udělení zápočtu za první semestr je požadováno:

- Body 1 až 3 zadání.

Podrobné závazné pokyny pro vypracování práce viz https://www.fit.vut.cz/study/theses/

Vedoucí práce: **Žádník Martin, Ing., Ph.D.**
Vedoucí ústavu: Sekanina Lukáš, prof. Ing., Ph.D.
Datum zadání: 1. listopadu 2020
Datum odevzdání: 12. května 2021
Datum schválení: 30. října 2020

Abstrakt

Cieľom tejto práce je stanoviť metriky pre detekciu Distributed Denial-of-Service (DDoS) útokov a stanovenie hraníc bežnej sieťovej prevádzky v danej počítačovej sieti na rôznej úrovni detailu. Na základe zvolených metrík a údajov o sieťových tokoch je vytvorený sieťový profil, ktorý je následne uložený v pamäti. Vrámci implementačnej časti sa táto práca venuje implementácií programu pre zber a výpočet stanovených metrík, ich spracovaniu, uloženiu a poskytuje jednoduché rozhranie poskytujúce prístup k uložených dátam.

Abstract

The aim of this work is to propose metrics for DDoS attacks detection and setting the thresholds of normal network traffic in a given computer network at different levels of detail. Based on the selected metrics and network flow data, a network profile is extracted and afterwards stored in memory. Within the implementation part, this work deals with the implementation of program for the collection and calculation of specified metrics, their processing, storage and provides a simple interface providing access to stored data.

Kľúčové slová

Mitigácia DDoS, Databázy časových radov, Monitorovanie sieťovej prevádzky, NetFlow, IPFIX, Nemea, RRDTool

Keywords

DDoS mitigation, Time Series databases, Network Monitoring, NetFlow, IPFIX, Nemea, RRDTool

Citácia

LIGOCKÁ, Alexandra. *Profilovanie sieťovej prevádzky pre mitigáciu DDoS*. Brno, 2021. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Martin Žádník, Ph.D.

Profilovanie sieťovej prevádzky pre mitigáciu DDoS

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracovala samostatne pod vedením pána Ing. Martina Žádnika, Ph.D. Uviedla som všetky literárne pramene, publikácie a ďalšie zdroje z ktorých som čerpala.

.....
Alexandra Ligocká
10. mája 2021

Poďakovanie

Chcela by som poďakovať Ing. Martinovi Žádnikovi, Ph.D. za vedenie tejto práce, odbornú pomoc, užitočné rady a ochotu pri konzultáciách.

Obsah

Zoznam obrázkov

Zoznam tabuliek

Výpisy kódu

Kapitola 1

Úvod

V dnešnej dobe predstavuje internet nevyhnutnú súčasť spoločnosti, ktorá s veľkým množstvom výhod prináša taktiež negatívne stránky. Čoraz častejšie sa stretávame s výskytom anomálií v sieťovej prevádzke, ktoré môžu ohroziť bezpečnosť, spoľahlivosť a dostupnosť sietí. Blokovanie dostupnosti internetových služieb môže spôsobiť následky v rozsahu od nepríjemnosti pre užívateľov internetových služieb až po nezanedbateľné finančné straty, napríklad pre webové stránky internetových obchodov. Útoky cielené na blokovanie dostupnosti počítačových systémov alebo služieb sú všeobecne označované ako útoky na odmietnutie služby Denial-of-Service (DoS) a distribuované odmietnutie služby Distributed Denial-of-Service (DDoS).

DDoS útok je druh kybernetického útoku, pri ktorom útočník zahlcuje servery a iné sieťové prvky enormným množstvom dát. Tieto útoky sú vedené so zámerom o znefunkčnenie služby pre legitímnych užívateľov, zvyčajne dočasným prerušením alebo pozastavením služieb hosťujúceho servera, čo sa môže prejaviť napríklad nedostupnosťou webovej stránky. DDoS útok je spustený z množstva zariadení, často distribuovaný v botnetoch. Znemožnenie takéhoto útoku častokrát nie je možné zastaviť blokovaním jedného zdroja, pretože prichádzajúca sieťová prevádzka zaplavuje obeť z viacerých rôznych zdrojov.

Neustále sa vyvíjajúca podoba útokov predstavuje tému pre mnohých výzkumníkov v odbore kybernetickej bezpečnosti. Útoky DDoS stále patria k najbežnejším problémom aj napriek dosiahnutému technologickému pokroku v poslednom desaťročí.

1.1 Ciele práce

Hlavným cieľom práce je navrhnúť metriky pre detekciu DDoS útokov a poskytnúť administrátorom informácie o bežných hodnotách sieťovej prevádzky pre dané časové obdobie v dobe prebiehajúceho útoku, čo uľahčí nastavenie mitigačných pravidiel. Hodnoty jednotlivých metrík sú získané agregáciou údajov o sieťových tokoch obsiahnutých v hlavičke paketu a ďalej použité pre vytvorenie profilu sieťovej entity na rôznej úrovni detailu. Výsledky tejto práce budú môcť byť použité na zmiernenie dopadu DDoS útokov, umožnenie sieťovým administrátorom stanovenia hraníc bežnej sieťovej prevádzky a odhalenie podozrivých vzorov správania.

V tejto práci sú skúmané najmä objemové charakteristiky pre rôzne úrovne detailov (napríklad služba, IP adresa, podsieť), uhlov pohľadu (počet kontaktovaných podsietí, po-

čet služieb, počet prenesených bajtov) a objemových charakteristík (napríklad počet tokov, paketov, bytov, použitých portov).

1.2 Štruktúra práce

Táto práca je rozdelená do jednotlivých kapitol. Druhá kapitola objasňuje dôležitosť monitorovania sietí, vytvárania profilov sieťových entít, prístupy k monitorovaniu sietí a niektoré z dostupných monitorovacích technológií. Vrámci tretej kapitoly sú diskutované DDoS útoky, ich klasifikácia do skupín a možné spôsoby obrany voči nim. Druhá časť tejto kapitoly obsahuje návrh metrík pre vytváranie profilu siete. Ďalšia kapitola tejto práce je venovaná návrhu implementácie a použitým technológiám. Piata kapitola nadväzuje na predchádzajúcu kapitolu a popisuje spôsob implementácie, použité postupy a nastavenia jednotlivých technológií. Šiesta kapitola obsahuje popis dosiahnutých výsledkov vrámci tejto práce, predovšetkým popis experimentov, zhodnotenie pamäťovej náročnosti a výstup experimentov.

Kapitola 2

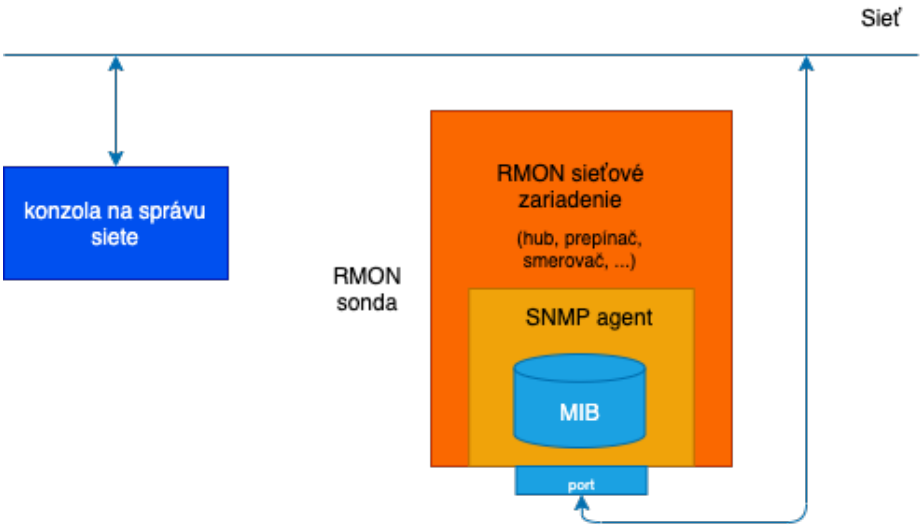
Monitorovanie siete a profily sieťových entít

Motiváciou pre monitorovanie sietí a vytváranie sieťových profilov je zlepšenie situačného povedomia s cieľom odhaliť podozrivé či škodlivé správanie. Prehľad o správaní jednotlivých prvkov v sieti je dôležitým krokom k efektívnej a spoľahlivej identifikácií a obrany voči kybernetickým útokom. Vrámci tejto kapitoly budú diskutované prístupy k monitorovaniu sietí a existujúce riešenia. Ďalej je popísaný prínos a postup vytvárania profilov sieťových entít.

2.1 Monitorovanie siete

Monitorovanie sietí je základným nástrojom pre mnohých správcov sietí pre porozumenie správania siete, no môže byť použité aj na získanie prehľadu o príčinách preťaženia siete, odhalenie abnormálnych vzorov správania alebo pre účely vystavovania faktúr na základe využívania siete. Hlavnou funkciou monitorovania siete je včasná identifikácia trendov a vzorov správania v sieťovej prevádzke.

Systémy na monitorovanie siete zahŕňajú softvérové a hardvérové nástroje, ktoré dokážu sledovať rôzne aspekty siete a jej fungovanie. Monitorovanie dnešných sietí je možné rozdeliť do dvoch hlavných skupín. Prvá skupina techník je založená na inšpekcií obsahu paketu. Na tomto princípe je založená väčšina systémov IDS. Druhou skupinou je zber a analýza dát, ktorá je uskutočňovaná na rôznej úrovni detailu v závislosti od potrieb užívateľa.
Pre získavanie informácií o sieťach je nutné využiť pokročilé a spoľahlivé techniky pre získavanie detailných štatistík. Monitorovanie siete na základe IP tokov umožňuje zhromažďovať a zaznamenávať kompletnú sieťovú prevádzku smerujúcu z a do zariadenia. Tento prístup spracováva informácie z hlavičiek paketov a agreguje ich do záznamov, nazývaných IP toky. Všeobecne sieťový tok predstavuje sekvenciu paketov so zhodnou zdrojovou a cieľovou IP adresou, zhodným zdrojovým a cieľovým portom a číslom protokolu. Výhodou tohto princípu je, že umožňuje spracovať väčší objem sieťovej prevádzky pri vyšších rýchlostiach prenosu. Údaje o tokoch v sieti poskytujú detailnejšie informácie ako napríklad využitie protokolu SNMP. Ďalšou možnou alternatívou je využitie architektúry Remote Monitoring (RMON), ktorá je rozšírením SNMP. Hlavnou časťou tejto architektúry sú RMON sondy (nazývané agenti), ktoré dokážu zbierať štatistiky o rozhraniach a vytvárať históriu pre vybrané štatistiky. Taktiež umožňujú nastaviť prahové hodnoty, vytvárať udalosti (alerty) pri prekročení týchto prahových hodnôt a ďalšie funkcie [36].



Obr. 2.1: RMON architektúra [18]

Ako je zobrazené na obrázku 2.1, úlohou RMON agentov je monitorovať údaje na sieti. Tieto údaje si pomocou príkazov SNMP získava konzola RMON. Existujú dve špecifikácie vzdialeného monitorovania siete - RMON1 a jeho rozšírenie RMON2. Zariadenia SNMP, napríklad rozbočovač, zvyčajne potrebujú ďalší softvér, ktorý ho premení na RMON sondu. Informácie, ktoré môže vyžadovať RMON konzola od agenta, sú uložené v hierarchickej databáze - Management information base (MIB) a patria medzi ne:

- štatistiky,
- história,
- alarmy,
- záznamy N najaktívnejších pripojení v konkrétnom časovom rámci,
- informácie o zachytených paketoch,
- informácie o udalostiach, ...

RMON2 MIB poskytuje rozšírenie informácií poskytovaných RMON1 MIB. Okrem už spomenutých k nim patria zoznam protokolov, ktoré môže sonda monitorovať, história používateľov či konfigurácia sondy [18].

Požiadavky na výkon a bezpečnosť sietí sa v posledných rokoch zvýšili a preto sú spomenuté riešenia nedostatočné. V dnešnej dobe sa medzi najpoužívanejšie technológie, využívajúce princíp monitorovania na základe tokov radí Netflow, opísaná v podkapitole 2.3 vyvinutá spoločnosťou Cisco Systems.

2.2 Profily sieťových entít

Údaje o sieťových tokoch sú agregáciou informácií z hlavičky obsiahnutej v datagramoch (paketoch) a je možné ich použiť na vytvorenie profilu sieťového prenosu, detekciu škodlivej premávky, určenie vhodných nastavení rôznych detekčných systémov a uľahčenia nastavení

mitigačných pravidiel pre daný časový interval v dobe útoku. Tieto údaje zahŕňajú rôzne informácie o komunikujúcich zariadeniach, ako je napríklad zdrojová a cieľová IP adresa, porty, použité protokoly, rôzne agregácie bajtov a mnohé ďalšie. Profil sieťovej entity predstavuje „súpis“ všetkých aktivít v sieti. Tieto profily umožňujú sledovať diania vrámci siete, poskytujú informácie pre nastavenia zabezpečenia sietí a môžu odhaliť prvky siete, ktoré porušujú pravidlá alebo vykazujú podozrivú aktivitu.

2.3 Monitorovanie sieťového toku

V technológií Netflow [1] je tok popísaný ako postupnosť IP paketov prechádzajúcich určitým bodom v sieti za jednotku času [10]. Každý dátový tok je identifikovaný pomocou piatich až siedmych kľúčových atribútov:

- zdrojová IP adresa,
- cieľová IP adresa,
- zdrojový port,
- cieľový port,
- protokol,
- typ služby,
- rozhranie monitorovacieho zariadenia.

Tieto údaje môžu byť ďalej analyzované a agregované pre vytváranie rôznych pohľadov na prevádzku v sieti. Túto technológiu je možné použiť aj vo vysoko-rýchlostných sieťach, vďaka tomu, že zbiera len vybrané informácie o tokoch a tým sa redukujú nároky na pamäť a výpočtový výkon.

Architektúra NetFlow, zobrazená na obrázku 2.2 obsahuje tri hlavné komponenty:

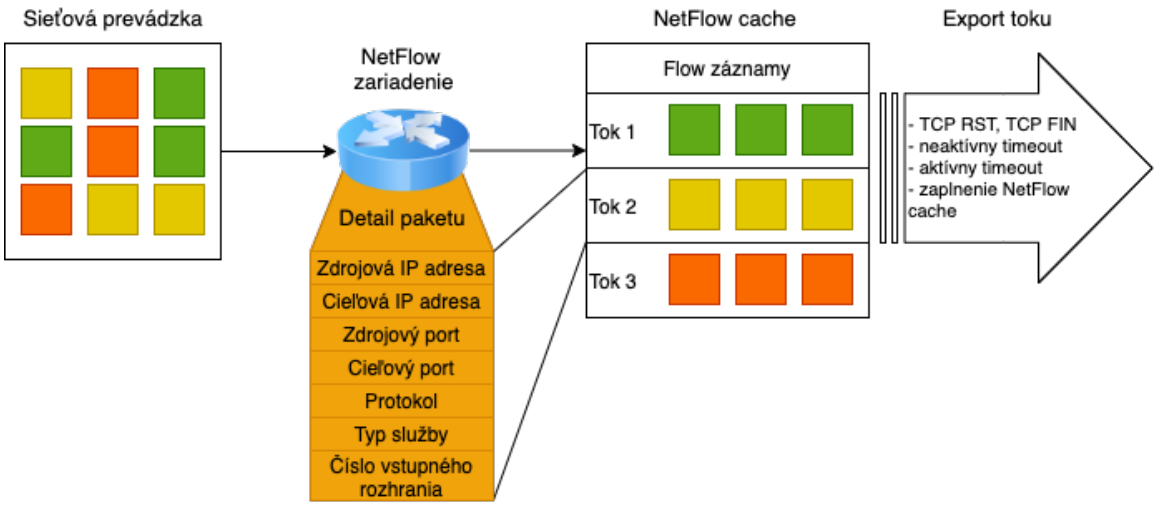
- NetFlow exportér (agent), ktorý je typicky pripojený do monitorovanej siete. Hlavnou úlohou tohto prvku je analyzovať prechádzajúce pakety a agregovať ich do sieťových tokov. Ukončené toky sú exportované na jeden alebo viac kolektorov pomocou NetFlow protokolu. Medzi voľne dostupných agentov patrí *fprobe*[1]. Z komerčne dostupných agentov, *nProbe*[2], ktorý dokáže exportovať toky vo formáte NetFlow alebo IPFIX.
- NetFlow kolektor je prvok, ktorý je schopný prijímať toky odoslané exportérom. Nad prijatými tokmi vykonáva agregáciu, filtrovanie, kompresiu dát a ukladá záznamy na neskoršie použitie.
- NetFlow analyzátor predstavuje aplikáciu, ktorá analyzuje informácie o tokoch zhromaždených kolektorom.

[1] http://fprobe.sourceforge.net/
[2] https://www.ntop.org/products/netflow/nprobe/

Obr. 2.2: Architektúra technológie NetFlow [28]

Princíp monitorovania tokov na základe technológie NetFlow je ilustrovaný na obrázku 2.3 [28]. V určitom bode siete dochádzka k monitorovaniu pomocou sieťového zariadenia. Pomocou kľúčových atribútov je paket pridaný do už existujúceho toku, uloženého v pamäti monitorovacieho zariadenia alebo je pridaný ako nový tok. Pamäť zariadenia sa nazýva *NetFlow cache*, predstavuje databázu tokov, ktoré sú uložené až do momentu odoslania informácií. Okrem základných atribútov je zaznamenávaný aj počet paketov tvoriacich tok, počet prenesených bajtov, čas začiatku toku (čas príchodu prvého paketu) a čas konca toku (čas príchodu posledného paketu daného toku). K exportu informácií dôjde po vypršaní platnosti konkrétneho záznamu a to nastane pri:

- detekovaní konca toku, napríklad pri príchode TCP paketu s príznakom RST alebo FIN,
- neaktivite toku, označovanom aj neaktívny timeout, kedy určitý čas neboli pozorované ďalšie pakety a predpokladá sa, že tok bol ukončený. Časový rámec nečinnosti je zvyčajne nakonfigurovaný na 15 sekúnd.
- príliš dlho trvajúcom toku - aktívny timeout, kedy predvolené nastavenie časového limitu môže byť až 30 minút,
- zaplnení *NetFlow cache* pamäte.



Obr. 2.3: Monitorovanie toku pomocou technológie NetFlow

2.4 Protokoly pre prenos tokov

Po ukončení toku je nutné aby boli informácie odoslané z exportéra na kolektor. Existuje niekoľko protokolov, ktorými môže byť tento prenos realizovaný. Medzi najvýznamnejšie protokoly používané pre prenos záznamov sieťového toku patrí už spomínaný NetFlow, jeho najviac využívané verzie NetFlow v5 a NetFlow v9 a IP Flow Information Export (IPFIX), ktorý bol vytvorený so zámerom vytvoriť spoločný univerzálny štandard exportu informácií na základe protokolu NetFlow v9, skupinou Internet Engineering Task Force (IETF) pre zjednotenie štandardov prenosu záznamov sieťových tokov[39]. Štandard IPFIX definuje akým spôsobom sa majú informácie formátovať a prenášať medzi exportérom a kolektorom.

2.5 Systém NEMEA

Systém NEMEA[7] (Network Measurements Analysis) je detekčný systém monitorujúci tok dát na analýzu sieťového prenosu. V praxi ide o súbor nezávisle bežiacich modulov NEMEA, ktoré nepretržite spracúvajú prichádzajúce dáta (správy). Správy zvyčajne obsahujú informácie o sieťových tokoch (napríklad formáty NetFlow alebo IPFIX), ale správy sú všeobecnejšie - môžu predstavovať zistené bezpečnostné udalosti. Modul NEMEA predstavuje spustiteľný súbor, ktorý je možné v operačnom systéme spustiť vo viacerých inštanciách.

Tento systém sa skladá z troch hlavných častí, zobrazených na obrázku 2.4:

1. NEMEA moduly - jednotlivé moduly majú rôzne využitie, systém obsahuje viacero modulov na detekciu podozrivej premávky, moduly na výpočet štatistík, filtrovanie, agregáciu správ, a iné. Tieto moduly prijímajú, spracovávajú a posielajú správy pomocou vstupných a výstupných rozhraní (IFC), ktorých počet je daný na základe vývojára a účelu modulu. Podľa účelu daného modulu ich môžeme rozdeliť na:

   - Detektory, ktoré zisťujú škodlivý prenos ako je DoS skenovanie, DNS tunelovanie a iné.
   - Moduly, ktoré sú zodpovedné za export a ukladanie údajov o tokoch, predspracovanie, filtrovanie, agregáciu, a iné.

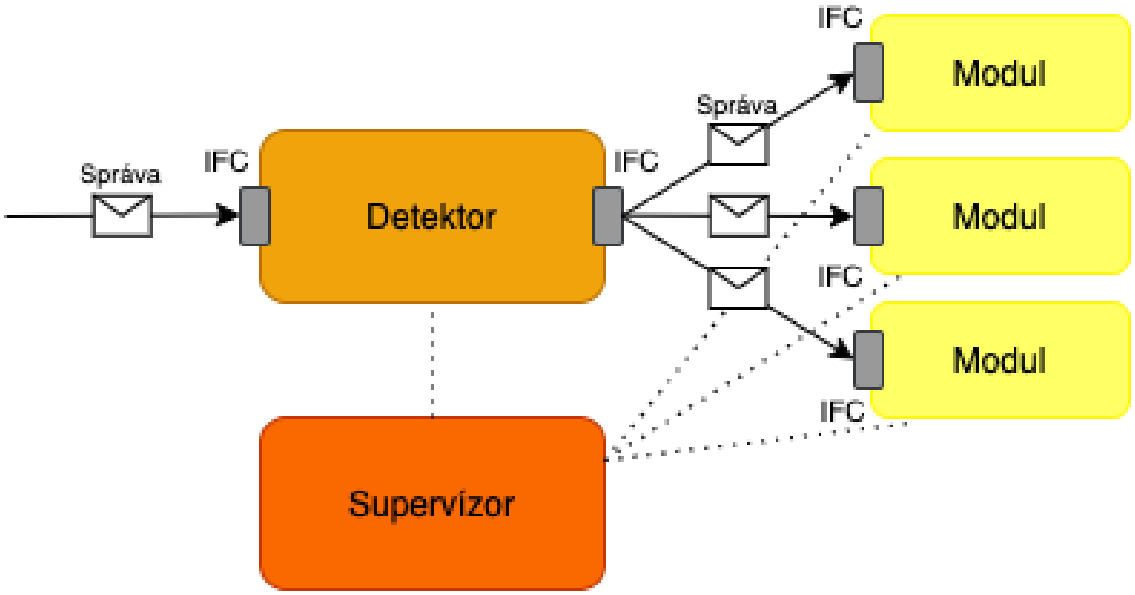
2. NEMEA Framework - táto časť systému obsahuje implementáciu spoločných funkcií všetkých modulov.

   - TRAP (Traffic Analysis Platform) predstavuje súbor funkcií zabezpečujúcich odosielanie a prijímanie správ medzi jednotlivými rozhraniami modulov. Táto knižnica implementuje obojsmerné rozhranie, ktoré reprezentuje vstup a výstup jednotlivých modulov. Moduly si medzi sebou vymieňajú rôzne druhy informácií, napríklad NetFlow dáta, výsledky detekcií, štatistické údaje, dáta vo formáte JSON a iné. Pri spúšťaní modulu je nutné špecifikovať rozhranie, na ktorom bude modul prijímať dáta a na ktorom bude dáta odosielať. Na komunikáciu je možné využiť sockety - Unixový, TCP, TLS, rozhranie súborov a ako výstupné rozhranie aj *black hole*, ktoré slúži na zahodenie všetkých správ.
   - UniRec (Unified Record) implementuje efektívny dátový formát odosielaných a prijímaných správ. Formát UniRec predstavuje súbor neštruktúrovaných záznamov. Skladá sa z polí (*fields*), ktoré obsahujú prenášané dáta a taktiež pre každé

z týchto polí je špecifikovaný dátový typ. Pre efektívny prístup k jednotlivým položkám sú vytvárané šablóny (*templates*) na popis formátu jednotlivých správ. Výhodou tohto formátu je jeho podobnosť so štruktúrami v jazyku C. Táto vlastnosť umožňuje jednoduchý prístup k jeho jednotlivým položkám. Oproti bežným štruktúram je možné definovať položky s dynamickou dĺžkou, pridávať položky do šablón za behu programu a na prístup k členu štruktúry je potrebný menší počet prístupov do medzipamäte CPU.

- Common knižnica obsahuje implementáciu dátových štruktúr a bežných algoritmov používaných v moduloch.

3. Supervízor - nástroj, ktorý sa stará o chod modulov podľa zadanej konfigurácie. Pomocou tejto časti systému je možné nastaviť spustenie a konfiguráciu viacerých modulov súčasne.



Obr. 2.4: Základné časti systému NEMEA[7]

Značnou výhodou systému NEMEA je jeho modulárnosť, ktorá je dosiahnutá vďaka spoločnému rozhraniu určenému ku komunikácií a prepojeniu jednotlivých modulov. Vďaka tejto vlastnosti je možné kombinovať rôzne moduly a pripájať moduly k už bežiacim bez nutnosti ich zastavenia či prerušenia. K ďalším výhodám patrí fakt, že je vhodný na analýzu okamžite zachytených dát o tokoch, ale aj spracovanie uložených dát. V tejto práci sú použité rôzne technológie, ktoré budú opísané v nasledujúcich kapitolách. NEMEA predstavuje jej najvýznamnejšiu časť, pretože sa používa na zhromažďovanie a manipuláciu so sieťovými dátami.

Kapitola 3

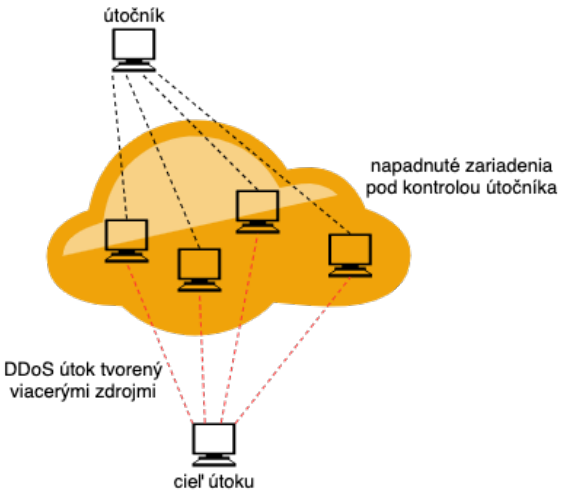
Klasifikácia DDoS útokov a metriky pre ich mitigáciu

Poznanie a porozumenie jednotlivých typov DDoS útokov je kľúčovým krokom k efektívnemu a spoľahlivému vývoju ochranného mechanizmu. Prvá časť tejto kapitoly popisuje DDoS útoky a ich klasifikáciu, opisuje najčastejšie sa vyskytujúce druhy DDoS útokov a druhá časť sa venuje známym obranným mechanizmom. Informácie v tejto kapitole boli čerpané prevažne z prác [16] a [14].

Nasledujúca kapitola naväzuje na túto kapitolu a sú v nej diskutované metriky navrhnuté na základe najčastejšie sa vyskytujúcich DDoS útokov použité pre vytváranie profilu sieťovej entity. V kontexte tejto práce je pojem metrika chápaná ako sledovaná vlastnosť sieťovej entity. Hodnoty týchto metrík môžu byť použité na stanovenie bežných hraníc sieťovej prevádzky pre nastavenie mitigačných nástrojov.

3.1 DDoS útoky a ich klasifikácia

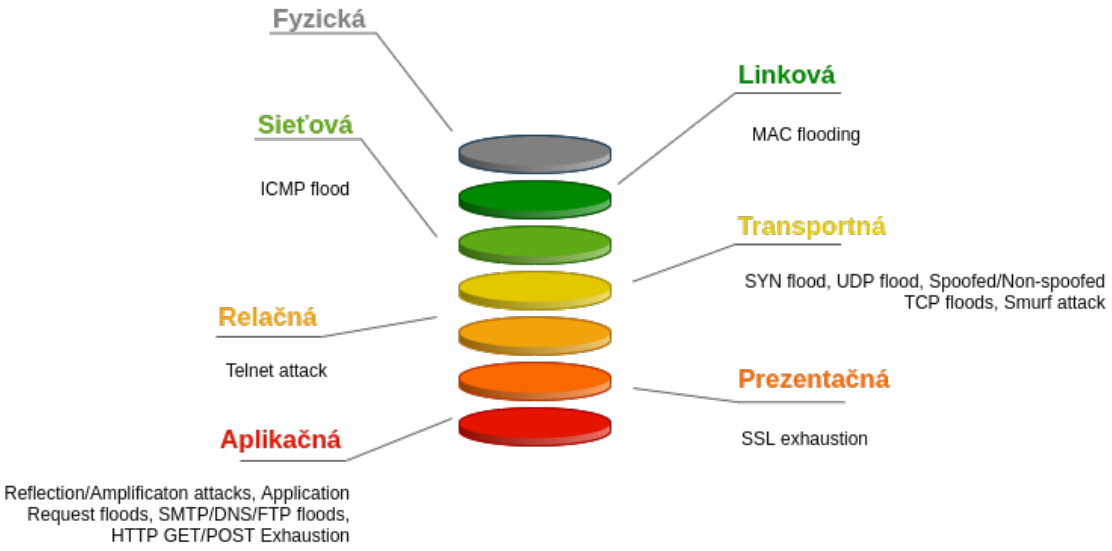
Útoky typu DDoS - odmietnutie služby, tvoria jednu z hlavných hrozieb počítačových sietí a patria k najťažším bezpečnostným problémom internetu. Obzvlášť závažným zásahom do spoľahlivej sieťovej prevádzky je útok typu DDoS - distribuované odmietnutie služby, pre masívny, koordinovaný útok, ktorý dokáže v krátkom čase vyčerpať zdroje svojho cieľa a odoprieť legitímnym užívateľom prístup k službám. Tento typ útoku využíva sieť počítačov k vyprodukovaniu prevádzky voči serveru, na ktorom beží cieľová služba útoku. Častokrát sú tieto útoky vedené bez vedomia majiteľov útočiacich počítačov následkom napadnutia a úspešného infikovania systémov. Toto väčšie množstvo útočiacich zariadení komplikuje zastavenie útoku pomocou filtrovania vstupu a taktiež je obtiažne rozoznať prenos oprávnených užívateľov nakoľko prenos pochádza z viacerých zdrojov. Niektoré útoky zahŕňajú falšovanie IP adries (*spoofing* IP adries) odosielateľov čo komplikuje identifikáciu útoku.



Obr. 3.1: Útočná sieť zariadení[20]

Útočná sieť počítačov je tvorená najmä zraniteľnými zariadeniami v sieti. Zraniteľné počítače v sieti môžu predstavovať tie, ktoré nemajú žiadny antivírusový softvér, antivírusový softvér nie je aktualizovaný alebo zariadeniam chýbajú aktualizácie systému a tak nie sú zabezpečené niektoré systémové chyby, ktoré už boli odhalené. Tieto zariadenia boli infikované škodlivým softvérom, čo umožňuje, aby ich vzdialene ovládal útočník, označujú ako roboti (alebo zombie) a skupina robotov sa nazýva botnet. Botnety, môžu byť okrem šírenia DDoS útokov navrhnuté tak, aby plnili aj iné nelegálne, alebo škodlivé úlohy zahŕňajúce odosielanie nevyžiadanej pošty, krádeže údajov či podvodného klikania na reklamy[11].

Rôzne typy útokov sa zameriavajú na rôzne časti sieťového pripojenia. Existuje 7 vrstiev modelu OSI, ktoré sú uvedené na obrázku 3.2 a k jednotlivým vrstvám aj príklady útokov.



Obr. 3.2: Vrstvy modelu OSI a príklady útokov na jednotlivých vrstvách

Pri DDoS útokoch na aplikačnej vrstve sa útočníci zameriavajú na procesy na aplikačnej vrstve. Protokoly aplikačnej vrstvy sa delia na dve hlavné kategórie - používateľské a pod-

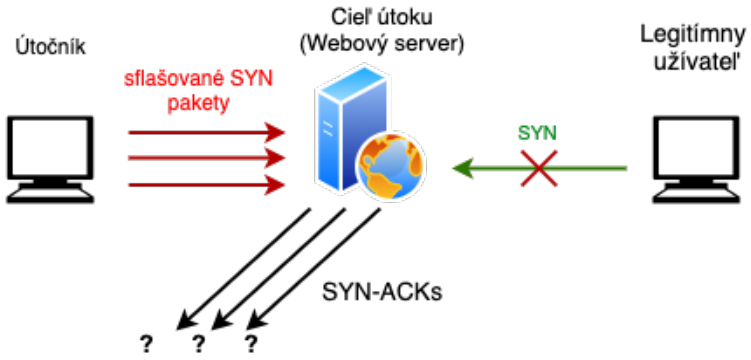
porné protokoly. Používateľské protokoly poskytujú užívateľom služby, napríklad pomocou protokolu HTTP, SMTP, SSH a iné. Cieľom podporných protokolov je poskytovať spoločné funkcie systému ako je napríklad DNS a TLS. Ktorýkoľvek z týchto protokolov môže byť zneužitý pre iniciovanie útoku, ktorý sa vykonáva cielene na danú službu. Príkladom takéhoto útoku je *HTTP Flood* kedy útočníci zaplavujú cieľový server množstvom požiadaviek HTTP, čo vedie k odmietnutiu služby pre legitímnych užívateľov[12].

Útoky na prezentačnej vrstve zneužívajú protokol TLS/SSL, ktoré sú použité najmä pre zabezpečenie webových služieb (online nakupovanie, online bankovníctvo, ...). SSL DDoS útok prebieha odosielaním bezcenných údajov na server, čo spôsobí problémy s pripojením pre legitímnych používateľov alebo zneužitím overenia totožnosti[27].

Relačná vrstva poskytuje synchronizáciu a ukončovanie pripojení. Útočník môže využiť chyby v serverovom softvéri Telnet bežiacom na prepínači, čo znemožňuje služby a bráni užívateľom vo vykonávaní správy[5].

Útoky smerované na transportnú vrstvu sú založené na generovaní enormného objemu sieťovej premávky na zablokovanie dostupnosti služieb alebo zdrojov, zvyčajne zahŕňajú zneužitie protokolov TCP a UDP ICMP a DNS a zameriavajú sa na vyčerpanie šírky pásma siete. Tieto útoky sú najčastejšie klasifikované ako záplavové útoky (*flooding attacks*) a súvisia s útokmi na aplikačnej vrstve.

- *Flooding attacks*: Hlavným cieľom útočníkov je vyčerpanie pásma siete užívateľa zaslaním veľkého počtu paketov (napríklad *UDP flood*, ktorý sa vyznačuje zaplavením náhodných portov väčším počtom paketov s UDP protokolom, *ICMP útoky*, ktoré fungujú na podobnom princípe ako UDP flood zasielaním paketov s ICMP protokolom so zámerom zahltiť prichádzajúcu aj odchádzajúcu prevádzku).
- *Protocol exploitation flooding attacks*: Účelom tohto typu útoku je vyčerpanie zdrojov zneužím implementačných chýb niektorých protokolov (napríklad TCP SYN-ACK flood, RST/FIN flood).



Obr. 3.3: TCP SYN flood útok[13]

- *Reflection-based flooding attacks*: Útočníci odosielaju sfalšované požiadavky Echo ICMP a následné odpovede spôsobia zahltenie zdrojov obete. Príkladom takéhoto útoku je Smurf attack.

Obr. 3.4: Smurf DDoS útok[22]

- *Amplification-based attacks*: Pri každej prijatej správe útočníci generujú väčšie množstvo správ na zosilnenie prenosu smerom k obeti útoku. Tento typ útoku je kombinovaný s *flooding* útokmi.

DDoS flooding attacks na aplikačnej vrstve zaberajú menšiu šírku sieťového pásma a sú účinné aj vďaka tomu, že okrem sieťových zdrojov využívajú zdroje servera. Tento typ útokov predstavuje z hľadiska ochrany výzvu, pretože žiadajú informácie o aplikácií a je náročné odlíšiť legitímne a škodlivé požiadavky pri sledovaní prevádzky na sieťovej vrstve. V dôsledku šifrovania webového prenosu pomocou SSL a HTTPS nedokáže ochrana proti DDoS útokom skúmať obsah samotného paketu.

Príkladom útoku na sieťovej vrstve je *ICMP flood*, ktorý využíva ICMP správy na preťaženie šírky pásma cieľovej siete.

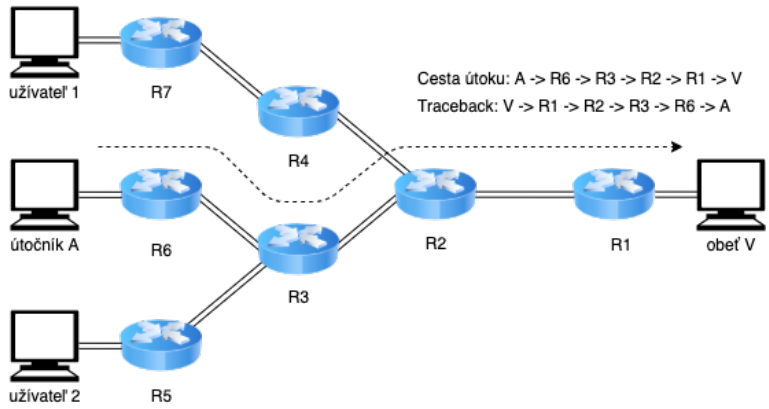
Linková vrstva ustanovuje, udržiava a rozhoduje o prenose dát. Príkladom útoku na tejto vrstve je *MAC flooding*, pri ktorom útočník odosieľa viac fiktívnych Ethernetových rámcov, každý s inou MAC adresou a tým sa snaží o vyčerpanie pamäte v prepínači [9].

3.1.1 Obranné mechanizmy

Ako je popísané v [15] detekcia DDoS útokov je obtiažna z toho dôvodu, že neexistujú žiadne spoločné charakteristiky, ktoré by bolo možné použiť na jednoznačnú identifikáciu DDoS útoku. Taktiež distribuovaná povaha týchto útokov spôsobuje problémy so spätným vystopovaním zdroja, nakoľko môžu útočníci použiť IP spoofing s cieľom zakryť svoju skutočnú identitu.

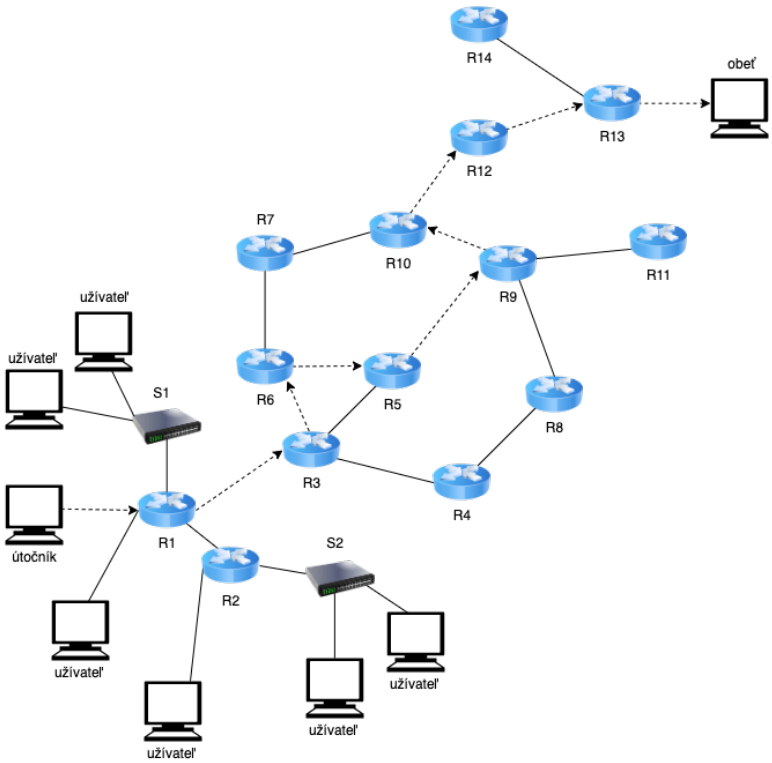
Mirkovic a kol. [25] vo svojej práci popisujú stratégie na obranu voči DDoS útokom. Pre ochranu voči útokom na L3/L4 vrstve patria filtrovanie a obmedzenie rýchlosti. Existuje niekoľko známych metód, ktoré môžu byť rozdelené do nasledujúcich kategórií:

- *IP Traceback* metódy - tieto metódy spočívajú v určení pôvodu paketu. Spätné sledovanie cesty k IP adrese je dôležité pre identifikáciu zdrojov údajov. Vrámci týchto metód boli navrhnuté rôzne princípy, medzi ktoré patrí pravdepodobnostné označovanie paketov (*probabilistically marking packets*) pri prechode smerovačmi v sieti, ktoré predstavil Savage a kol. [35] spočívajúce v označovaní paketov IP adresou alebo okrajmi cesty, ktorou paket prešiel, no obe tieto možnosti prinášajú značné nevýhody - je potrebný veľký počet paketov na určenie alebo detailnejšie informácie o stave.



Obr. 3.5: IP Traceback mechanizmus [33]

Deterministické označovanie paketov (*deterministic packet marking*), kde Belenky a kol. [3] predstavili princíp označovania prichádzajúcich paketov v mieste ich vstupu do siete vložením hornej alebo spodnej polovice IP adresy vstupného rozhrania do poľa ID fragmentu paketu a rezervným bitom nastaviť, ktorá časť adresy je vložená a týmto postupom znížili počet potrebných paketov na získanie správnej cesty. Medzi ďalšie metódy patria prístupy založené na smerovači (*Router marking* 3.6), mimopásmové prístupy a sledovanie stôp aktívnych tokov útokov.



Obr. 3.6: Router marking mechanizmus [33]

- Princíp *Management information base (MIB)*, ktorý spočíva vo vytváraní hierarchickej databázy entít v komunikačnej sieti. Každý záznam je identifikovaný pomocou identifikátora objektu (OID) a poskytuje informácie o paketoch a štatisky smerovania, ktoré môžu byť neskôr použité na zistenie anomálií[19].
- Filtrovacie mechanizmy predstavujú princíp ochrany pred DDoS útokmi, ktorý funguje na základe označenia legitímnych paketov. Toto označovanie paketov je možné použiť na filtrovanie prevádzky a riadenie (povolenie alebo zakázanie) prenosu údajov na základe IP adresy z ktorej prichádzajú dáta, IP adresy kam dáta smerujú alebo použitých aplikačných protokolov [8].

Kapitola 4

Návrh metrík

Metriky použité vrámci tejto práce boli navrhnuté so zámerom skúmať príznaky charakteristické pre odhalenie potencionálneho DDoS útoku. Hodnoty jednotlivých príznakov môžu byť použité na skúmanie správania jednotlivých prvkov siete a stanovanie hodnôt bežnej sieťovej prevádzky v rôznych časových obdobiach. Tieto metriky sú zamerané na skúmanie prevádzky najmä na sieťovej a transportnej vrstve a pri významných protokoloch ako sú HTTP a DNS aj na aplikačnej vrstve.

V nasledujúcej tabuľke 4.1 sú uvedené názvy metrík a ich popis. Položka *označenie metriky* predstavuje zároveň označenie vrámci implementačnej časti práce. Pokiaľ v názve nie je uvedené inak, jedná sa o komunikáciu v smere od odosielateľa k príjemcovi. Metriky boli čerpané z prác [21] a [23].

Tabuľka 4.1: Navrhnuté metriky

Označenie metriky	**Popis**
CNT_FLOWS	počet tokov
CNT_BYTES	celkový počet prenesených bajtov
CNT_BYTES_IN	počet prenesených bajtov od zroja k cieľu
CNT_BYTES_OUT	počet prenesených bajtov od cieľa k zdroju
CNT_PACKETS	celkový počet prenesených paketov
CNT_PKTS_IN	počet prenesených paketov od zroja k cieľu
CNT_PKTS_OUT	počet prenesených paketov od cieľa k zdroju
CNT_IPS	počet IP adries
CNT_PEERS	počet komunikujúcich zariadení
CNT_SRC_PORTS	počet zdrojových portov
CNT_DST_PORTS	počet cieľových portov
CNT_PROTO	počet protokolov
CNT_PKTS_UDP	počet paketov s protokolom UDP
CNT_PKTS_TCP	počet paketov s protokolom TCP
CNT_PKTS_ICMP	počet paketov s protokolom ICMP
CNT_PKTS_HTTP	počet paketov s protokolom HTTP
AVG_PKT_SIZE_TCP	priemerná veľkosť paketu s protokolom TCP
AVG_PKT_SIZE_UDP	priemerná veľkosť paketu s protokolom UDP
AVG_PKT_SIZE_ICMP	priemerná veľkosť paketu s protokolom ICMP
Pokračovanie na nasledujúcej strane	

Tabuľka 4.1 – pokračovanie z predchádzajúcej strany

Označenie metriky	**Popis**
AVG_PKT_SIZE_HTTP	priemerná veľkosť paketu s protokolom HTTP
CNT_SYN_ACK	počet paketov s TCP príznakmi SYN a ACK
CNT_RST	počet paketov s TCP príznakom RST
CNT_FIN_ACK	počet paketov s TCP príznakmi FIN a ACK
CNT_ACK_ANY	počet paketov s TCP príznakom ACK a akýmkoľvek iným TCP príznakom
CNT_FIN_ANY	počet paketov s TCP príznakom FIN a akýmkoľvek iným TCP príznakom
CNT_PSH_ANY	počet paketov s TCP príznakom PSH a akýmkoľvek iným TCP príznakom
CNT_BYTES_IP	počet bajtov na IP adresu
CNT_PKTS_IP	počet paketov na IP adresu
CNT_PROTO_IP	počet protokolov na IP adresu
CNT_SRC_PORTS_IP	počet zdrojových portov na IP adresu
CNT_DST_PORTS_IP	počet cieľových portov na IP adresu
CNT_BYTES_PROTO	počet bajtov na protokol
CNT_PKTS_PORT	počet paketov na port
CNT_PROTO_PORT	počet protokolov na port
CNT_PKTS_FLOW	počet paketov na tok
CNT_BYTES_FLOW	počet bajtov na tok
RAT_SD_PORTS	pomer zdrojových a cieľových portov
RAT_IO_BYTES	pomer počtu bajtov v smere od zdroja k cieľu
RAT_IO_PACKETS	pomer počtu paketov v smere od zdroja k cieľu
AVG_FLOW_DUR	priemerná dĺžka toku (v milisekundách)
AVG_BYTES_PS_IN	priemerný počet prichádzajúcich bajtov za sekundu
AVG_BYTES_PS_OUT	priemerný počet odchádzajúcich bajtov za sekundu
AVG_PKTS_PS_IN	priemerný počet prichádzajúcich paketov za sekundu
AVG_PKTS_PS_OUT	priemerný počet odchádzajúcich paketov za sekundu
CNT_HTTP_GET	Počet HTTP GET dotazov
CNT_HTTP_POST	Počet HTTP POST dotazov
DNS_PKTS_P_SEC	Počet paketov s DNS protokolom za sekundu

Kapitola 5

Návrh nástroja

Táto kapitola popisuje návrh nástroja učeného na získavanie informácií o sieťových tokoch a počítanie hodnôt vybraných vlastností určených pre profilovanie sietí so zámerom na mitigáciu DDoS útokov. Taktiež sú popísané vstupy a výstupy navrhnutého nástroja. Ďalej sa táto kapitola venuje aj spôsobu ukladania výstupných dát a použitým dátovým štruktúram pre ukladanie histórie o komunikujúcich entitách.

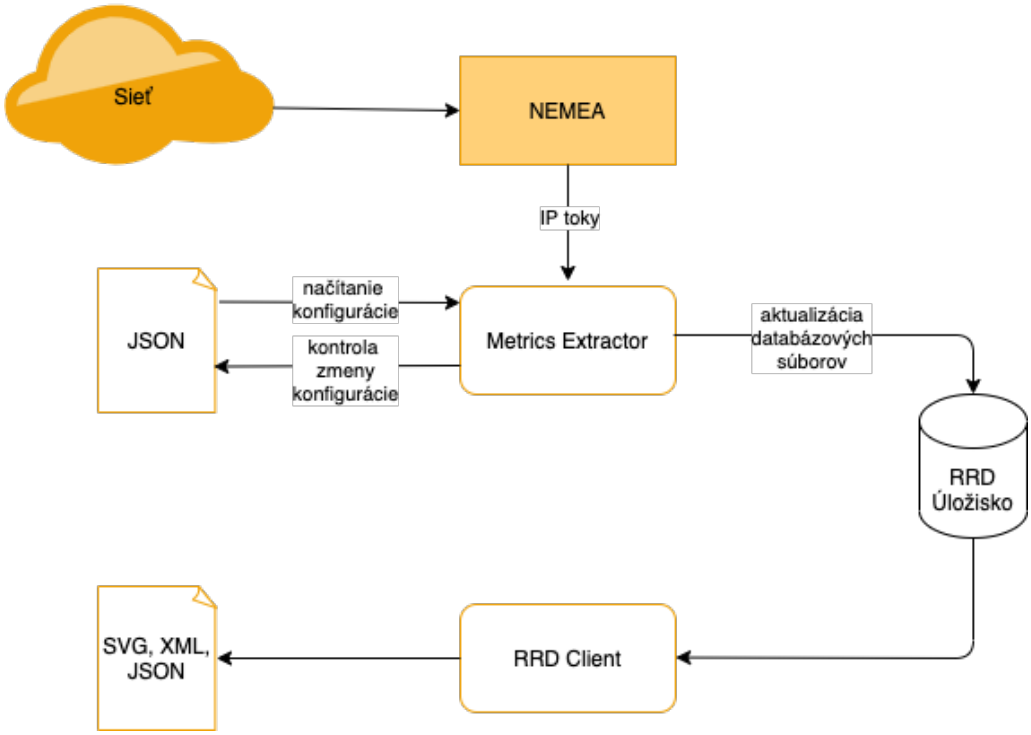
5.1 Architektúra nástroja

Navrhnutý modul sa skladá z 2 hlavných častí. Prvá časť nástroja predstavuje modul určený pre zber a spracovanie dát, výpočet hodnôt jednotlivých metrík pre nakonfigurované úrovne a vytváranie výstupných súborov. Druhý, menší modul, slúži na poskytnutie rozhrania k vytvoreným databázovým súborom a umožňuje jednoduchý prístup k uloženým dátam.

5.1.1 Modul *Metrics Extractor*

Modul, navrhnutý vrámci tejto práce, je navrhnutý ako samostatný modul systému NEMEA 2.5. Jednotlivé moduly vrámci tohto systému sú definované rozhraním a svojou funkcionalitou. Modul, navrhnutý vrámci tejto práce pracuje s jedným vstupným rozhraním a neodosiela žiadne informácie, takže nemá výstupné rozhranie. Hlavnou úlohou modulu je prijímať dáta na vstupnom rozhraní, agregovať ich a počítať hodnoty jednotlivých príznakov podľa vstupnej konfigurácie pre danú sieť. Na vstupe očakáva IP toky, ktoré následne spracováva a konfiguračný súbor s jednotlivými úrovňami detailu, ktoré užívateľ má záujem sledovať, popísaný v 6.2.2. Vstupné dáta môžu byť v rôznom formáte, napríklad sieťová prevádzka vo formáte *pcap* súboru či *UniRec* záznamy.

Zámerom tejto práce je zhromažďovať dáta o sieťovej prevádzke a taktiež ich uloženie pre neskoršiu analýzu. Dátové úložisko bolo vybrané so zámerom spotrebovať čo najmenšie množstvo pamäte a preto bol vybraný nástroj RRDtool, popísaný v 5.2. Výsledkom behu tohto modulu je niekoľko databázových súborov, ktoré reprezentujú správanie sieťových entít.



Obr. 5.1: Architektúra navrhnutého nástroja

5.1.2 Modul *RRD Client*

Tento modul je navrhnutý tak, aby poskytol užívateľom jednoduchý prístup k dátam uloženým vrámci databázového súboru, ktorý je výsledkom behu modulu *Metrics Extractor* 5.1.1. Úlohou tohto modulu je spracovať databázové súbory, vytvoriť grafy reprezentujúce správanie sledovanej siete a jej vybraných vlastností, výpočet štatistických údajov či vytvorenie inej reprezentácie dát ľahko čitateľných pre užívateľa (napríklad *JSON* a *XML*).

5.2 Úložisko dát

NEMEA je navrhnutá tak, že dáta sú spracovávané a analyzované priebežne v pamäti s minimálnou réžiou pre uchovanie dát. Pre zachovanie konceptu tohoto princípu bolo potrebné vybrať vhodné dátové úložisko.

Rozsiahlejšie monitorovacie systémy vyžadujú efektívne úložisko a konsolidáciu nameraných dát. Pre rozsiahle siete môžu namerané dáta a následne vyrátané hodnoty vybraných metrík predstavovať komplikácie spojené s réžiou na úložisko. V minulosti sa najrozšírenejším typom databáz javili relačné databázy, no dnes existujú technológie optimalizované na spracovanie konkrétnych typov údajov, ktoré je možné využívať na bežne dostupnom hardvéri.

Vzhľadom na objem, charakter dát a fakt, že je potreba tieto dáta udržiavať v dlhých časových intervaloch je možným kandidátom databáza časovo závislých radov. Databázy časových radov sú optimalizované na ukladanie časových radov pomocou dvojíc času a hodnoty. V posledných rokoch sa ponuka týchto technológií značne rozšírila. Na základe porovnania databáz časovo závislých radov [2], bolo nájdených 50 Open Source systémov a 33 proprietárnych, medzi ktoré patria systémy so závislosťou na systéme správy data-

báz, bez závislosti a proprietárne systémy. Medzi najčastejšie vyhľadávané patria napríklad InfluxDB, RRDtool, Druid, Prometheus, a mnohé ďalšie.

Existuje množstvo spôsobov ako ukladať informácie identifikujúce sieťové entity, medzi ktoré patrí najmä IP adresa. Pre príklad, môžu byť použité relačné databázy, no pre naše účely by bol tento prístup nevhodný. Požadovanú funkcionalitu je možné dosiahnuť použitím pravdepodobnostných dátových štruktúr, ako je napríklad Bloomov filter. Výmenou za dosiahnutie úplnej istoty sa tieto štruktúry javia rýchle, priestorovo efektívne a dokážu odpovedať na množinu členských dotazov v konštantnom čase.

5.3 RRDtool databáza

Round Robin Database Tool (RRDtool) je vysoko výkonný, Open Source systém pre zaznamenávanie a vytváranie grafov pre údaje časových rád[30]. Tento systém bol prvýkrát predstavený Tobiasom Oetikerom v roku 1999 [29], kde RRDtool popísal ako skratku pre Round Robin Database tool. Jednou z výhod tohto systému je, že údaje ukladá veľmi kompaktným spôsobom, ktorý sa v priebehu času nebude rozširovať a na základe týchto údajov umožňuje užívateľovi vytvárať grafy. Stabilná verzia bola vydaná 2019-5-27 a je dostupná pod licenciou GNU General Public License (GNU GPL). Tento druh databázy využívajú rôzne softvéry pre monitorovanie sietí, systému aplikácií ako je napríklad Munin [17], OpenNMS [37], Nagios [26] a mnohé ďalšie.

RRDtool je nová generácia úzko zameraného nástroja pre zber a vykreslenie dát Multi Router Traffic Grapher (MRTG), oproti ktorému RRDtool predstavuje komplexné programovacie prostredie pre zber dát a vytváranie grafov.

Funkcie pre prácu s týmito databázovými súbormi poskytuje nástroj `rrdtool`. Tento nástroj umožňuje vytvárať databázy, upravovať a aktualizovať dáta, vytvárať grafy a zálohy. Štruktúra databázových súborov typu RRDtool je odlišná od iných lineárnych databáz. Na rozdiel napríklad od relačných databáz, nie sú tabuľky definované stĺpcami a prepojeniami medzi jednotlivými tabuľkami. Databázy RRDtool majú pomerne jednoduchú štruktúru, parametre, ktoré je potrebné definovať sú premenné, ktoré obsahujú hodnoty archívov hodnôt. Nakoľko hlavnou úlohou týchto databáz je ukladať asociatívne dvojice času a hodnôt, je nutné taktiež definovať niekoľko parametrov súvisiacich s časom. Kvôli svojej štruktúre obsahuje definícia databázy aj určenie akcií, ktoré sa majú vykonať v prípade chýbajúcich hodnôt pri aktualizácií databázy.

Dôležitými výrazmi spojenými s databázami RRDtool sú zdroj údajov DS, typ zdroja údajov DST, Round Robin archív RRA a konsolidačná funkcia CF. Základy práce s RRDtool sú prevzaté z manuálu [32].

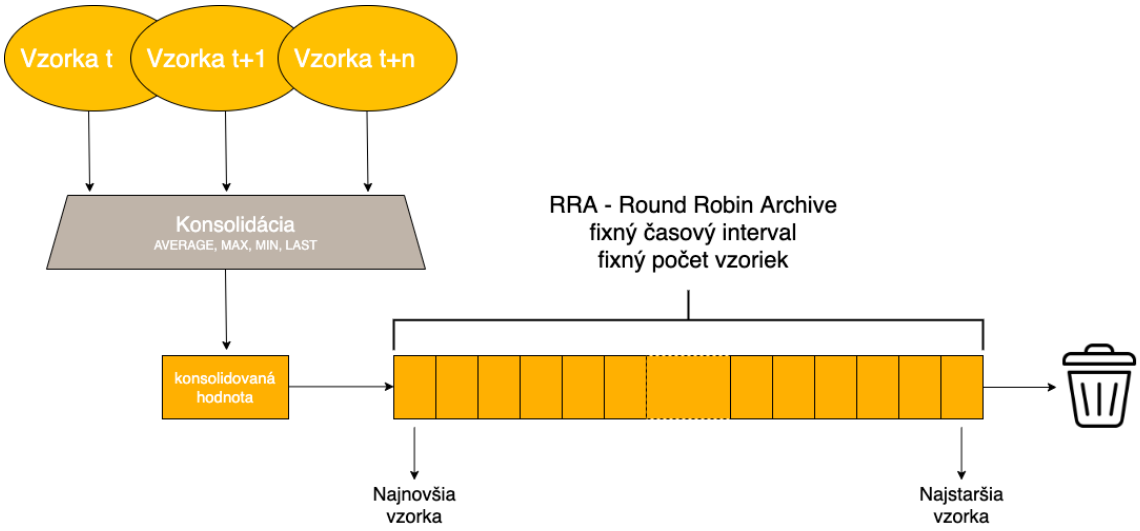
5.3.1 Zber dát

Pri sledovaní stavu systému je potrebné mať dáta k dispozícií v ľubovoľnom časovom intervale. Bohužiaľ, môžu nastať prípady, kedy nie je možné načítať dáta v požadovaný časový interval, preto RRDtool umožňuje aktualizáciu databázy v ľubovoľný okamih. Hodnota zdroja údajov je automaticky interpolovaná v poslednom časovom intervale a následne zapísaná do databázy. Originálna hodnota je taktiež uložená a zohľadnená pri interpolácií nasledujúcej položky. Zdroj údajov DS predstavuje premennú, ktorá sa vzťahuje na parameter sledovanej hodnoty. Vrámci databázy je možné vytvoriť viacero zdrojov údajov a po každom intervale kroku je aktualizovaná hodnota vrámci jednotlivých zdrojov údajov. Typ údajov uložených v DS je špecifikovaný pomocou typu DST a môže to byť jeden z `COUNTER,`

`DERIVE`, `ABSOLUTE`, `GAUGE`. Výber konkrétnych typov pre účely tejto práce bude popísaný vrámci nasledujúcej kapitoly.

5.3.2 Round Robin archívy

Samotné databázové súbory sa skladajú z archívov typu Round Robin, ktoré obsahujú konsolidované hodnoty údajov. Jedná sa o veľmi efektívny spôsob ukladania údajov za určitý čas pri použití známeho množstva úložného priestoru. Vrámci jednej databázy je možné špecifikovať niekoľko RRA archívov a podľa nastavenia jednotlivých archívov je možné ukladať dáta v rôznych časových rozmedziach, no je nutné dopredu zadefinovať fixný počet vzoriek v danom intervale. Použitie týchto archívov zaručuje, že RRA časom nerastie a staré dáta sú automaticky premazávané. Konsolidačné funkcie CF umožňujú uchovávať údaje veľmi dlho a postupne znižovať rozlíšenie údajov pozdĺž časovej osi a taktiež to umožňuje ukladať istý typ dát, napríklad maximálny počet paketov.



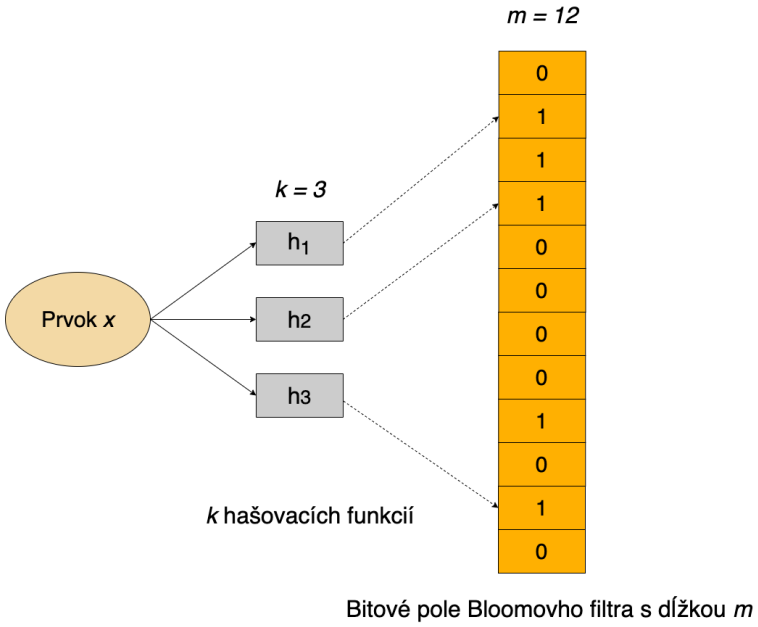
Obr. 5.2: Ukladanie údajov do RRA archívov

Na obrázku 5.2 je ilustrovaný spôsob spracovania vzoriek dát konsolidačnými funkciami, vloženie novej hodnoty do RRA archívu a proces premazávania uložených vzoriek.

5.4 Bloomov filter

Bloomov filter je priestorovo efektívna pravdepodobnostná dátová štruktúra, ktorá bola predstavená Burtonom H. Bloomom v [4]. Táto dátová štruktúra poskytuje bežné operácie s množinami, vkladanie a dotazy na členstvo prvku, pričom odpovedá s konštantnou časovou zložitosťou. Nevýhodou tejto štruktúry je skutočnosť, že výsledkom dotazu na členstvo prvku je buď to, že prvok sa môže nachádzať alebo to, že určite nie je v sade. Vzhľadom na to, že táto štruktúra je pravdepodobnostná, môže nastať chyba pri overovaní príslušnosti prvku v sade. Pri tejto chybe sa o prvku, ktorý do sady nepatrí dozvieme, že do sady patrí, no nikdy nie naopak. Prázdny Bloomov filter predstavuje bitové pole s dĺžkou m bitov, pričom všetky hodnoty sú nastavené na 0. Pre prácu s Bloomovým filtrom je potreba definovať k rôznych hašovacích funkcií, pričom každá z nich mapuje prvok na jeden bit v poli. Vloženie prvku x do množiny spočíva vo vypočítaní hodnoty každej z k hašovacích funkcií,

dostaneme hodnoty $h_1(x), h_2(x), ...h_k(x)$ a hodnoty poľa na týchto hodnotách nastavíme na 1. Dotaz na príslušnosť prvku prebieha podobne. Opäť sa vypočítajú hodnoty každej z k hašovacích funkcií a dostaneme hodnoty $h_1(x), h_2(x), ...h_k(x)$. Skontrolujeme hodnoty zapísané na týchto indexoch v poli a pokiaľ je aspoň jedna z týchto hodnôt rovná 0, je možné s určitosťou rozhodnúť, že prvok x sa v množine prvkov nenachádza. V prípade, že sú tieto hodnoty rovné 1, je relatívne veľká pravdepodobnosť, že sa prvok v množine nachádza.



Obr. 5.3: Vloženie prvku do Bloomovho filtra

Kapitola 6

Implementácia

V tejto kapitole je popísaná implementácia jednotlivých častí navrhnutého systému. Vzhľadom na to, že výsledný systém je integrovaný do väčšieho monitorovacieho systému, NEMEA 2.5, a využíva niektoré jeho dôležité časti, je popísané zapojenie do tohoto systému. Ďalej je opísaný formát konfiguračných súborov a možnosti nastavenia programu.

6.1 Vývojové prostredie

Programová časť tejto práce bola implementovaná pod operačným systémom Linux Ubuntu 18.04.5. Boli použité programovacie jazyky C++ a Python 3.6. Pre zálohovacie účely bol použitý systém GIT[1].

6.2 Extrakcia metrík

Hlavnou úlohou tejto práce je zbierať dáta, spracovávať a počítať štatistiky pre jednotlivé nakonfigurované úrovne detailu, ktoré má užívateľ záujem sledovať. Tento modul je implementovaný ako súčasť systému NEMEA, využíva platformu TRAP, opísanú v 2 a vstupné dáta očakáva vo formáte UniRec 2. Na základe polí tohto formátu sú hodnoty ďalej spracovávané, počítané rozličné štatistiky a vytvárané profily pre rôzne úrovne.

Všetky hodnoty UniRec polí, opísané v 6.1 je možné získať napríklad prepojením s modulom `ipfixprobe`[2].

[1] Distribuovaný systém riadenia revízií. http://git-scm.com/.

[2] https://github.com/CESNET/ipfixprobe

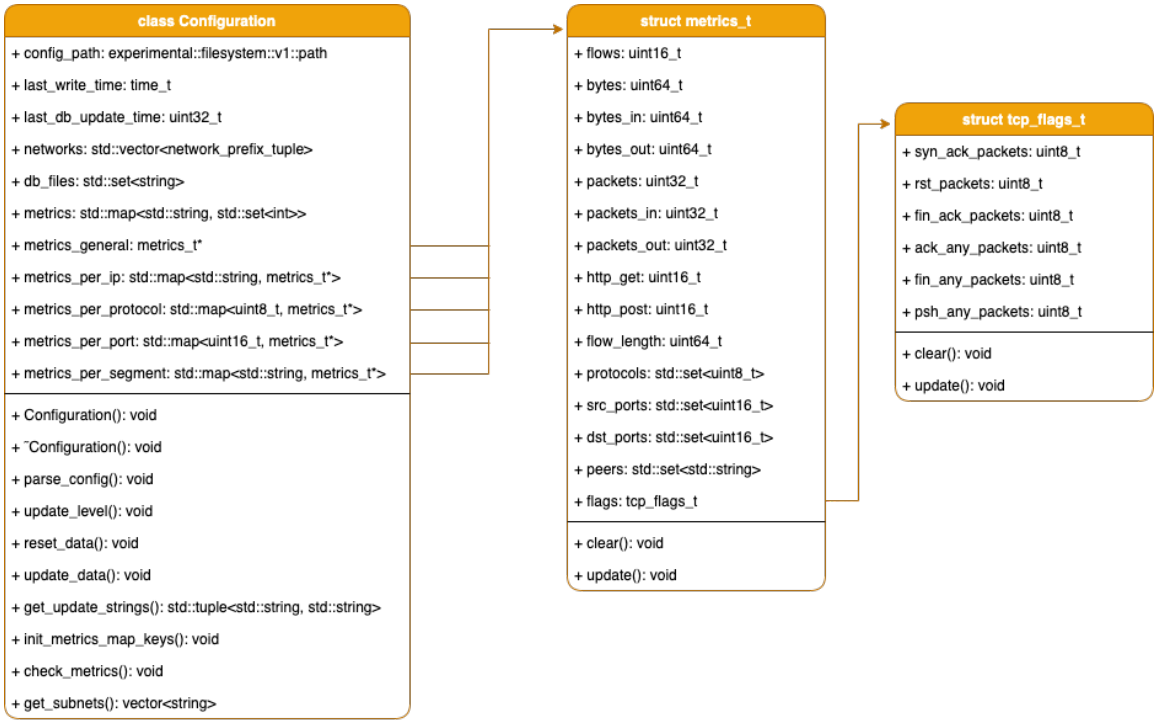
UniRec pole	Dátový typ	Popis
SRC_IP	ipaddr	zdrojová IP adresa
DST_IP	ipaddr	cieľová IP adresa
SRC_PORT	uint16	zdrojový port
DST_PORT	uint16	cieľový port
PROTOCOL	uint8	protokol
TCP_FLAGS	uint8	TCP príznaky
BYTES	uint32	počet bajtov (smer od zdroja k cieľu)
BYTES_REV	uint32	počet bajtov (smer od cieľa k zdroju)
PACKETS	uint32	počet paketov (smer od zdroja k cieľu)
PACKETS_REV	uint32	počet paketov (smer od cieľa k zdroju)
TIME_FIRST	time	prvá časová pečiatka
TIME_LAST	time	posledná časová pečiatka
TIME_LAST	time	posledná časová pečiatka
HTTP_REQUEST_METHOD	string	metóda HTTP požiadavku [1]

[1]Nutné zapojenie HTTP pluginu modulu `ipfixprobe`

Tabuľka 6.1: Použité UniRec polia

6.2.1 Trieda `Configuration` a pomocné štruktúry

Trieda `Configuration` slúži na načítanie konfigurácie zo súboru, uloženie metrík pre jednotlivé úrovne detailu a hodnôt medzi aktualizáciami databázy. Niektoré atribúty sa skladajú z ďalších štruktúr, ako je zobrazené na obrázku 6.1.
Štruktúra `metrics_t` slúži na ukladanie hodnôt pre rôzne úrovne detailu získaných zo vstupného rozhrania a na základe týchto hodnôt sú ďalej počítané štatistiky. Dáta zo vstupného rozhrania sa spracovávajú v cykle a pri každom príchode dát sa aktualizujú dáta uložené v triede `Configuration` pre každú nakonfigurovanú úroveň. Po každej aktualizácií databázy sú uložené dáta vymazané pomocou funkcie `clear()`.



Obr. 6.1: Diagram tried a dátových štruktúr

Trieda `Configuration` obsahuje nasledujúce atribúty:

- `config_path` označuje cestu ku konfiguračnému súboru. Formát a položky konfiguračného súboru sú popísané v nasledujúcej podkapitole 6.2.2.
- `last_write_time` čas poslednej zmeny konfiguračného súboru. Počas behu programu sa kontroluje, kedy bol konfiguračný súbor uložený v `config_path` naposledy upravený a v prípade zmeny konfigurácie sú upravené ďalšie atribúty tejto triedy, modifikované sledované vlastnosti niektorej z nakonfigurovaných úrovni alebo pridané nové úrovne.
- `networks` obsahuje zoznam nakonfigurovaných sietí.
- `db_files` obsahuje zoznam mien databázových súborov.
- `metrics` kľúčom v tejto dátovej štruktúre je reťazec popisujúci úroveň detailu a dáta prislúchajúce tomu kľúču predstavujú množinu nakonfigurovaných metrík.
- `metrics_general` štruktúra uchovávajúca dáta pre prevádzku na sieti ako celok, bez konfigurácie urovne detailu.
- `metrics_per_ip` dátová štruktúra uchovávajúca dáta pre úroveň IP adries.
- `metrics_per_port` dátová štruktúra uchovávajúca dáta pre úroveň portov.
- `metrics_per_protocol` dátová štruktúra uchovávajúca dáta pre úroveň protokolov.
- `metrics_per_segment` dátová štruktúra uchovávajúca dáta pre úroveň podsietí.

a metódy:

- `parse_config`, ktorá spracuje vstupný konfiguračný súbor.
- `update_level`, volá sa pre každú nakonfigurovanú úroveň, spracuje dáta a aktualizuje databázu.
- `reset_data` vymaže dáta pre každú položku dátových štruktúr nakonfigurovaných úrovní.
- `update_data` získa potrebné informácie o toku, spracuje ich a uloží do príslušných štruktúr.
- `get_update_strings`, ktorá vytvorí reťazce so syntaxou pre funkciu `update`[31] nástroja `rrdtool`.
- `init_metrics_map_keys` nainicializuje hodnoty štruktúr pre niektoré nakonfigurované úrovne v závislosti od metrík špecifikovaných v konfiguračnom súbore.
- `check_metrics` skontroluje, či boli v konfiguračnom súbore špecifikované metriky. V prípade, že táto konfigurácia chýba, sú použité prednastavené hodnoty.

6.2.2 Formát konfiguračného súboru

Táto práca podporuje sledovanie rôznych úrovní detailu - IP adresa, podsieť, port, protokol vrámci vybranej sieť. Vybrané metriky popísané v kapitole 4 sú vybrané s účelom odhaliť potencionálny DDoS útok. Vstupný konfiguračný súbor očakáva trieda `Configuration`, opísaná v 6.2.1, vo formáte `JSON` s kľúčovými slovami uvedenými v 6.1. Popis jednotlivých sekcií vrámci konfiguračného súboru je popísaný pod uvedeným príkladom.

```
[
    {
        "network": ["0.0.0.0/0", ...]
    },
    {
        "level": "general",
        "metrics": [
            "CNT_FLOWS",
            "CNT_BYTES",
            "CNT_PACKETS",
            "CNT_PROTO",
            "CNT_FIN_ACK",
            ...
        ]
    },
    {
        "level": "ip",
        "keys": ["10.42.0.99", ...],
        "metrics": [
            "AVG_PKT_SIZE_TCP",
            "CNT_FIN_ACK",
            ...
        ]
    },
    {
```

```
26          "level": "port",
27          "keys": ["443", "80", ...],
28      },
29      {
30          "level": "protocol",
31          "keys": ["9", "6", ...],
32          "metrics": [
33              "CNT_FLOWS",
34              "CNT_BYTES",
35              "CNT_PACKETS",
36              ...
37          ]
38      },
39      {
40          "level": "subnet",
41          "keys": ["0.0.0.0/8", ...],
42          "metrics": [
43              "CNT_FLOWS",
44              "CNT_BYTES",
45              "CNT_PACKETS",
46              ...
47          ]
48      }
49  ]
```

Výpis kódu 6.1: Príklad konfiguračného súboru

Sekcie konfiguračného súboru:

- `network`: kľúčové slovo pre nastavenie sietí s prefixom, ktoré majú byť monitorované. V prípade, že táto položka chýba, program je nastavený na sieť `0.0.0.0/0`.
- `level`: kľúčové slovo označujúce požadovanú úroveň detailu. Očakávané hodnoty tejto sekcie sú `general`, `ip` pre výber sledovaných IP adries, `port` portov, `protocol` protokolov a `subnet` podsietí.
- `keys`: kľúčové slovo pre špecifikovanie vybraných hodnôt pre jednotlivé úrovne.
- `metrics`: kľúčové slovo pre špecifikovanie vybraných metrík. Pokiaľ toto pole nie je nakonfigurované, program pracuje s metrikami `CNT_PACKETS`, `CNT_BYTES` a `CNT_FLOWS`.

6.3 Aplikácia Bloomovho filtra

Úlohou tejto práce je sledovať sieťovú prevádzku vrámci nakonfigurovanej siete (prípadne sietí). Pre urýchlenie spracovania dotazu na príslušnosť IP adresy toku je použitý Bloomov filter popísaný v 5.4. Pre tieto účely bola vybraná knižnica *libbloom*[38], nakoľko je už použitá vrámci modulu *Bloom History*[23] a obsahuje implementáciu vylepšenia rýchlosti.

6.4 Vytvorenie databázy

Databáza je vytvorená pomocou funkcie `create` nástroja `rrdtool`. Vrámci tejto práce sa vytvára niekoľko databázových súborov, pre každú nakonfigurovanú úroveň - IP adresu, podsieť, každý port a každý protokol je vytvorený samostatný databázový súbor. Pôvodným zámerom bolo ukladať informácie o portoch a protokoloch vrámci jedného databázového

súboru, no po analýze časovej náročnosti spracovania tohto návrhu, bolo zvolené, že dáta pre každú nakonfigurovanú úroveň budú uložené v samostatnom súbore.

Pri vytváraní databázových súborov je nutné špecifikovať niekoľko parametrov pre jej správne fungovanie. Ako je popísané v kapitole 5.3, je nutné nastaviť jeden alebo viac zdrojov údajov (DS), ich dátový typ (DST) a jeden alebo viac Round Robin archívov.

Okrem už písaných parametrov je pri nastavení zdroja údajov je nutné špecifikovať aj nasledujúce parametre:

```
Data Source:DS-Name:DST:HeartBeat:Min:Max
```

- `DS-Name` definuje meno zdroja údajov.
- `HeartBeat` definuje maximálny počet sekúnd, ktorý môže uplynúť pred tým, ako sa bude hodnota daného DS považovať za neznámu.
- `Min` Minimálna prijateľná hodnota. Hodnoty menšie ako toto číslo sa považujú za neznáme.
- `Max` Maximálna prijateľná hodnota. Hodnoty presahujúce toto číslo sa považujú za neznáme.

Ako názov zdroja údajov je nastavené meno nakonfigurovanej metriky, podľa tabuľky4.1. Názvy metrík boli prispôsobené požiadavkám nástroja RRDtool, nakoľko dĺžka názvu je obmedzená na 19 znakov. Dátový typ hodnôt zdrojov údajov je nastavený na „`GAUGE`“. Tento dátový typ ukladá skutočnú hodnotu, neuskutočňuje žiadne prepočty hodnôt a umožňuje ukladanie desatinných čísel. Hodnoty pre parametre `Min` a `Max` sú voliteľné, takže sú nastavené na „U“ (reprezentujúce „Unknown“).

Vrámci tejto práce je v databázových súboroch vytvorených desať Round Robin archívov. Každý z týchto archívov ukladá dáta v rôznych časových intervaloch. Nastavenie parametrov archívov je popísané v tabuľke 6.2.

Syntax a význam parametrov pre vytvorenie archívu sú nasledovné:

```
Round Robin Archives: RRA:CF:XFF:Steps:Rows
```

- `RRA` kľúčové slovo, ktoré definuje Round Robin archív.
- `CF` predstavuje konsolidačnú funkciu. Môže mať hodnotu „`AVERAGE`“, „`MIN`“, „`MAX`“, „`LAST`“. Pre účely tejto práce bola vybraná „`AVERAGE`“.
- `XFF` definuje časť konsolidačného intervalu, ktorá môže byť tvorená z neznámych hodnôt, tak aby výsledná konsolidovaná hodnota bola považovaná za známu. Udáva sa vo forme pomeru povolených neznámych hodnôt k počtu hodnôt v intervale a pohybuje sa v intervale `<0, 1>`.
- `Steps` popisuje koľko primárnych hodnôt sa použije na vytvorenie konsolidovanej hodnoty použitej vo výslednom archíve.
- `Rows` definuje, koľko generácií údajov sa uchováva v RRA archíve.

Hodnota `steps`	Rozlíšenie	Interval uložených dát
1	5s	minúta
3	15s	30 minút
6	30s	1 hodina
12	1min	6 hodín
24	2min	12 hodín
36	3min	24 hodín
72	6min	7 dní
120	10min	30 dní
180	15min	6 mesiacov (180 dní)
240	20min	1 rok (365 dní)

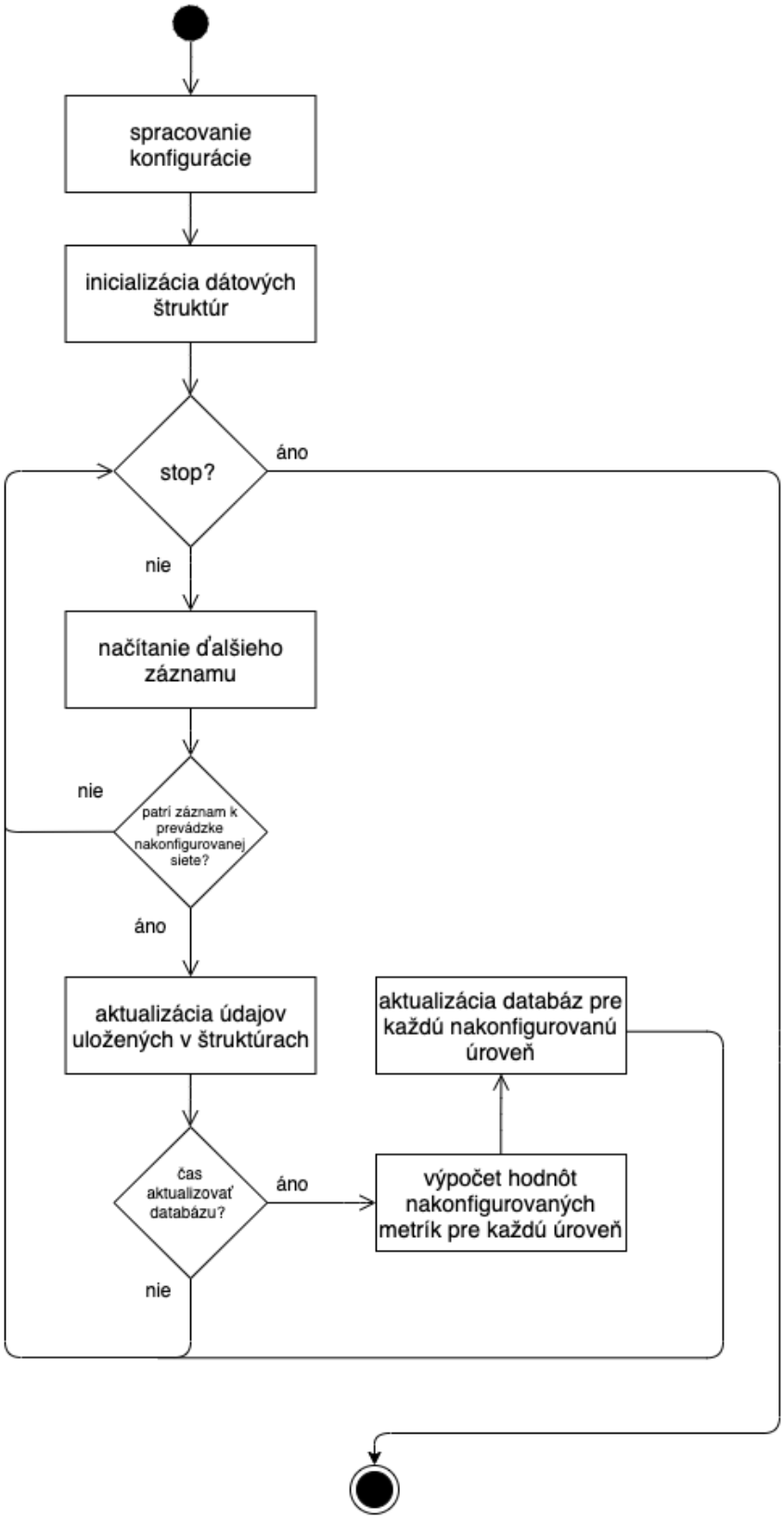
Tabuľka 6.2: Nastavenie parametrov Round Robin archívov

Po vytvorení databázy, majú jednotlivé súbory konečnú, plnú veľkosť a dáta sú predvyplnené hodnotou „UNKNOWN“. Veľkosť databázového súboru záleží od počtu zdrojov údajov (DS), počtu Round Robin archívov (v našom prípade je ich 10, ako je popísané v tabuľke 6.2), počtu dátových bodov uložených v každom RRA archíve a priestoru pre uloženie informácií v záhlaví.

Veľkosť databázových súborov sa môže meniť v prípade zmene konfigurácie a to pri pridaní sledovaných úrovni alebo metrík.

6.5 Popis behu programu

Princíp behu nástroja vytvoreného vrámci tejto práce je ilustrovaný na obrázku 6.2. Program pri svojom spustení očakáva na vstupe konfiguračný súbor, so syntaxou popísanou v podkapitole 6.2.2. Tento súbor a jeho jednotlivé sekcie sú spracované pomocou knižnice Jansson [24]. Do inštancie triedy `Configuration` sa uloží cesta ku konfiguračnému súboru aby bolo možné detekovať zmeny konfigurácie a taktiež sú uložené aj metriky pre jednotlivé úrovne detailu. Dáta zo vstupného rozhrania tohto modulu sú spracovávané v cykle a toky spadajúce do nakonfigurovanej siete (prípadne sietí) sú ďalej spracovávané. Dáta sú medzi aktualizáciami databázy uložené v triede `Configuration`. Na základe konfigurácie sú dáta agregované, rátané štatistiky a pripravené dáta pre aktualizáciu databázy príslušnej úrovne z konfiguračného súboru. Databázové súbory sú aktualizované pri každom skončení toku, no minimálny čas medzi dvoma aktualizáciami je v počiatočnom nastavení každú sekundu a po každej aktualizácií sú uložené dáta vymazané. Vytvorené databázové súbory predstavujú výstup modulu. Databázových súborov je vytvorených niekoľko, v závislosti od konfigurácie. V hociktorom momente behu programu užívateľ môže skúmať dáta uložené v databázových súboroch pomocou modulu popísaného v kapitole 6.7.



Obr. 6.2: Popis behu modulu *Metrics Extractor*

6.6 Výstup programu

Výstupom programu sú databázové súbory RRDtool databázy, popísané v podkapitole 5.3. Nakoľko tento typ databázy nepodporuje dotazy jazyka SQL, bol ako súčasť tejto práce vytvorený modul `rrd_client` v jazyku Python, ktorý je opísaný v nasledujúcej podkapitole 6.7 a poskytuje prístup k dátam z databázových súborov, ktoré sú výsledkom tejto práce.

6.7 Spracovanie výstupných dát

Výstupné dáta zapísané v RRDtool databázových súboroch je možné pomocou RRDtool funkcie `dump` previesť do formátu XML, no nakoľko tento formát nie je pre užívateľa dostatočne čitateľný, bol vytvorený modul `rrd_client`, implementovaný v jazyku Python, ktorý obsahuje triedu `RRD_client`. V tomto module bola použitá knižnica jazyka Python `rrdtool`[3].

Obr. 6.3: Diagram triedy `RRD_client`

Tento modul je možné nainštalovať a použiť v interaktívnom režime interpretu Python alebo spustiť z terminálu. Poskytuje rôzne funkcie na spracovanie a vizualizáciu databázových súborov, napríklad:

- vytváranie grafov, pomocou funkcie `graph`. Táto funkcia umožňuje vytvoriť graf pre ľubovoľné zdroje údajov (reprezentujú metriky) v rôznych časových intervaloch. V

[3] https://pypi.org/project/rrdtool/

počiatočnom nastavení sa vytvorí graf pre poslednú hodinu, deň, týždeň, mesiac a rok.

- výpis uložených metrík, pomocou funkcie `data_sources`.
- výpis hodnôt metrík, vybranej metriky, ktoré zabezpečujú funkcie `values` a `values_of`.
- rôzne štatistické funkcie ako minimálna hodnota, maximálna, priemerná, súčet, medián a ďalšie.
- uloženie údajov z databázy vo formáte JSON, XML.

Kapitola 7

Experimenty a testovanie

Táto kapitola obsahuje popis vykonaných experimentov a ukážky ich výstupov s cieľom overenia funkčnosti navrhnutého nástroja.

V časti 7.1 je popísaný spôsob získavania a preposielania dát do implementovaného modulu, ktoré tvorili základ testovania pri implementácií ale aj záverečných experimentoch. V nasledujúcej časti 7.2 je popísaný uskutočnený experiment s TCP SYN útokom. Ďalej sú v tejto časti prezentované výstupy nástroja a výsledky experimentu.

7.1 Získavanie dát

Na testovanie nástroja boli použité dáta z rôznych zdrojov. Najväčším zdrojom dát bolo odchytávanie sieťovej prevádzky na lokálnom stroji, na ktorom bol modul zároveň vyvíjaný pomocou modulu *ipfixprobe*, ktorý je súčasťou systému NEMEA. Dáta z tohto modulu boli exportované pomocou Unixového socketu na vstupné rozhranie navrhnutého modulu.

7.2 Popis experimentu

Pre účely otestovania funkčnosti navrhnutej metódy bol vykonaný experiment, na ktorom je zachytená normálna sieťová prevádzka na lokálnom zariadení a umelo vygenerovaný TCP SYN flood útok. Dáta pre tento experiment boli získavané spôsobom popísaným v podkapitole 7.1. Zároveň bola táto prevádzka zachytávaná nástrojom Wireshark [40], pre neskoršiu analýzu a preverenie v tomto nástroji. Pre zachytenie abnormálneho správania, vykazujúceho známky DDoS útoku, bol využitý nástroj hping3 [34]. Pomocou tohto nástroja boli generované pakety s protokolom TCP.

```
1 sudo hping3 -c 10000 -d 120 -S -w 64 -p 80 --flood --rand-source 10.42.0.140
```

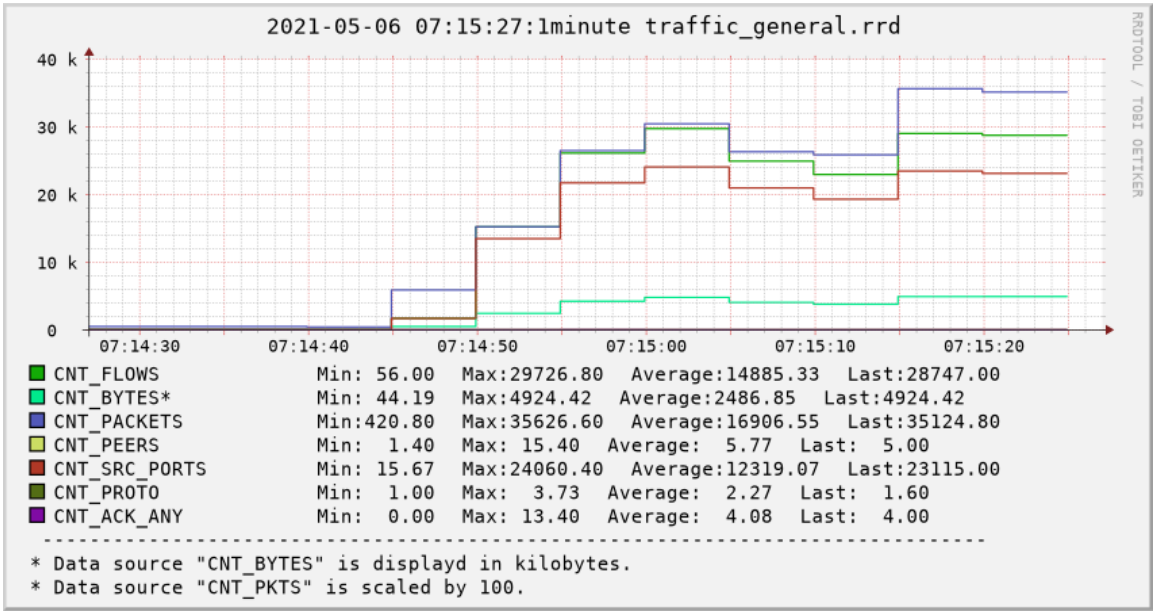
Výpis kódu 7.1: Príkaz nástroja hping3 na vygenerovanie SYN flood útoku

Pomocou príkazu 7.1 bolo odoslaných 10000 paketov, každý s veľkosťou 120 bajtov, nastaveným TCP SYN príznakom a sfalšovanými IP adresami na cieľ útoku špecifikovaného IP adresou 10.42.0.140.

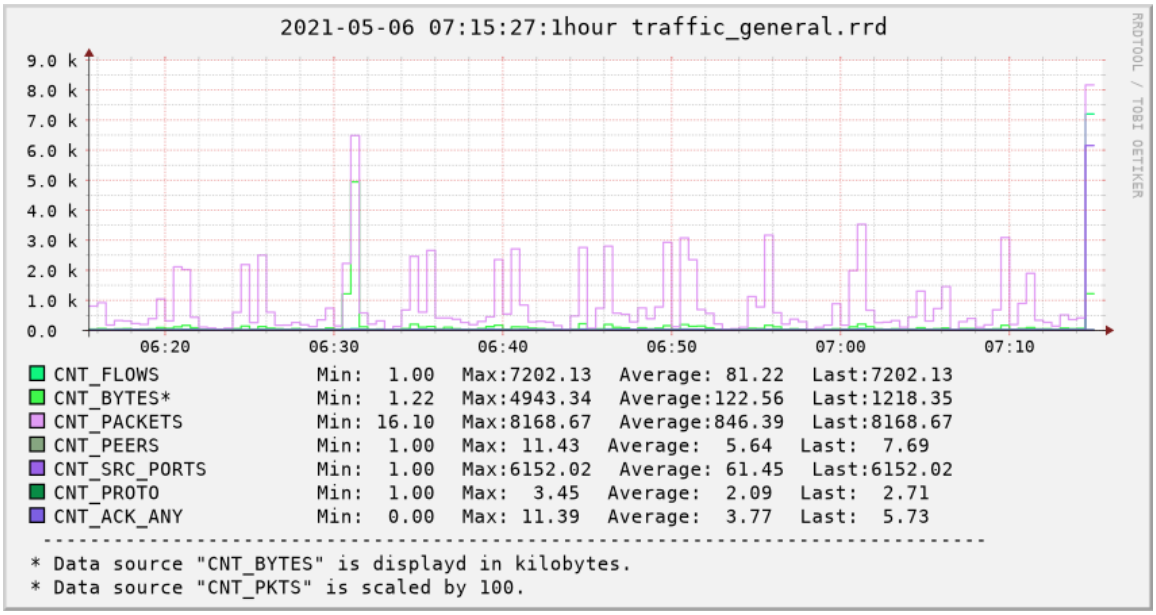
Konfiguračný súbor, použitý pre tento experiment je obsiahnutý v prílohe B.1 tejto práce.

7.3 Výstup experimentu

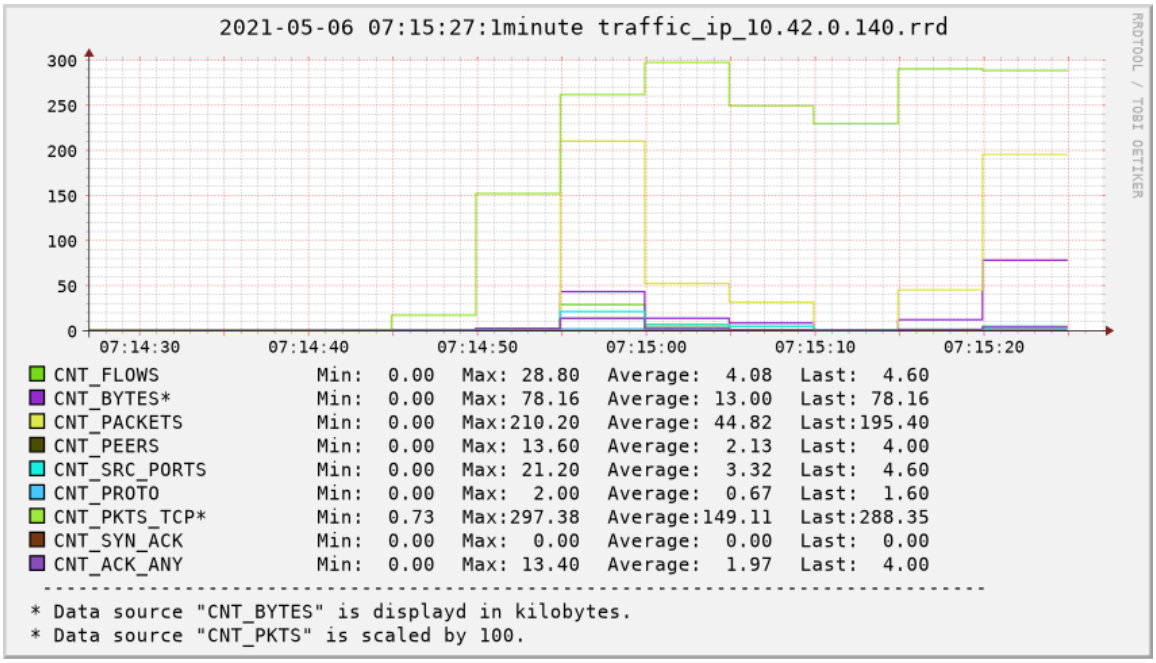
Príkazom 7.1 bolo generované veľké množstvo TCP paketov na IP adresu 10.42.0.140 dňa 6.5.2021 okolo 7:14h.



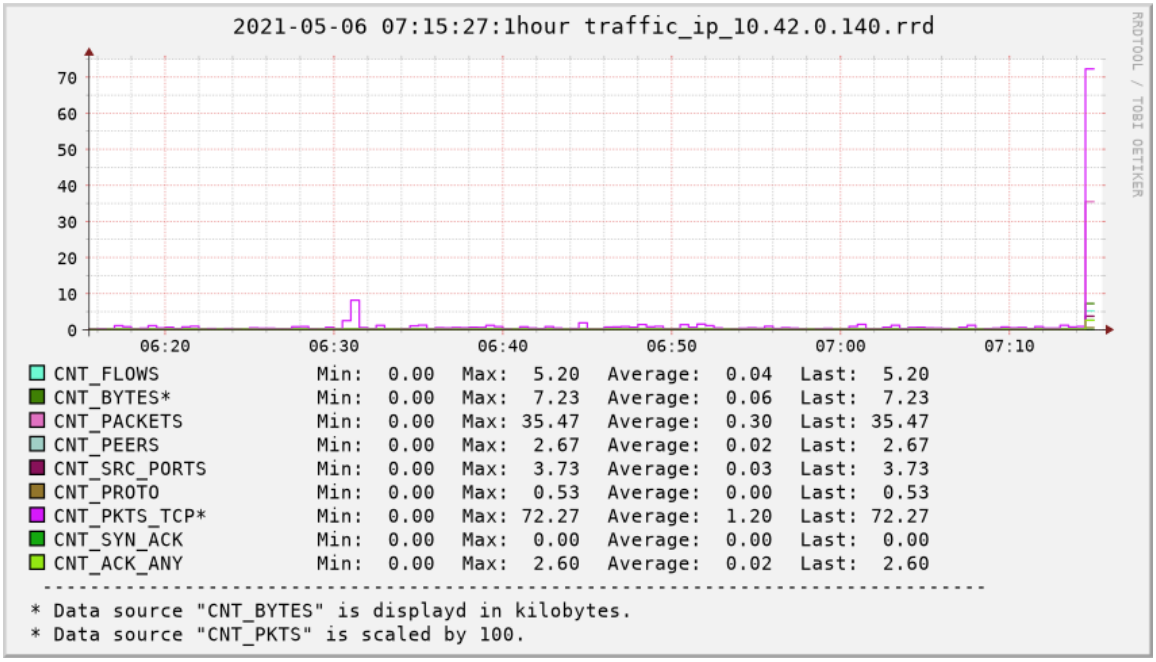
Obr. 7.1: Hodnoty vybraných metrík zachytených za 1 minútu



Obr. 7.2: Hodnoty vybraných metrík zachytených za 1 hodinu



Obr. 7.3: Hodnoty vybraných metrík zachytených za 1 minútu na IP adrese 10.42.0.140



Obr. 7.4: Hodnoty vybraných metrík zachytených za 1 hodinu na IP adrese 10.42.0.140

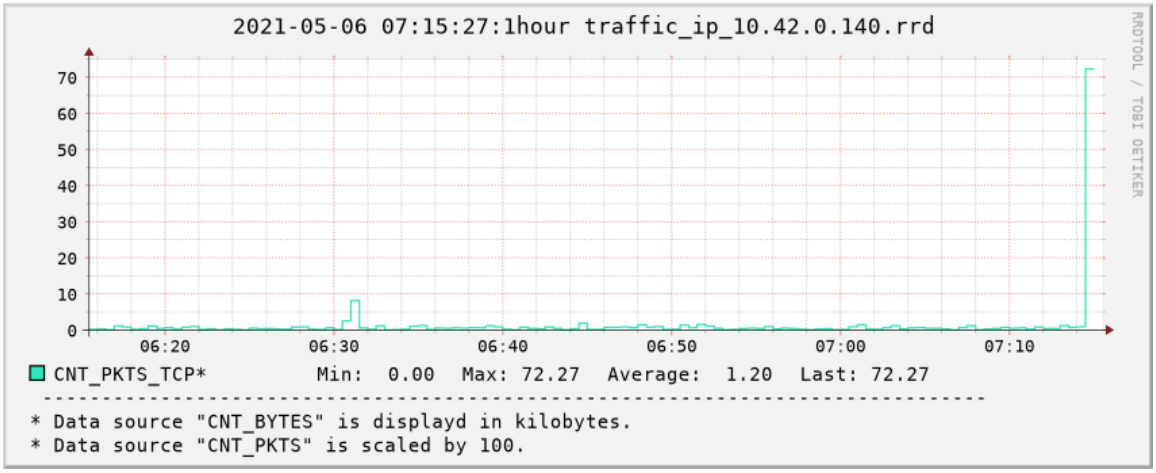
7.4 Zhodnotenie experimentu

Vrámci experimentu bolo analyzovaných **16345710** paketov prevažne s adresami protokolu IPv4 zachytených počas približne 22 hodín.

Protokol	Počet bajtov	Počet paketov	Počet percent bajtov	Počet percent paketov
TCP	2295578268	12439422	53.9996	76.1021
UDP	29105656	3638207	0.6846	22.2579

Tabuľka 7.1: Rozloženie bajtov a paketov pre protokoly

V predošlej podkapitole na obrázkoch 7.1 a 7.3 je zachytená sieťová prevádzka v dobe prebiehajúceho útoku počas 1 minúty. Graf 7.1 zachytáva sieťovú prevádzku všetkých IP adries v sieti a zahŕňa zdroj útoku aj cieľ útoku. Na grafe 7.3 je zachytená prevádzka na cieli útoku špecifikovanom IP adresou 10.42.0.140. Na tomto obrázku je možné pozorovať prudký nárast počtu TCP paketov za krátky časový interval, čo vykazuje známky abnormálneho správania.



Obr. 7.5: Hodnoty počtu TCP paketov za hodinu na IP adrese 10.42.0.140

Na grafe 7.5 sú zobrazené hodnoty počtu TCP paketov uložených v RRDtool databáze pre IP adresu 10.42.0.140 za jednu hodinu. Na tomto grafe je možné pozorovať obvyklé hodnoty vybranej metriky a taktiež niekoľko násobné navýšenie v čase prichádzajúceho útoku. Cieľom tohto experimentu bolo extrahovať vybrané metriky a správne ich vizualizovať do grafov. Okrem tohto cieľa bol experiment vykonaný so zámerom potvrdiť, že abnormálne správanie bude navrhnutým modulom zachytené v krátkom časovom intervale.

7.4.1 Pamäťová náročnosť uložených profilov

Vrámci experimentu boli vytvorené 4 databázové súbory pre jednotlivé úrovne podľa konfigurácie uvedenej v prílohe. Ako je popísané v kapitole 5.3, databázové súbory majú konečnú veľkosť už pri ich vytvorení, nakoľko sa hodnoty RRA archívov pred-vyplnia a s časom sa veľkosť nebude navyšovať. Veľkosti jednotlivých súborov sú uvedené v tabuľke 7.2.

Názov súboru	Veľkosť
traffic_general.rrd	3 006 440 B
traffic_port_80.rrd	3 006 440 B
traffic_protocol_6.rrd	3 435 728 B
traffic_ip_10.42.0.140.rrd	3 865 016 B
Spolu	**13 313 624 B**

Tabuľka 7.2: Veľkosti jednotlivých databázových súborov

Kapitola 8

Možnosti ďalšieho pokračovania

Vrámci tejto práce boli navrhnuté metriky a implementovaný prototyp pre zber a spracovanie rôznych metrík, ktorých úlohou je zostaviť profil siete so zámerom na mitigáciu DDoS útokov. Nasadením programu do reálnej siete s väčšou sieťovou prevádzkou môže dôjsť k zisteniu niekoľkých nedostatkov popísaných ďalej v tejto kapitole.

8.1 Metriky

Metriky navrhnuté vrámci tejto práce sa v reálnych sieťach môžu ukázať ako nedostatočné a ďalším rozšírením nástroja môže byť ich rozšírenie, nakoľko boli navrhnuté metriky týkajúce sa útokov najmä na sieťovej a transportnej vrstve.

8.2 Pridanie nastavenia hraníc akceptovateľnej prevádzky

Vrámci tejto práce sú počítané štatistiky za účelom poskytnúť sieťovým administrátorom lepší prehľad prevádzke vrámci nakonfigurovaných sietí. Ďalším vývojom tejto práce by mohlo byť pridanie konfigurácie maximálnych hodnôt pre jednotlivé úrovne a taktiež prepojenie s modulom poskytujúcim upozornenia v prípade prekročenia týchto hraníc.

8.3 Predikcia

Databázové súbory, ktoré sú výstupom tejto práce uchovávajú dáta reprezentujúce minulosť o sieťovej prevádzke a zachytávajú správanie jednotlivých sieťových entít. Na základe týchto dát by bolo možné pridať predikčné algoritmy pre zistenie vývoja sieťovej prevádzky do budúcnosti. Nástroj RRDtool poskytuje okrem konsolidačných funkcií aj funkcie na odhalenie odlišného správania v časových radoch jednotlivých zdrojov údajov pomocou algoritmu Holt-Winters, ktorý umožňuje modelovanie a predpovedanie správania časových radov.

Kapitola 9

Záver

Vrámci tejto práce boli navrhnuté metriky pre mitigáciu rôznych druhov DDoS útokov na sieťovej a transportnej vrstve. Bol navrhnutý program na spracovanie prichádzajúcich dát na základe užívateľom špecifikovanej konfigurácie. Konfigurácia tohoto programu umožňuje užívateľovi nastaviť monitorovanie ľubovoľnej siete, prípadne niekoľkých sietí. Štatistiky o prevádzke na konfigurovaných sieťach sú počítané pre rôzne úrovne detailu - podsiete, IP adresy, služby, porty.

Úvodný krok tejto práce spočíval v zoznámení sa s potrebou monitorovania sietí, vytvárania profilov sieťových entít a prínosom tohto princípu pre napríklad sieťových administrátorov. Súčasné prístupy k monitorovaniu sietí sú opísané v kapitole 2, ďalej táto kapitola opisuje základné informácie o systéme, NEMEA, do ktorého je navrhnutý nástroj integrovaný.

Pre vytvorenie kolekcie metrík bolo potrebné naštudovať literatúru popisujúcu rôzne DDoS útoky a príznaky týchto útokov. Kapitola 3 popisuje rozdelenie DDoS útokov podľa výskytu na jednotlivých vrstvách modelu OSI a venuje sa aj známym obranným mechanizmom. Navrhnuté metriky určené pre odhalenie DDoS útokov na sieťovej, transportnej a čiastočne aj aplikačnej vrstve sú uvedené v kapitole 4.

Ďalej boli preskúmané možnosti ukladania histórie sieťovej prevádzky pre poskytnutie priestorovo efektívneho úložiska dát bola vybraná RRDtool databáza. Bolo potrebné naštudovať spôsoby správnej konfigurácie tohto nástroja a taktiež jeho použitie v programe napísanom v programovacom jazyku C++.

Prvým krokom k príprave implementácie programu pre zber dát, agregáciu a počítanie štatistík bolo potrebné naštudovať princíp činnosti modulu `ipfixprobe`[6], nakoľko tento modul systému NEMEA zabezpečuje spracovanie dát zo sieťového rozhrania a odosielanie údajov o tokoch na vstupné rozhranie implementovaného nástroja a spôsob vymieňania si správ vo forme UniRec záznamov s rôznymi poľami medzi jednotlivými časťami systému NEMEA.

Prvotným výstupom tejto práce boli grafy, tvorené vrámci nástroja implementovaného vrámci tejto práce v jazyku C++. Pre poskytnutie jednoduchšieho spracovania uložených informácií a menšiu časovú náročnosť navrhnutého nástroja bol navrhnutý modul v jazyku Python, nakoľko knižnica `rrdtool` jazyka Python použitá v tomto module, poskytuje prehľadnejšie zdokumentované funkcie pre základne operácie s databázovými súbormi. Tento modul, určený pre spracovanie výstupných databázových súborov, je popísaný v podkapitole 6.7 a okrem grafického výstupu podporuje aj iné formáty.

Kapitola 7 obsahuje popis vykonaných experimentov a taktiež aj ukážky výstupu nad zachytenými dátami. Táto kapitola taktiež stručne popisuje nástroje použité pri vykonaní

experimentu a analýze. Z uskutočneného experimentu je možné pozorovať, že navrhnutý nástroj počíta jednotlivé nakonfigurované metriky a dokáže sieťové anomálie vykresliť do grafu, z ktorých je možné pozorovať odchýlky od bežného správania.

Nakoľko sa táto práca venuje metrikám na sieťovej, transportnej vrstve a aplikačnej vrstve pre protokoly HTTP a DNS, vytvára to priestor pre ďalší rozvoj. V kapitole 8 sú diskutované možnosti ďalšieho rozvoja navrhnutého nástroja.

Literatúra

[1] *Introduction to Cisco IOS NetFlow - A Technical Overview* [online]. Cisco, Jan 2019 [cit. 2021-05-02]. Dostupné z: https://www.cisco.com/c/en/us/products/collateral/ios-nx-os-software/ios-netflow/prod_white_paper0900aecd80406232.html.

[2] Bader, A., Kopp, O. a Falkenthal, M. Survey and Comparison of Open Source Time Series Databases. In: Mitschang, B., Nicklas, D., Leymann, F., Schöning, H., Herschel, M. et al., ed. *Datenbanksysteme für Business, Technologie und Web (BTW 2017) - Workshopband*. Bonn: Gesellschaft für Informatik e.V., 2017, s. 249–268.

[3] Belenky, A. a Ansari, N. On deterministic packet marking. In:. 2007, sv. 51, č. 10, s. 2677–2700 [cit. 2021-04-12]. DOI: https://doi.org/10.1016/j.comnet.2006.11.020. ISSN 1389-1286. Dostupné z: https://www.sciencedirect.com/science/article/pii/S1389128606003562.

[4] Bloom, B. H. Space/Time Trade-Offs in Hash Coding with Allowable Errors. *Commun. ACM*. New York, NY, USA: Association for Computing Machinery. júl 1970, zv. 13, č. 7, s. 422–426. DOI: 10.1145/362686.362692. ISSN 0001-0782. Dostupné z: https://doi.org/10.1145/362686.362692.

[5] Bozicevic, V. *DDos Quick Start Guide* [online]. Mar 2021 [cit. 2021-04-19]. Dostupné z: https://www.globaldots.com/blog/ddos-quick-start-guide.

[6] Cejka, T., Bartos, V., Svepes, M., Rosa, Z. a Kubatova, H. *Pfixprobe - IPFIX flow exporter* [online]. [cit. 2021-04-11]. Dostupné z: https://github.com/CESNET/ipfixprobe.

[7] Cejka, T., Bartos, V., Svepes, M., Rosa, Z. a Kubatova, H. NEMEA: A Framework for Network Traffic Analysis. In: *12th International Conference on Network and Service Management (CNSM 2016)*. 2016 [cit. 2021-04-12]. DOI: 10.1109/CNSM.2016.7818417. Dostupné z: http://dx.doi.org/10.1109/CNSM.2016.7818417.

[8] Chapman, D. B. a Zwicky, E. D. Chapter 6: Packet Filtering. In: *Building Internet Firewalls*. [cit. 2021-04-12]. ISBN 1-56592-124-0. Dostupné z: http://web.deu.edu.tr/doc/oreily/networking/firewall/ch06_01.htm.

[9] CISA. *DDoS QUICK GUIDE* [online]. Október 2020 [cit. 2021-04-20]. Dostupné z: https://us-cert.cisa.gov/sites/default/files/publications/DDoS%20Quick%20Guide.pdf.

[10] Claise, B. *Cisco Systems NetFlow Services Export Version 9* [online]. Oct 2004 [cit. 2021-05-03]. Dostupné z: https://www.rfc-editor.org/rfc/rfc3954.txt.

[11] Cloudflare. *What is a Botnet?* [online]. [cit. 2021-05-03]. Dostupné z: https://www.cloudflare.com/learning/ddos/what-is-a-ddos-botnet/.

[12] Cloudflare. *What's an Application Layer DDoS Attack* [online]. [cit. 2021-04-19]. Dostupné z: http://www.cloudflare.com/learning/ddos/application-layer-ddos-attack.

[13] Defence Inteligence. *These 6 DNS Attacks Threaten Your Business* [online], 18. May 2017 [cit. 2021-05-04]. Dostupné z: https://defintel.com/blog/index.php/2017/05/these-6-dns-attacks-threaten-your-business.html.

[14] Douligeris, C. a Mitrokotsa, A. DDoS attacks and defense mechanisms: a classification. In: *Proceedings of the 3rd IEEE International Symposium on Signal Processing and Information Technology (IEEE Cat. No.03EX795)*. 2003, s. 190–193 [cit. 2021-04-10]. DOI: 10.1109/ISSPIT.2003.1341092.

[15] Douligeris, C. a Mitrokotsa, A. *DDoS attacks and defense mechanisms: classification and state-of-the-art*. 2003. [cit. 2021-04-10]. Dizertačná práca. University of Piraeus.

[16] Durcekova, V., Schwartz, L. a Shahmehri, N. Sophisticated Denial of Service attacks aimed at application layer. In: *2012 ELEKTRO*. 2012, s. 55–60 [cit. 2021-04-10]. DOI: 10.1109/ELEKTRO.2012.6225571.

[17] Edgewall Software. *Munin* [online], 10. marca 2021 [cit. 2021-04-12]. Dostupné z: http://munin-monitoring.org/.

[18] Emmitt, J. *RMON: A Closer Look at Remote Network Monitoring* [online], 11. June 2020 [cit. 2021-05-09]. Dostupné z: https://www.kaseya.com/blog/2020/06/11/rmon-remote-network-monitoring/.

[19] Encyclopedia, P. *Management Information Base (MIB)* [online]. [cit. 2021-04-12]. Dostupné z: https://networkencyclopedia.com/management-information-base-mib/.

[20] GmbH, X. D. *DEFENDING SERVICES AGAINST DDOS ATTACKS IS ABOUT UNDERSTANDING THE COMPLEXITY OF MULTI-VECTOR THREATS* [online], 11. May 2018 [cit. 2021-05-03]. Dostupné z: https://www.xantaro.net/en/tech-blogs/complexity-of-multi-vector-ddos-threats/.

[21] Homoliak, I. *METRIKY PRO DETEKCI ÚTOKŮ V SÍŤOVÉM PROVOZU*. Česká Republika, 2012. [cit. 2021-04-13]. Diplomová práca. Vysoké Učení technické v Brně, Fakulta informačních technologií. Dostupné z: https://www.vutbr.cz/www_base/zav_prace_soubor_verejne.php?file_id=118417.

[22] Imperva. *What is a Smurf attack* [online]. [cit. 2021-05-04]. Dostupné z: https://www.imperva.com/learn/ddos/smurf-attack-ddos/.

[23] Kresťan, F. *Automatic parameters estimation for DDoS attacks mitigation*. Česká Republika, 2019. [cit. 2021-04-11]. Diplomová práca. České vysoké učení technické v Praze, Fakulta informačních technologií. Dostupné z: https://netmon.fit.cvut.cz/theses/krestanfilip-2019.pdf.

[24] Lehtinen, P. *Jansson* [online]. [cit. 2021-04-11]. Dostupné z: https://github.com/akheron/jansson.

[25] Mirkovic, J. a Reiher, P. A Taxonomy of DDoS Attack and DDoS Defense Mechanisms. ACM SIGCOMM Computer Communications Review. In:. Apríl 2004, sv. 34, č. 2 [cit. 2021-04-12]. DOI: 1145/997150.997156. ISSN 0146-4833.

[26] Nagios Enterprises, LLC.. *Nagios* [online]. 2021 [cit. 2021-04-12]. Dostupné z: https://www.nagios.org/.

[27] Netscout. *What is an SSL/TLS Exhaustion DDoS Attack?* [online]. [cit. 2021-04-19]. Dostupné z: https://www.netscout.com/what-is-ddos/ssl-tls-exhaustion.

[28] Networks, A. *What Is NetFlow? Intro to Monitoring Network Traffic - Auvik Networks* [online], 11. Feb 2021 [cit. 2021-05-03]. Dostupné z: https://www.auvik.com/franklyit/blog/netflow-basics/.

[29] Oetiker, T. *[rrd-announce] ANNOUCNE: RRDtool 1.0.0*, 16. Jul 1999 [cit. 2021-04-12]. Dostupné z: https://lists.oetiker.ch/pipermail/rrd-announce/1999-July/000007.html.

[30] Oetiker, T. About RRDtool. *The Time Series Database* [online], 20. Feb 2017 [cit. 2021-03-20]. Dostupné z: https://oss.oetiker.ch/rrdtool/index.en.html.

[31] Oetiker, T. a Forster, F. *Rrd-beginners*, 2. Apr 2019 [cit. 2021-04-11]. Dostupné z: https://oss.oetiker.ch/rrdtool/doc/rrdupdate.en.html.

[32] Patel, K. *Rrd-beginners*. Apr 2019 [cit. 2021-04-12]. Dostupné z: https://oss.oetiker.ch/rrdtool/tut/rrd-beginners.en.html.

[33] Pilli, E., Joshi, R. a Niyogi, R. Router and Interface Marking for Network Forensics. In: [online]. Január 2011, sv. 361, s. 209–220 [cit. 2021-05-07]. DOI: 10.1007/978-3-642-24212-0_16. ISBN 978-3-642-24211-3.

[34] Sanfilippo, S. *Hping* [online]. 2006 [cit. 2021-05-05]. Dostupné z: http://www.hping.org/.

[35] Savage, S., Wetherall, D., Karlin, A. a Anderson, T. Practical network support for IP traceback. In:. August 2000 [cit. 2021-04-12]. DOI: 10.1145/347057.347560.

[36] Siggins, M. *What is RMON?* [online]. 2021 [cit. 2021-04-14]. Dostupné z: https://www.dpstele.com/blog/what-is-rmon.php.

[37] The OpenNMS Group, Inc.. *OpenNMS* [online]. 2021 [cit. 2021-04-12]. Dostupné z: https://www.opennms.com/.

[38] Virkki, J. J. *Libbloom* [online]. [cit. 2021-04-11]. Dostupné z: https://github.com/jvirkki/libbloom.

[39] Wikipedia.org. *IP Flow Information Export* [online], 7. Nov 2020 [cit. 2021-05-03]. Dostupné z: https://en.wikipedia.org/wiki/IP_Flow_Information_Export.

[40] Wireshark Foundation. *About Wireshark* [online]. [cit. 2021-05-05]. Dostupné z: https://www.wireshark.org/.

Príloha A

Skratky

ACK Acknowledgment Number.

CF Consolidation Function.

DDoS Distributed Denial-of-Service.

DNS Domain Name System.

DoS Denial-of-Service.

DS Data Source.

DST Data Source Type.

FIN Finish.

GNU GPL GNU General Public License.

HTTP Hypertext Transfer Protocol).

ICMP Internet Control Message Protocol.

IDS Intrucion Detection Systems.

IETF Internet Engineering Task Force.

IP Internet Protocol.

IPFIX IP Flow Information Export.

MAC Media Access Control.

MIB Management information base.

MRTG Multi Router Traffic Grapher.

OID Object Identifier.

OSI Open Systems Interconnection Reference Model.

PSH Push.

RMON Remote Monitoring.

RRA Round Robin Archive.

RRDtool Round Robin Database Tool.

RST Reset the Connection.

SMTP Simple Mail Transfer Protocol.

SNMP Simple Network Management Protocol.

SQL Structured Query Language.

SSH Secure Shell Protocol.

SSL Secure Sockets Layer.

SYN Synchronize sequence numbers.

TCP Transmission Control Protocol.

TLS Transport Layer Security.

UDP User Datagram Protocol.

XML eXtensible Markup Language.

Príloha B

Konfiguračný súbor použitý pre experimenty

```
[
    {
        "network": ["0.0.0.0/0"]
    },
    {
        "level": "general",
        "metrics": [
            "CNT_FLOWS",
            "CNT_PEERS",
            "CNT_BYTES",
            "CNT_PACKETS",
            "CNT_SRC_PORTS",
            "CNT_PROTO",
            "CNT_ACK_ANY"
        ]
    },
    {
        "level": "ip",
        "keys": ["10.42.0.140"],
        "metrics": [
            "CNT_PKTS_TCP",
            "CNT_SYN_ACK"
        ]
    },
    {
        "level": "port",
        "keys": ["80"]
    },
    {
        "level": "protocol",
        "keys": ["6"],
        "metrics": [
            "CNT_SYN_ACK"
        ]
    }
]
```

Výpis kódu B.1: Príklad konfiguračného súboru

Príloha C

Obsah priloženej SD karty

- **docs** - dokumentácia a návody k implementačnej časti bakalárskej práce
- **src** - zdrojové kódy navrhnutých modulov, priečinok so zdrojovými súbormi LaTeX-u textovej časti bakalárskej práce
- **experiment** - databázové súbory vykonaného experimentu a ich grafické spracovanie
- **README** - popis obsahu priloženej SD karty