



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

DEPARTMENT OF COMPUTER SYSTEMS

**OBOHACOVÁNÍ IP TOKŮ O DATA
Z EXTERNÍCH ZDROJŮ**

ENRICHMENT OF IP FLOW RECORDS WITH DATA FROM EXTERNAL SOURCES

BAKALÁŘSKÁ PRÁCE

BACHELOR'S THESIS

AUTOR PRÁCE

AUTHOR

ADAM ZVARA

VEDOUcí PRÁCE

SUPERVISOR

Ing. MARTIN ŽÁDNÍK, Ph.D.

BRNO 2023

Zadání bakalářské práce



148136

Ústav: Ústav počítačových systémů (UPSY)
Student: **Zvara Adam**
Program: Informační technologie
Specializace: Informační technologie
Název: **Obohacování IP toků o data z externích zdrojů**
Kategorie: Počítačové sítě
Akademický rok: 2022/23

Zadání:

1. Nastudujte protokol IPFIX a modulární kolektor IPFIXcol2.
2. Seznamte se s problematikou obohacování záznamů IP toků o data z externích zdrojů.
3. Navrhněte podpůrné rozhraní a možné moduly pro obohacování IP toků na kolektoru.
4. Implementujte rozhraní a vybrané rozšiřující moduly kolektoru.
5. Vytvořenou implementaci ověřte a proveďte výkonnostní testy.
6. Diskutujte dosažené výsledky a možná rozšíření.

Literatura:

- Dle pokynů vedoucího.

Při obhajobě semestrální části projektu je požadováno:

- Splnění bodů 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: 1.11.2022
Termín pro odevzdání: 10.5.2023
Datum schválení: 31.10.2022

Abstrakt

Táto práca sa zaoberá obohacovaním tokov získaných z monitorovacej architektúry typu NetFlow/IPFIX realizovaného na úrovni kolektoru. Jej hlavným cieľom je rozšírenie kolektoru IPFIXcol2 o rozhrania využiteľné pri vytváraní nových modulov vykonávajúcich obohacovanie tokov. Na základe nového návrhu boli vytvorené implementácie modulov ASN a GeoIP obohacujúce toky číslami autonómnych systémov a geolokačnými informáciami. Následne boli nad týmito modulmi zrealizované testy hodnotiace ich dopad na výkonnosť kolektoru.

Abstract

This thesis addresses the enhancement of network flows from the viewpoint of a collector, obtained through NetFlow/IPFIX monitoring architecture. The primary objective is to extend the IPFIXcol2 collector with interfaces that can be utilized to develop new modules capable of enriching network flows. The proposed approach involves designing and implementing two modules, namely ASN and GeoIP, to enrich flows with information related to autonomous system numbers and geolocation. After implementing these modules, performance evaluations are conducted to assess their impact on the collector's overall efficiency.

Kľúčové slová

monitorovanie sietí, IPFIX, NetFlow, tok, kolektor, IPFIXcol2, modularita, obohacovanie tokov, autonómne systémy, IP geolokácia

Keywords

network monitoring, IPFIX, NetFlow, flow, collector, IPFIXcol2, modularity, flow enrichment, autonomous systems, IP geolocation

Citácia

ZVARA, Adam. *Obohacování IP toků o data z externích zdrojů*. Brno, 2023. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedoucí práce Ing. Martin Žádník, Ph.D.

Obohacování IP toků o data z externích zdrojů

Prehlásenie

Prehlasujem, že som túto bakalársku prácu vypracoval samostatne pod vedením pána Ing. Martina Žádníka Ph.D. Další informace mi poskytl Ing. Lukáš Huták. Uviedol som všetky literárne pramene, publikácie a ďalšie zdroje, z ktorých som čerpal.

.....
Adam Zvara
7. mája 2023

Podakovanie

Rád by som poďakoval Ing. Lukášovi Hutákovi za cenné rady, ktoré mi poskytoval pri vytváraní tejto práce. Zároveň by som chcel poďakovať mojej rodine a priateľke, ktorí ma podporovali počas celej doby môjho štúdia.

Obsah

1	Úvod	2
2	Monitorovanie siete protokolom IPFIX	4
2.1	Aktívne a pasívne monitorovanie	4
2.2	Architektúra NetFlow a IPFIX	5
2.3	Protokol IPFIX	9
2.4	Popis práce kolektoru	12
3	Analýza kolektoru IPFIXcol2	15
3.1	Prehľad architektúry	15
3.2	Popis činnosti jadra kolektoru	16
3.3	Správa šablón	19
3.4	Zásuvné moduly	20
4	Návrh obohacovania IPFIX správ	22
4.1	Autonómne systémy a geolokácia	22
4.2	Zhodnotenie pôvodného návrhu	23
4.3	Nový návrh obohacovania správ	24
4.4	Popis činnosti zastrešujúcich modulov	29
5	Realizácia obohacovania správ a testovanie	30
5.1	Ukázkový obohacujúci modul	32
5.2	Moduly ASN a GeoIP	32
5.3	Optimalizácia modifikácie záznamov	35
5.4	Budúci rozvoj projektu	37
6	Záver	39
	Literatúra	41
	A Obsah priloženého pamäťového média	43
	B Konfigurácia nových modulov	44

Kapitola 1

Úvod

V dnešnej digitálnej dobe pracujeme s internetom pri rôznych každodenných aktivitách. Môže ísť o rozličné formy komunikácie, zdieľania dát či online transakcie, ktoré predstavujú veľké množstvo citlivých údajov prenášaných cez sieť. S jeho zvýšeným používaním sa však spája nebezpečenstvo v podobe útokov, ktorých hlavným cieľom je získať prístup k týmto údajom. Bezpečnosť na internete sa tak dostáva do popredia a v súčasnosti patrí k jeho kľúčovým častiam.

Existuje viacero možností, ktorými môžu poskytovatelia internetových služieb odhaliť útoky vo svojej sieti. Medzi jednu z nich patrí monitorovanie, ktoré predstavuje proces získavania informácií o zariadeniach komunikujúcich v sieti. Na základe analýzy týchto spojení je možné v reálnom čase identifikovať potenciálne hrozby a predísť ich výskytu. Okrem bezpečnostného významu má monitorovanie dôležitú úlohu aj pri správe siete. K technológiám, ktoré sa najčastejšie používajú na monitorovanie sietí patria NetFlow a IPFIX.

Ich podstatou je analýza paketov prechádzajúcich vybranými bodmi siete, získavanie informácií o sieťových tokoch a ich odosielanie monitorovacou sondou na zberné miesto zvané kolektor. Ten umožňuje ukladanie získaných dát v rôznych formátoch alebo ich preposielanie systémom pre automatizovanú analýzu. Príkladom open-source kolektoru vyvíjaného spoločnosťou CESNET z.s.p.o. je IPFIXcol2. Pri exportovaní dát na kolektor sa však môžeme stretnúť s problémom, kedy informácie získané o toku nemusia byť dostačujúce na jeho popis. To môže nastať v prípade, že exportér nemá k dátam prístup alebo ním jednoducho nie sú podporované. V takýchto prípadoch je potrebné, aby kolektor bol schopný dohľadať chýbajúce informácie a toky nimi obohatiť.

Cieľom tejto práce je rozšírenie kolektoru IPFIXcol2 o možnosti rozširovania IPFIX správ ako aj integrovanie vybraných modulov využívajúcich toto rozhranie. Prvým z nich je modul ASN, ktorý obohacuje toky o čísla autonómnych systémov (angl. *autonomous system number*), ktoré unikátne identifikujú prefixy IP adries pod správou jedného operátora. Tieto identifikátory sa pri monitorovaní môžu používať napríklad na lepšiu detekciu zdrojov toku, výpadkov alebo vzorov správania zariadení v sieti. Druhý modul s podobným využitím má názov GeoIP a je určený k pridávaniu označení geografických regiónov, z ktorých tok pochádza alebo kam smeruje. Okrem možností pridávania dát do správ je potrebné, aby bol kolektor schopný časti záznamov upraviť alebo vymazať. Táto vlastnosť sa dá využiť napríklad za účelom zníženia potrebného miesta v prípade dlhodobého uloženia získaných informácií, alebo očistenia tokov od neplatných dát získaných z monitorovacej sondy. Z toho vyplýva sekundárny cieľ práce, ktorým je umožniť kolektoru filtrovať celé toky alebo ich časti.

Nadväzujúca kapitola 2 bližšie popisuje spôsob monitorovania sietí pomocou technológií NetFlow a IPFIX, časti monitorovacej architektúry a použité postupy, no zároveň sa venuje aj samotnému protokolu IPFIX. Základné vlastnosti a spôsoby práce so získanými informáciami o tokoch tohoto protokolu sú potrebné pre pochopenie fungovania kolektoru. Kapitola 3 sa venuje návrhu kolektoru IPFIXcol2 a detailne vysvetľuje všetky jeho súčasti, ktorých znalosť je nevyhnutná pre vývoj nových modulov a rozhrania na obohacovanie správ. Princíp samotného obohacovania sa nachádza v kapitole 4 spolu s jeho aplikovaním na spomínaných moduloch. Kapitola 5 popisuje vybrané časti implementácie a spôsoby testovania novej časti kolektoru s dôrazom kladeným na výkonnostné testy. Záverečná kapitola 6 sumarizuje dosiahnuté výsledky tejto práce.

Kapitola 2

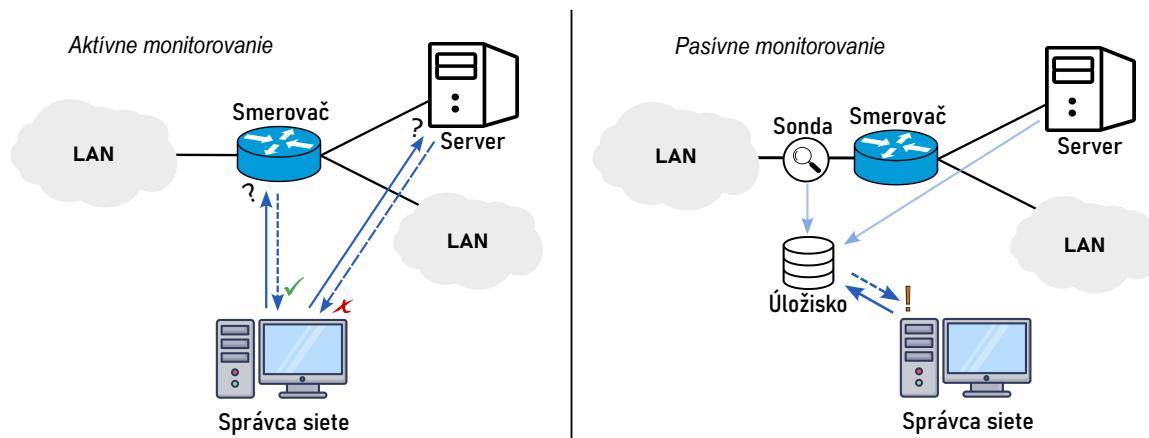
Monitorovanie siete protokolom IPFIX

Monitorovanie patrí v dnešnej dobe k nevyhnutným aktivitám spojených so správou sietí. S narastajúcim počtom komunikujúcich zariadení sa zvyšuje nárok na ich správu. Pomocou sledovania internetovej prevádzky a aktivity zariadení sú administrátori schopní detekovať a riešiť incidenty skôr, než by mohli viesť k problémom spojených so stratou dát alebo úplným výpadkom siete. Okrem toho ním získavajú všeobecný prehľad o dôležitých vlastnostiach siete, ktorými sú napríklad vyťaženosť alebo dostupnosť. Pomocou toho je možné identifikovať kritické úseky infraštruktúry alebo časy prevádzky a na základe týchto informácií efektívne využívať zdroje spojené s jej riadením. Spolu s bezpečnostným významom monitorovania začína byť zrejmé, prečo sa jedná o nástroj, ktorý netreba podceňovať.

Monitorovacích nástrojov a technológií, po ktorých môže človek siahnuť, je veľké množstvo a každá z nich sa zameriava na iný aspekt siete. V tejto kapitole sú predstavené niektoré z nich, najmä protokoly NetFlow a IPFIX, ktorými sa zaoberá zvyšok práce. Nasleduje predstavenie ich architektúry spolu s podrobným popisom zberu dát a formátu prenášaných správ, ako aj funkcie samotného kolektora.

2.1 Aktívne a pasívne monitorovanie

Existujú dva spôsoby monitorovania sietí, a to aktívne a pasívne [8]. **Aktívne** monitorovanie spočíva v periodickom zisťovaní stavu liniek, aplikácií či iných sieťových prvkov. Ak zariadenie určitú dobu neodpovedá, je považované za nedostupné a informácia je predaná správcovi siete. **Pasívne** monitorovanie je charakterizované najmä sledovaním a zaznamenávaním aktivít v sieti bez nutnosti periodického zisťovania stavu zariadení. Do tejto kategórie patria už spomínané protokoly NetFlow a IPFIX, ktorých úlohou je sledovanie komunikácie zariadení pomocou analýzy paketov prechádzajúcich cez sieť. Rozdiely týchto prístupov je možné vidieť na obrázku 2.1. V praxi sa využívajú oba spôsoby kombinovane, čo poskytuje administrátorom komplexnejší prehľad o aktivitách na sieti.



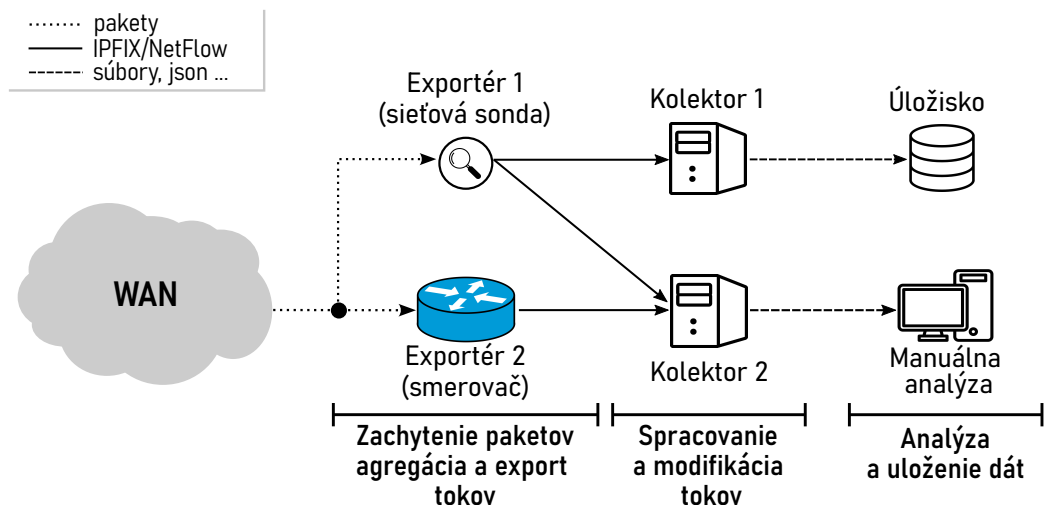
Obr. 2.1: Porovnanie spôsobov aktívneho a pasívneho monitorovania sietí

2.2 Architektúra NetFlow a IPFIX

Protokol NetFlow, navrhnutý spoločnosťou Cisco v roku 1996, položil základy monitorovania pomocou sledovania tokov tak, ako ho poznáme dnes [10]. Existuje niekoľko verzií¹ tohto protokolu a medzi jednu z najstarších a v súčasnosti stále používaných sa radí verzia 5, ktorá fixne definuje zbierané údaje o tokoch. Od tejto verzie sa odvíjal rad ďalších až po prelomovú 9. verziu, ktorá prešla na dynamický spôsob určovania monitorovaných vlastností pomocou šablón popisujúcich podobu záznamov tokov v prenášaných správach. Tento mechanizmus dodáva monitorovaniu flexibilitu a správy môžu vďaka nemu obsahovať viacero druhov záznamov, ktoré sledujú rozličné atribúty tokov. Protokol IPFIX, popísaný v sekcii 2.3, predstavuje nástupcu NetFlow, ktorý vznikol s cieľom vytvorenia štandardizovaného spôsobu komunikácie medzi monitorovacími zariadeniami rôznych výrobcov.

Na obrázku 2.2 je zobrazená architektúra využívaná pri monitorovaní sietí pomocou NetFlow/IPFIX [6]. Proces monitorovania je možné rozdeliť na 3 základné fázy. Prvou z nich je zachytenie paketov vo vybraných bodoch siete, ktoré sú následne analyzované. Zariadenie realizujúce túto činnosť, tiež označované ako **exportér**, agreguje získané dáta z paketov v podobe tokov. Exportérom môže byť ľubovoľný sieťový prvok (napr. smerovač) obohatený o schopnosť merania tokov, klasický počítač pripojený do siete s monitorovacím softvérom alebo špeciálna sieťová sonda. Informácie o tokoch sú odosielané z exportéru pomocou príslušného protokolu na zberné miesto zvané **kolektor**. Druhá fáza popisuje činnosť kolektoru, ktorou je spracovanie a úprava dát. Kolektor prijíma správy z viacerých exportérov, následne môže v závislosti od jeho konfigurácie toky filtrovať, agregovať, anonymizovať, obohacovať ďalšími informáciami, atď. Upravené záznamy o tokoch odosiela v príslušnom formáte na úložisko, alebo ich priamo poskytuje ostatným aplikáciám k analýze, ktorá je poslednou fázou procesu monitorovania. Analýza môže prebiehať automatizovane alebo manuálne. Prvý spôsob je vhodný na spracovanie veľkého množstva dát a najčastejšie sa využíva pri detekcii bezpečnostných anomálií, má však využitie aj pri tvorení štatistík o sieti, alebo pri automatizovanom hlásení chýb zariadení. Pri druhom spôsobe správcovia sietí využívajú aplikácie tvoriace vizualizácie nad získanými dátami, ktoré im poskytujú lepší prehľad o správaní zariadení na sieti.

¹Prehľad verzií NetFlow dostupný na: ibm.co/3ZD2rve



Obr. 2.2: Príklad monitorovacej architektúry NetFlow/IPFIX [6]

Sieťový tok

Základným pojmom spojeným s uvedenými protokolmi je **tok** (angl. *flow*). Podľa štandardu RFC 7011 [4] je internetovou komunitou používaných niekoľko definícií tohto pojmu. Samotný štandard definuje sieťový tok nasledovne:

Sieťový tok tvorí sada paketov prechádzajúcich pozorovacím bodom počas určitého časového intervalu. Všetky pakety patriace jednému toku obsahujú množinu spoločných vlastností, ktorá ho jednoznačne identifikuje.

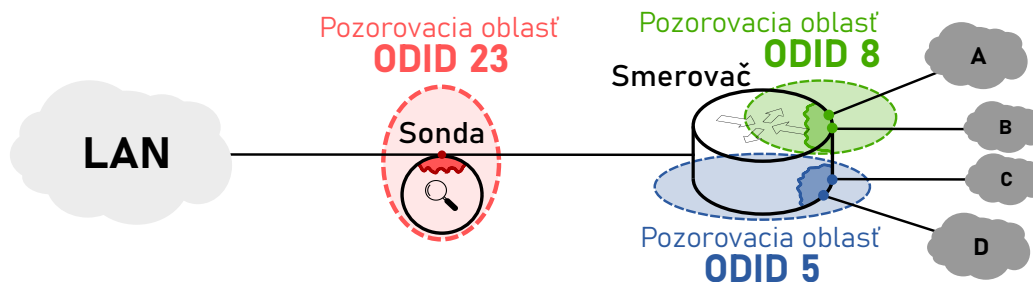
Sieťový tok obsahuje informácie charakterizujúce komunikáciu zariadení na sieti. Sú tvorené najmä položkami získanými zo sieťových a transportných hlavičiek paketov, môžu však pochádzať aj z linkovej či aplikačnej vrstvy. Okrem hodnôt získaných priamo z paketu (napr. dĺžka paketu) môže tok obsahovať aj hodnoty z neho odvodené (napr. číslo autonómneho systému). Množina spoločných vlastností, na základe ktorej sú pakety rozdeľované do príslušných tokov, sa nazýva **klúč toku** (angl. *flow key*). Klúč si môže každé monitorovacie zariadenie definovať samostatne, najčastejšie sa však používa päťica zložená zo zdrojovej a cieľovej IP adresy, zdrojového a cieľového čísla portu a čísla transportného protokolu.

Činnosť exportéru

Ako bolo spomenuté v úvode sekcie 2.2, exportérom môže byť akékoľvek zariadenie pripojené do siete s možnosťou merania a exportovania tokov. V špeciálnom prípade, kedy je exportér dedikovaným zariadením, je označovaný ako sonda. Hlavnou úlohou sondy je pasívne odpočúvanie komunikácie na sieti, čo dáva správcovi siete výhodu, keďže monitorované zariadenia nie sú schopné odhaliť jej prítomnosť. RFC 5470 [12] rozdeľuje jej činnosť na dva procesy: meranie a export.

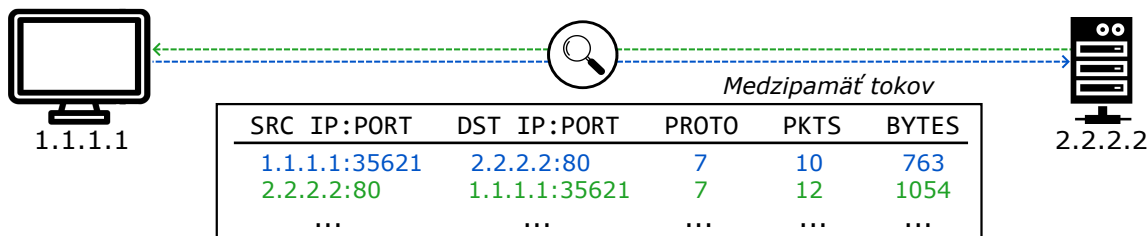
Proces merania začína pozorovaním paketov prechádzajúcich cez rozhranie exportéru. Typicky môže exportér obsahovať viacero rozhraní a na každom z nich môže dochádzať ku zberu rozličných dát pomocou samostatných meracích procesov. Vytvorené toky sú však odosielané spoločným exportným procesom, ktorý znemožňuje presné určenie miesta vytvorenia toku, čo môže byť z pohľadu správcu siete dôležité k určeniu pôvodu problému v sieti.

Štandard preto definuje pojem pozorovacej oblasti (angl. *Observation domain*), ktorá identifikuje meracie procesy v rámci exportéru. Pozorovacie oblasti sú od seba odlišené pomocou identifikátoru ODID (*Observation Domain ID*). Štandardom je odporúčané, aby pridelené identifikátory ODID boli spomedzi všetkých zariadení v danej sieti unikátne, čo umožňuje jednoduché vyhľadanie exportéru, z ktorého tok pochádza. Na obrázku 2.3 je možné vidieť význam zavedenia identifikátoru pozorovacej oblasti konkrétne pri smerovači exportujúcom toky z viacerých sietí. V prípade absencie ODID by pre toky odosielané z tohto zariadenia nebolo možné jednoznačne určiť ich pôvod.



Obr. 2.3: Príklady pozorovacích oblastí architektúry NetFlow/IPFIX

Vlastnosti tokov sú udržiavané v podobe záznamov v **medzipamäti tokov** (angl. *flow cache*) [6]. Proces merania analyzuje pakety prichádzajúce z bodov pozorovania, pre ktoré sú následne extrahované hodnoty kľúčových vlastností unikátnych pre daný tok. Na ich základe sa pokúsi sonda o nájdenie toku v medzipamäti, s ktorým následne pracuje. V prípade, že takýto tok ešte nebol pozorovaný, vytvorí novú položku na jeho popis. Príklad záznamov tokov je možné vidieť na obrázku 2.4, ktorý zachytáva komunikáciu klienta so serverom, kde každý riadok medzipamäte predstavuje jeden tok.



Obr. 2.4: Proces zberu informácií o tokoch pri komunikácií monitorovaných zariadení

Štandardom nie je definované, aké informácie sa o tokoch zbierajú. Existuje však základná a všeobecne využívaná sada meraných vlastností, ktorá je popísaná v tabuľke 2.1. Táto sada obsahuje atribúty hlavičiek sieťovej a transportnej vrstvy, no výrobca exportéru môže často siahnuť aj po atribútoch z aplikačnej vrstvy, ktoré obsahujú zaujímavejšie informácie z pohľadu analýzy. Položky záznamov o tokoch pri technológii IPFIX, popísané pomocou informačných elementov (ich podrobnejší popis sa nachádza v sekcii 2.3), majú jasne stanovený dátový typ a sémantiku. Priebežná aktualizácia položiek počas procesu merania je daná agregáčnou funkciou, ktorá je špecifická v závislosti od typu položky [2]. Je zrejmé, že kľúčové položky definujúce tok nie sú agregované a zároveň nemusí existovať agregáčna funkcia ani pre niektoré neklúčové položky. Príkladom takýchto položiek môže byť adresa zisťovaná pri DNS rezolúcii alebo adresa webovej stránky protokolu HTTP. V ta-

komto prípade nedochádza ku agregácii, ale k jednorázovému uloženiu hodnoty skúmanej vlastnosti do toku.

Sledované vlastnosti	Dátový typ	Agregačná funkcia
Zdrojová IP adresa	IP adresa	
Cieľová IP adresa	IP adresa	
Zdrojový port	Nezáporné číslo	Kľúčová položka
Cieľový port	Nezáporné číslo	
Protokol	Nezáporné číslo	
Čas prvého paketu	Dátum	Minimum
Čas posledného paketu	Dátum	Maximum
TCP príznaky	Nezáporné číslo	Bitový OR
Počet bytov	Nezáporné číslo	Suma
Počet paketov	Nezáporné číslo	Suma

Tabuľka 2.1: Prehľad základných informácií o toku [7]

Druhou časťou funkcie sondy je proces exportu, ktorý rozhoduje o ukončení a odoslaní expirovaných tokov. Ich záznamy sú následne odobrané z medzipamäte tokov a vložené do novej IPFIX správy, ktorá je odoslaná kolektorovi. Tok je odoslaný na export, ak je splnená jedna z nasledujúcich podmienok [7]:

- **Uplynutie aktívneho časovača**

Aktívny časovač sa používa na pravidelné zisťovanie stavu dlho prebiehajúcich tokov, ktoré môžu vzniknúť pri prenose väčšieho množstva dát cez sieť. Ak časový úsek, ktorý tok zachytáva, presiahne tento limit, dochádza k jeho násilnému ukončeniu. Dlhé trvajúce toky sú tak fragmentované a odosielené v podobe kratších tokov. V prípade absencie tohto časovača by takéto toky boli odoslané na kolektor až po ukončení komunikácie medzi zariadeniami a správca by nebol priebežne informovaný o prebiehajúcej komunikácii. Limit sa rádovo pohybuje v jednotkách minút.

- **Uplynutie neaktívneho časovača**

Neaktívny časovač indikuje, že tok nebol určitú dobu aktualizovaný a má značný význam pri transportných protokoloch, pri ktorých nie je možné detekovať ukončenie komunikácie (napr. UDP). Limit sa rádovo pohybuje v desiatkach sekúnd.

- **Prirodzené ukončenie komunikácie**

Pri niektorých transportných protokoloch (napr. TCP paket s príznakmi *FIN* alebo *RST*) je možné ukončiť tok, ak došlo ku zisteniu ukončenia spojenia medzi komunikujúcimi zariadeniami.

- **Na základe iných dôvodov**

Do tejto kategórie môže patriť zmena nastavení sondy alebo zaplnenie medzipamäte nad istú úroveň, ktorá by znemožňovala bezproblémové vytváranie nových tokov.

Výsledkom činnosti exportéru sú IPFIX správy, ktoré popisujú zachytené toky. Tieto správy sú zasielané na jeden alebo viacero kolektorov pomocou protokolu IPFIX, ktorému sa venuje nasledujúca sekcia.

2.3 Protokol IPFIX

Protokol IPFIX je v súčasnosti štandardizovanou metódou odosielania záznamov o tokoch z exportéru na kolektor. Jeho počiatky je možné pozorovať už od roku 2004, kedy pracovná skupina IPFIX² (*IP Flow Information Export*), patriaca pod IETF, definovala všeobecné požiadavky monitorovania sietí na základe sledovania tokov a taktiež analyzovala vhodné protokoly pre tento proces [3]. Ako vyplýva z úvodu sekcie 2.2, *NetFlow v9* bol zvolený za vhodného kandidáta pre vytvorenie nového protokolu, ktorého prvotný návrh bol definovaný v RFC 5101 začiatkom roku 2008. V priebehu nasledujúcich rokov však došlo ku miernym úpravám tejto verzie a v roku 2013 sa tieto zmeny premietli do súčasnej podoby IPFIX definovanej v RFC 7011.

Ako bolo viackrát spomenuté, protokol IPFIX slúži na prenos záznamov o tokoch z exportéra na kolektor [4]. Komunikácia opačným smerom nie je povolená, keďže sa jedná o jednosmerný protokol. Dôsledkom toho je fakt, že všetky informácie, ktoré kolektor vyžaduje ku dekódovaniu záznamov, musia byť súčasťou prenášaných správ (výnimkou sú však informačné elementy). Táto vlastnosť napomáha hlavnému cieľu protokolu IPFIX, ktorým je univerzálnosť. Tá umožňuje komunikáciu medzi exportérmi a kolektormi od rôznych výrobcov, čo je v praxi častým prípadom. Záznamy o tokoch sú v správach zakódované binárne, pričom dôraz je kladený na priestorovú efektívnosť prenášaných dát.

Informačné elementy a dátové typy

Definície prenášaných dát sú popísané informačným modelom, ktorý je ustanovený v RFC 7012 [2]. Informačný model tvoria **informačné elementy** slúžiace na popis položiek záznamov o tokoch nachádzajúcich sa v IPFIX správach. Definícia informačného elementu povinne obsahuje názov, identifikačné číslo, popis, dátový typ a status. Voliteľne môže obsahovať definíciu sémantiky, merané jednotky, rozsah platných hodnôt, atď. Správu informačných elementov zastrešuje organizácia IANA (Internet Assigned Numbers Authority). Príklady informačných elementov je možné vidieť v tabuľke 2.2.

PEN	ID	Názov	Dátový typ	Popis
0	2	packetDeltaCount	unsigned64	Počet prenesených paketov
	8	sourceIPv4Address	ipv4Address	Zdrojová IPv4 adresa
	11	destinationTransportPort	unsigned16	Cielové číslo portu
	80	destinationMacAddress	macAddress	Cielová MAC adresa
6876 ³	166	smtpSubject	string	Subjekt emailovej správy
	553	certToolId	unsigned32	Identifikátor CERT zariadenia
5915 ⁴	890	ingressInterfaceAttr	unsigned64	Port pripojeného virtuálneho zariadenia

Tabuľka 2.2: Časti definícií vybraných informačných elementov

Samotné informačné elementy taktiež podliehali zmenám spolu s protokolom IPFIX. Pôvodný návrh informačného modelu v RFC 5102 [11] definoval základnú množinu všeobecne sledovaných vlastností o tokoch naprieč rôznymi výrobcami, ktorá bola neskôr (v RFC 7012)

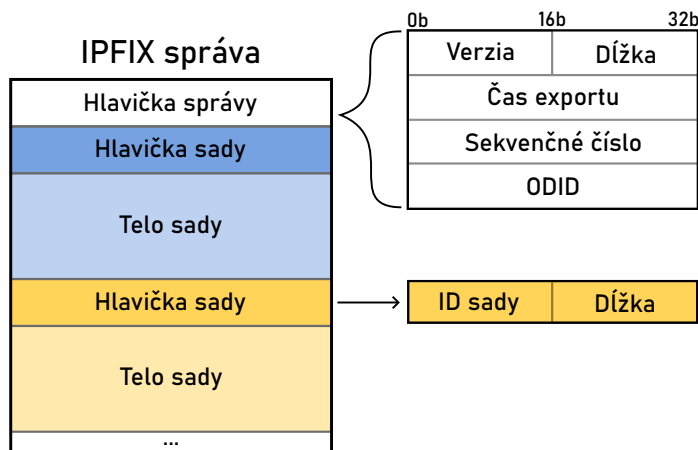
²Prehľad aktivít skupiny: datatracker.ietf.org/group/ipfix

³Registrované entity privátneho rozsahu CERT: tools.netsa.cert.org/cert-ipfix-registry

⁴Registrované entity privátneho rozsahu VMware (skrátенý odkaz): bit.ly/3yrBFtF

presunutá do **spoločného rozsahu**⁵ definícií nového modelu. Keďže pridávanie nových definícií do tohto rozsahu podlieha štandardizovanému procesu, výrobcovia exportérov môžu požiadať o pridelenie **privátneho rozsahu**⁶. Privátny rozsah je identifikovaný pomocou čísla PEN (Private Enterprise Number) a obsahuje definície informačných elementov, ktoré sú špecifické pre daného výrobcu, alebo majú experimentálny charakter. Pre štandardne definované informačné elementy pod správou organizácie IANA je vyhradený privátny rozsah s identifikačným číslom PEN 0.

Štruktúra IPFIX správy



Obr. 2.5: Štruktúra IPFIX správy

Popis IPFIX správy vychádza z jej definície v RFC 7011 [4]. Štruktúra správy, ktorú je možné vidieť na obrázku 2.5, začína povinnou hlavičkou, za ktorou voliteľne nasledujú definície sád obsahujúce dáta rôzneho charakteru v závislosti od ich typu. Na začiatku hlavičky správy je zakódovaná verzia použitého protokolu. Pre IPFIX je táto hodnota 10, pri iných verziách (NetFlow) predstavuje číslo použitého protokolu. Nasleduje dĺžka celej správy (vrátane hlavičky), čas exportu, sekvenčné číslo a identifikátor pozorovacej domény (ODID). Na kódovanie dát správy sa používa *network byte order* (alebo takzvaný *big-endian*) a zariadenia využívajúce inú endianitu musia pri práci s dátami využívať konvertovacie funkcie.

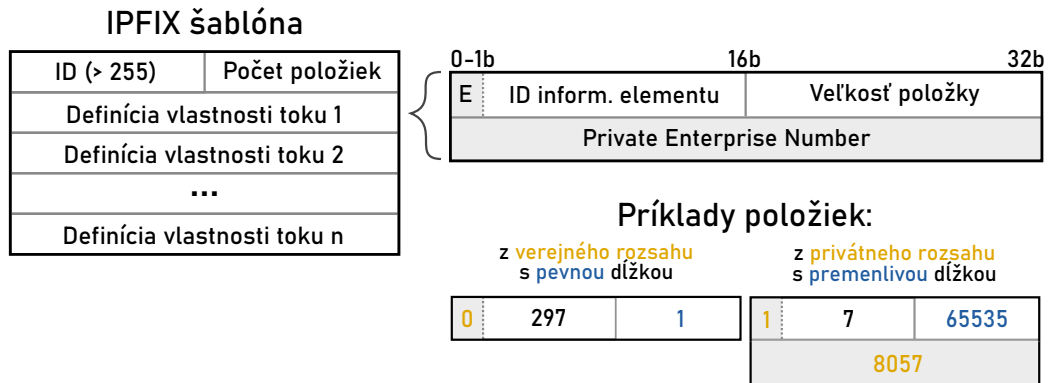
Za hlavičkou samotnej IPFIX správy nasledujú jednotlivé sady, kde každá je zložená z hlavičky a tela. Hlavička sady obsahuje identifikačné číslo spolu s celkovou dĺžkou sady, ktorú definuje. V závislosti od typu sady je odvodená hodnota identifikačného čísla a obsah tela sady. Existujú 3 typy sád, a to sady šablón záznamov o tokoch (angl. *Template Set*), sady šablón nastavení a prevádzkových parametrov exportéru (angl. *Option Template Set*) a dátové sady (angl. *Data Set*).

Šablónové sady obsahujú definície **šablón**, ktoré sú dôležitým mechanizmom IPFIX správ. Šablóny popisujú štruktúru prenášaných dátových záznamov tokov alebo prevádzkových parametrov. K určeniu položiek, z ktorých je dátový záznam zložený, sa používajú spomínané informačné elementy, ktoré popisujú význam jednotlivých polí (napr. zdrojová IP adresa, počet prenesených bajtov, atď.). Príklad štruktúry šablóny je na obrázku 2.6.

⁵Prehľad aktuálne platných informačných elementov: www.iana.org/assignments/ipfix

⁶Zoznam registrovaných privátnych domén:
www.iana.org/assignments/enterprise-numbers/enterprise-numbers

Každá šablóna začína identifikačným číslom, ktoré je z intervalu hodnôt $\langle 256 - 65535 \rangle$, za ktorým sa nachádza počet položiek šablóny. Ďalej nasleduje usporiadaná sekvencia definícií položiek toku, ktoré sú primárne zložené z identifikačného čísla informačného elementu a maximálnej veľkosti daného poľa. Ak pochádza informačný element z privátneho rozsahu, najvyšší bit identifikátora elementu je nastavený na jednotku a definícia je rozšírená o číslo PEN.

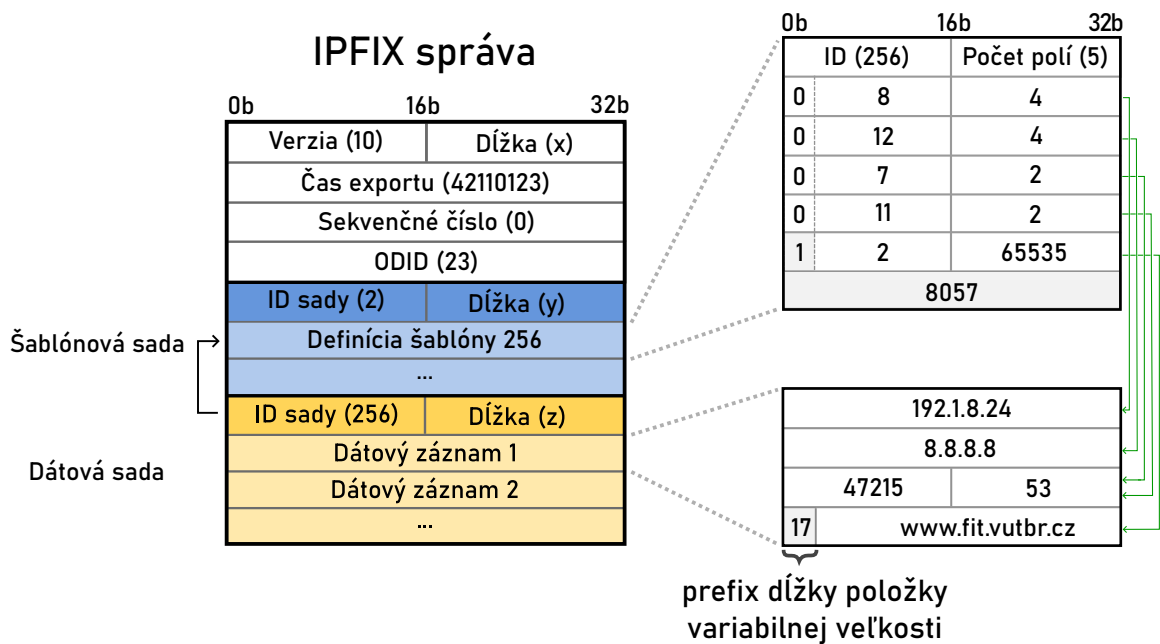


Obr. 2.6: Štruktúra IPFIX šablóny

Pozorný čitateľ si mohol všimnúť, že samotný informačný element definuje dátový typ, ktorým udáva veľkosť sledovanej vlastnosti. Prečo sa teda nachádza veľkosť aj v položke šablóny? Prvým dôvodom je fakt, že exportér nemusí striktno použiť veľkosť dátového typu a môže sa rozhodnúť zakódovať dáta v menšej veľkosti, ktorá je uložená do položky šablóny. Druhý dôvod súvisí so zakódovaním textových reťazcov, ktoré majú **premenlivú dĺžku**. Pevne definovaná veľkosť by predstavovala ich skrátenie alebo v opačnom prípade plytvanie pamäťou. Z toho dôvodu majú položky s variabilnou dĺžkou zakódovanú ich skutočnú veľkosť v prefixe dátového záznamu a veľkosť v definícii šablóny obsahuje rezervovanú hodnotu 65535, ktorá je vyhradená pre položky s dynamickou veľkosťou.

Sady šablón sa delia na dve skupiny. Prvou z nich sú vyššie spomenuté *sady šablón záznamov o tokoch*, ktoré výhradne popisujú monitorované toky. Takýto typ sady je definovaný identifikačným číslom 2 v hlavičke sady. Druhým typom sú *sady šablón nastavení a prevádzkových parametrov exportéru*, ktoré popisujú parametre monitorovania a sú identifikované číslom 3 v hlavičke sady. Šablóny tohto typu sú navyše rozšírené o definíciu políček určujúcich kontext záznamu.

Dátová sada je zložená z dátových záznamov obsahujúcich položky tokov. Identifikačné číslo dátovej sady nadobúda hodnoty od 256 a spája sadu so šablónou, ktorá ju popisuje. Dátové záznamy môžu byť statické, ak obsahujú výhradne položky s fixne danou veľkosťou, alebo dynamické, ak obsahujú aspoň jednu položku s premenlivou veľkosťou. Práca s dynamickými záznamami môže prinášať rôzne úskalia, keďže dĺžku takého záznamu nie je možné určiť z definície šablóny. Tieto okolnosti často vedú k potrebe manuálnej iterácie všetkých položiek, s cieľom určenia začiatku nového záznamu. Položky dynamických záznamov s premenlivou veľkosťou obsahujú prefix ich skutočnej dĺžky, ktorý je jedno-bajtový v prípade, že je jeho veľkosť menšia ako hodnota 255. V opačnom prípade je hodnota prvého bajtu 0xFF a veľkosť položky je zakódovaná v nasledujúcich dvoch bajtoch (maximálna veľkosť je teda 65535). Príklad dátových záznamov a ich prepojenia so šablónami je popísaný na obrázku 2.7.



Obr. 2.7: Príklad IPFIX správy s prepojením dátového záznamu na šablónu

2.4 Popis práce kolektora

Poslednou sekciovou časťou tejto kapitoly je priblíženie práce kolektora, ktorá začína nadviazaním spojenia s niekoľkými exportérmi a pokračuje prijímaním generovaných správ [1]. Pri odosielaní tokov z exportéru sa za účelom zníženia množstva prenesených dát nemusia v každej IPFIX správe vyskytovať definície šablón. Spravidla sú odosielané ihneď po vytvorení spojenia s kolektorom. Následujúce správy môžu obsahovať dátové sady, ktoré sa odkazujú na tieto šablóny. Tento fakt udáva jednu z činností kolektora, ktorou je udržiavanie odkazov na doposiaľ prijaté šablóny. Okrem definícií šablón môžu správy obsahovať aj záznamy o zrušení šablón (angl. *Template Withdrawal*) alebo o ich aktualizácii, na ktoré musí byť kolektor schopný reagovať.

Vplyv transportných protokolov na správu šablón

Udržiavanie odkazov na šablóny úzko súvisí s typom transportného protokolu použitého k prenosu správ z exportéru na kolektor. Najjednoduchším prípadom je použitie protokolu **TCP**, ktorý zabezpečuje spoľahlivé doručenie správ v správnom poradí. Pre protokoly **SCTP** alebo **UDP** je štandardom [1] mierne upravený proces odosiadania IPFIX správ. Protokol **SCTP**, ktorý spája vlastnosti **TCP** a **UDP**, kombinuje prístupy zaručeného a nezaručeného odosiadania správ. Z toho vyplýva podmienka, že všetky IPFIX správy obsahujúce definíciu alebo odstránenie šablóny, musia byť spoľahlivo odoslané so zachovaním poradia doručenia. Správy obsahujúce dátové záznamy môžu, no nemusia byť odosiadané týmto spôsobom.

V prípade **UDP** je situácia zložitejšia, keďže doručenie správ nie je garantované. Prvým problémom spojeným s týmto protokolom je možná strata paketov obsahujúcich definície šablón. V takom prípade by kolektor nebol schopný interpretovať prichádzajúce dátové záznamy, čo by predstavovalo výpadok informácií o stave siete. Čiastočné riešenie sa ponúka zo strany exportéru v podobe pravidelného odosiadania používaných šablón v určitých časo-

vých intervaloch. Periodické odosielanie použitých šablón tak zvyšuje šance ich úspešného doručenia na kolektor. Ďalším problémom je možné doručenie správ v nesprávnom poradí, kedy oneskorené správy s dátovými záznamami môžu používať šablóny, ktoré boli medzičasom redefinované. V takom prípade by mohol kolektor mylne interpretovať dátové záznamy, čo by následne mohlo viesť ku chybám vo výsledných dát.

Kolektor musí byť ďalej schopný rozlíšiť pôvod získaných záznamov, ktorý je spojený s **kontextom platnosti šablóny** [7]. Keďže zvyčajne sú správy na kolektor odosielané z rôznych exportérov, je zrejme, že každý z nich môže definovať rôzne šablóny popisujúce monitorované vlastnosti. Rovnaké identifikačné čísla šablón však môžu byť súčasne využívané viacerými exportérmi odosielajúcimi správy na kolektor v rovnakom čase. Ak exportér navyše obsahuje viacero pozorovacích domén, unikátnosť identifikátoru šablóny je zabezpečená iba v rámci domény, z ktorej pochádza. Kolektor musí s týmito situáciami počítať aby bol schopný úspešne interpretovať správy prijaté z rôznych exportérov, alebo z rôznych pozorovacích oblastí jedného exportéru. Z toho vyplýva, že kontext platnosti šablóny je viazaný na kombináciu transportného spojenia s daným exportérom a identifikátorom pozorovacej domény (ODID) šablóny.

Správa informačných elementov

Pre dekódovanie dátových záznamov sa vyžaduje znalosť príslušných informačných elementov, ktoré nie sú prenášané použitým protokolom. V prípade, že kolektor nemá k dispozícii tieto definície, nedokáže pracovať s dátovými záznamami, preto ich následne ignoruje alebo zahodí [1]. Z toho dôvodu je od kolektora prinajmenšom očakávaná znalosť elementov zo spoločného rozsahu. V ideálnom prípade by však mal podporovať možnosť dynamického doplnenia databázy informačných elementov o definície prvkov z privátnych rozsahov.

Práca s dátovými záznamami

Kolektor má po úspešnom prijatí a interpretácii správ možnosť pracovať s dátami, ktoré z nich boli získané. Medzi často využívané funkcie pri práci s prichádzajúcimi tokmi patrí ich **filtrovanie**. V závislosti od nasadenia kolektoru môže byť žiadúce, aby spracovával iba určitú časť záznamov spĺňajúcich podmienku danú filtračným výrazom. Filtrovaním tak dochádza k zníženiu záťaže potrebnej na spracovanie správ a zároveň k umožneniu rozdelenia monitorovania väčších sietí na menšie úseky. Ak kolektor navyše prijíma správy z viacerých exportérov, môže získané dáta ďalej agregovať, čím znižuje nároky potrebné na ich uskladnenie. **Profilovanie** tokov je nemenej využívaným prostriedkom, ktorý klasifikuje prijaté toky na základe vopred definovaných kritérií. Najčastejším prípadom profilovania je kategorizácia tokov podľa použitého protokolu, čo poskytuje správcovi informáciu o najpoužívanejších službách v sieti. Nezvyčajný nárast výskytu služby tak môže predstavovať bezpečnostné riziko, napríklad zvýšenie počtu TCP paketov s príznakom SYN je najčastejším indikátorom SYN-flood⁶ útoku [16].

Niektoré analyticky zaujímavé atribúty popisujúce toky nemusia byť nevyhnutne súčasťou exportovaných správ, ako bolo spomenuté v úvodnej kapitole. Jedným z dôvodov ich absencie môžu byť napríklad limitované prostriedky exportéru alebo veľké množstvo času spojené s ich získaním. V prípade, že je exportér nasadený na hlavných internetových linkách, musí byť každú sekundu schopný spracovať až státisíce paketov. Z toho dôvodu by

⁶SYN-flood alebo záplava SYN paketov je forma DoS (*Denial of Service*) útoku, pri ktorej útočník čiastočne nadviaže spojenie s cieľovým zariadením a neumožňuje jeho využívanie inými užívateľmi

získavanie informácií z tretích strán predstavovalo časovo kritické úseky práce exportéru, ktoré by mali dopad na jeho výkonnosť. Obohacovanie tokov preto často spadá na kolektory, ktoré pracujú už s vopred agregovanými dátami. Súčasným trendom v tejto oblasti je navyše použitie distribuovanej architektúry [17], ktorá je založená na rozložení spracovania tokov medzi pracovné uzly (skupinu kolektorov), čo umožňuje zvýšenie škálovateľnosti architektúry pri monitorovaní rozsiahlejších sietí.

Formát výstupných dát

Modifikované záznamy sú následne odosielané v rôznych formátoch na úložisko, alebo sú poskytované ďalším aplikáciám k analýze. Formáty podporované kolektorom závisia od jeho implementácie a zároveň aj od aplikácií pracujúcich nad týmito dátami. Príklady konkrétnych kolektorov a nimi podporovaných formátov sú popísané v práci: *Practical Experience with IPFIX Flow Collectors* [14]. Rada súčasných kolektorov podporuje export záznamov v databázových formátoch ako je MySQL alebo FastBit, ktoré je možné vidieť pri kolektoroch *nProbe* alebo *Vermont*. Ďalšou populárnou možnosťou je použitie binárnych súborov, pri ktorých si väčšina kolektorov definuje vlastný spôsob ukladania dát. Príkladom kolektorov využívajúcich binárne súbory sú *SiLK*, *nfcapd* alebo *IPFIXcol2* (využívajúci binárny formát FDS). Väčšina kolektorov podporuje okrem spomínaných spôsobov aj možnosť uloženia záznamov vo formátoch JSON alebo v textovej podobe.

Analýza dát

Existuje veľké množstvo aplikácií poskytujúcich analýzu získaných informácií. Jednou z najznámejších je sada nástrojov *nfdump*⁷, ktorá obsahuje softvérový exportér a kolektor spolu s rozšíreniami ako je filtrovanie, profilovanie a obohacovanie tokov. Nevýhodou práce s týmto nástrojom je jeho nastavenie a limitovaná vizualizácia dát, ktoré sú realizované prostredníctvom príkazového riadku. Tieto nevýhody sú však kompenzované nadstavbami, ktoré umožňujú užívateľom získané informácie integrovať s inými aplikáciami poskytujúcimi vizualizáciu dát. Pokročilejšími nástrojmi sú *NfSen*⁸, ktorý poskytuje základné webové rozhranie pre grafické zobrazovanie informácií o tokoch alebo *NEMEA*⁹ predstavujúca ľahko rozšíriteľný nástroj pre určovanie rôznych druhov útokov. Nástroje ako *ntop*¹⁰ alebo *Flowmon*¹¹, ktoré spájajú prvky detekcie útokov, správy sietí a pokročilej vizualizácie, sa vyskytujú v popredí nástrojov používaných na správu sietí.

⁷Stránky nástroja nfdump: github.com/phaag/nfdump

⁸Stránky nástroja NfSen: nfsen.sourceforge.net

⁹Stránky nástroja NEMEA: github.com/CESNET/Nemea

¹⁰Stránky nástroja ntop: ntop.org

¹¹Stránky nástroja Flowmon: flowmon.com

Kapitola 3

Analýza kolektoru IPFIXcol2

Kolektorov určených k zberu monitorovacích dát je v dnešnej dobe veľké množstvo, pričom každý z nich poskytuje iné možnosti práce s dátami. Jedným z verejne vyvíjaných je kolektor IPFIXcol¹, vytvorený tímom liberouter² patriacim pod združenie CESNET. Počiatky vývoja tohto kolektoru siahajú do roku 2012, teda do obdobia vzniku jeho prvej verzie [15]. V priebehu jeho používania sa však vyskytli problémy spojené s jeho návrhom a udržiavaním, ktoré vyústili do vytvorenia novej verzie kolektoru pod názvom IPFIXcol2³ v roku 2017, ktorému sa venujú nasledujúce časti tejto práce. Cieľom nasledujúcej kapitoly je priblíženie jednotlivých častí kolektoru IPFIXcol2, počínajúc popisom jeho architektúry, pokračujúc prehľadom jadra kolektoru a správy šablón, končiac sekciou o používaných moduloch. Poznatky o tomto kolektore vychádzajú z diplomovej práce jedného z jeho hlavných vývojárov Ing. Lukáša Hutáka s názvom *Nová generace IPFIX kolektoru* [7].

3.1 Prehľad architektúry

Počas tvorby kolektoru IPFIXcol2 bola zvláštna pozornosť venovaná jeho vlastnostiam, ako sú flexibilita a vysoká rýchlosť spracovania. Táto zvýšená pozornosť sa premietla v návrhu jeho architektúry. Jej základným prvkom je **zreťazená linka**, po ktorej prijaté správy putujú cez kolektor. Na tejto zreťazenej linke sa nachádzajú **moduly** vykonávajúce hlavné funkcie kolektoru, ktorými sú nadviazanie spojenia s exportérom, prijatie správ, správa šablón, modifikácia záznamov a pod. Modularita poskytuje užívateľom možnosť nastavenia kolektoru podľa ich potrieb, zároveň umožňuje tieto nastavenia kedykoľvek zmeniť. K popisu konfigurácie sa používa jazyk XML, pomocou ktorého je súčasne možné pridávať definície informačných elementov. Práca s modulmi ďalej ponúka možnosti zvyšovania výkonnosti pomocou **paralelizácie**, pretože je práca každej inštancie modulu vykonávaná v separátných vláknách.

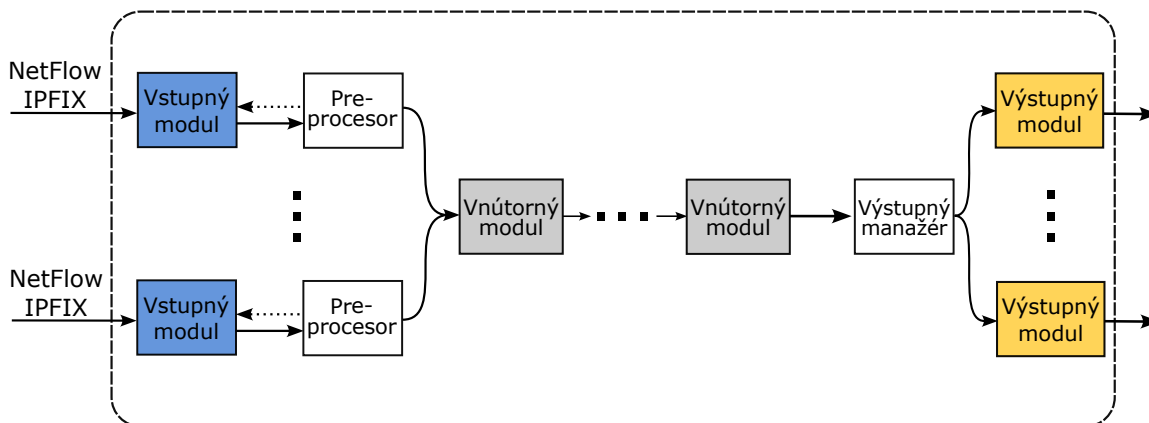
Architektúra kolektoru, zobrazená na obrázku 3.1, definuje tri druhy modulov: vstupné, vnútorné a výstupné. Úlohou vstupných modulov je získavanie IPFIX správ, ktoré môže byť realizované komunikáciou s exportérom pomocou príslušného protokolu alebo iného zdroja dát (napr. súborov). Každý vstupný modul je spojený s preprocesorom, ktorý je bližšie popísaný v sekcii 3.2. Pre začiatok stačí uviesť, že preprocesor slúži na kontrolu prichádzajúcich správ a správu prijatých šablón. Správy sú prenášané zreťazenu linkou ku

¹Odkaz na stránky kolektoru IPFIXcol: liberouter.org/technologies/ipfixcol

²Odkaz na stránky tímu liberouter: liberouter.org

³Odkaz na stránky kolektoru IPFIXcol2: github.com/CESNET/ipfixcol2

vnútorným modulom realizujúcim činnosti spojené s úpravou záznamov o tokoch, akými sú napríklad anonymizácia, filtrovanie a profilovanie. Ďalej smerujú na výstupný manažér, ktorý preposiela ich kópie jednotlivým výstupným modulom. Výstupné moduly konvertujú prijaté správy do príslušného formátu a odosielajú na úložisko, alebo ich poskytujú ďalším aplikáciám. Z popisu architektúry vyplýva, že zásuvné moduly sa špecializujú na určité časti práce kolektoru, zatiaľ čo správu týchto modulov, do ktorej patrí ich prepojenie a vzájomná komunikácia, realizuje **jadro** kolektoru.



Obr. 3.1: Architektúra kolektoru IPFIXcol2 [7]

3.2 Popis činnosti jadra kolektoru

Medzi základné činnosti jadra patrí konfigurácia kolektoru, vytváranie inštancií jednotlivých modulov a zabezpečovanie komunikácie medzi nimi, ktorá je realizovaná prenášaním správ pomocou zreťazenej linky. V rámci tejto komunikácie existujú štyri typy správ, ktoré sa v kolektore vyskytujú.

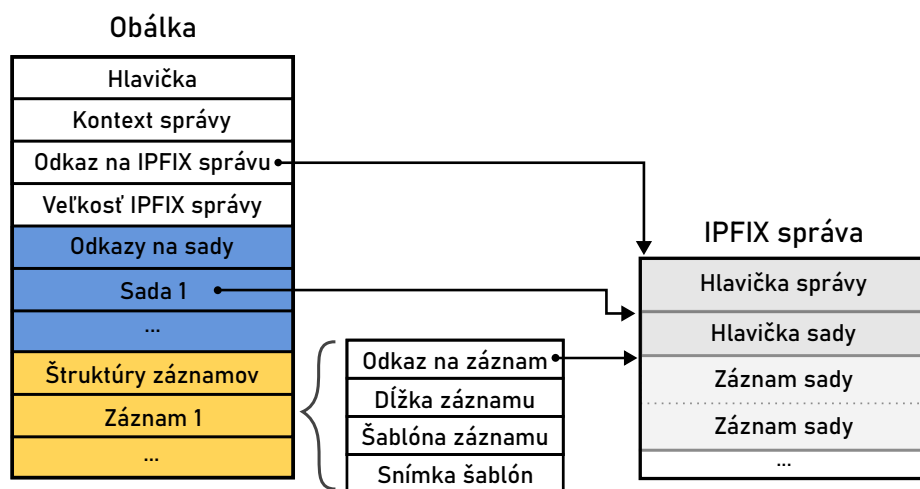
Dátová správa

Najčastejšie prenášanými správami v kolektore IPFIXcol2 sú dátové správy, ktorých úlohou je udržiavať informácie o surových Netflow/IPFIX správach. Tieto informácie sú uložené v takzvanej obálke, ktorú je možné vidieť na obrázku 3.2. Dátové správy vznikajú vo vstupných moduloch, ktoré do obálky vložia odkaz na surovú správu spolu s jej dĺžkou a kontextom. Kontext správy obsahuje údaje o spojení s exportérom a ODID, z ktorého informácie o tokoch pochádzajú. Na základe znalosti pôvodu správy je možné určiť šablóny definované týmto zdrojom, ktoré sú následne použité na interpretáciu dátových záznamov. Vytvorená dátová správa je odoslaná preprocesoru na ďalšie spracovanie.

Činnosť **preprocesora** začína určením verzie použitého protokolu z prijatej správy (NetFlow v5/v9, IPFIX). Pre správy typu Netflow navyše prebieha aj konvertovanie do IPFIX podoby, čo umožňuje kolektoru spracovávať správy v rôznych formátoch univerzálnym spôsobom. Preprocesor následne identifikuje všetky sady vyskytujúce sa v IPFIX správe. Odkazy na začiatky týchto sád sú doplnené do obálky dátovej správy. V prípade, že preprocesor narazí na šablónovú sadu, extrahuje z nej definície šablón. Udržiavanie doposiaľ videných šablón prebieha v manažéri šablón, ktorý je bližšie popísaný v sekcii 3.3. V závislosti od typu definície šablóny (vytvorenie, odstránenie, redefinícia) je aktualizovaný stav

manažera. Pri dátových sadách je pre každý záznam vytvorená položka v obálke dátovej správy obsahujúca informácie o danom zázname. Tieto položky (taktiež zobrazené na obrázku 3.2) predstavujú kľúčovú časť práce s dátovými správami. Patria k nim:

- *ukazovateľ na pozíciu záznamu v surovej správe* – používa sa na zistenie hodnoty položiek dátového záznamu
- *skutočná dĺžka záznamu* – v prípade pevnej dĺžky záznamu je zistená z definície šablóny, v opačnom prípade preprocesor odvodí dĺžku z prefixov dynamických záznamov
- *odkaz na šablónu záznamu* – šablóna popisujúca záznam je uložená v manažeri šablón
- *snímka šablón* – stav manažera šablón v čase prijatia šablóny (popísaný v sekcii 3.3)



Obr. 3.2: Obálka správy spolu so štruktúrou dátového záznamu

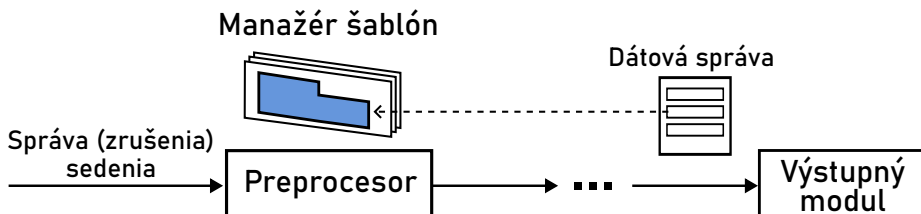
Význam ukladania týchto informácií spočíva v zjednodušení manipulácie s IPFIX správami počas práce vykonávanej ďalšími modulmi kolektoru. Väčšina informácií, ktoré moduly potrebujú ku svojej činnosti, sa nachádza priamo v obálke dátovej správy, a preto je potreba siahnúť po surovej správe minimálna. Navyše tým dochádza ku zlepšeniu výkonu kolektoru, keďže iterácia cez položky surovej správy prebieha jednorázovo v preprocesore a nie separátne v každom module.

Správa sedenia

Ďalším druhom správ vyskytujúcich sa v kolektore IPFIXcol2 sú správy sedenia (angl. *session message*), ktorých úlohou je poskytovať informácie o zmene stavu transportného spojenia s exportérom. Štruktúra tejto správy je zložená z identifikátoru sedenia a typu udalosti, ktorým môže byť nadviazanie alebo ukončenie spojenia. Správy sedenia vznikajú primárne vo vstupných moduloch, zároveň však môžu byť vytvorené počas činnosti preprocesoru. V prípade, že preprocesor detekuje chybu pri spracovaní prijatej správy, môže uzatvoriť spojenie, z ktorého správa pochádza, pričom je zároveň vygenerovaná nová správa sedenia popisujúca uzatvorenie tohto spojenia. Význam tohto typu správ je viditeľný pri správe šablón, kedy prijatie správy o ukončení sedenia preprocesorom značí, že šablóny, ktoré boli definované v tomto spojení už nebudú používané, a teda môžu byť odstránené.

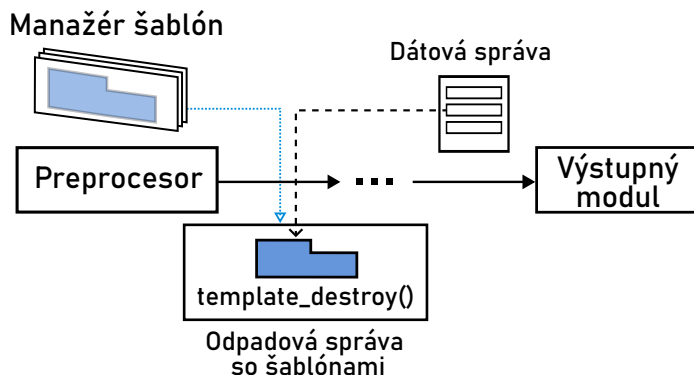
Odpadová správa

Nemenej využívaným typom správ sú odpadové správy (angl. *garbage message*), ktoré zabezpečujú korektné uvoľňovanie pamäte pre objekty spravované kolektorom. Ich využitie je najlepšie demonštrované na nasledujúcom príklade. Predstavme si situáciu zobrazenú na obrázku 3.3, kedy kolektor prijme správu o zrušení transportného sedenia s exportérom. Zároveň sa v zretazenej linke môže nachádzať prijatá, ale doposiaľ nespracovaná dátová správa obsahujúca záznamy, ktoré sa odkazujú na šablóny definované v tomto spojení. V takom prípade by preprocessor nemohol uvoľniť šablóny vo svojom manažéri, keďže sú ešte stále odkazované dátovou správou.



Obr. 3.3: Príklad dátovej správy obsahujúcej záznamy odkazujúce na aktívne šablóny v manažéri

Jedným z možných riešení je uložiť použité šablóny do odpadovej správy, ktorá je odoslaná do zretazenej linky. Odpadová správa, ktorá je zobrazená na obrázku 3.4, obsahuje odkaz na rušený objekt spolu s funkciou, ktorá sa použije na jeho odstránenie. Dôležitou vlastnosťou využívanou pri odpadovej správe je zachovanie poradia odosielaných správ v zretazenej linke kolektoru. Táto vlastnosť garantuje, že všetky dátové správy odkazujúce sa na rušené šablóny boli spracované skôr, než sa odpadová správa dostane na koniec zretazenej linky. Na základe týchto okolností je možné objekt odpadovej správy na výstupných moduloch jednoducho odstrániť.



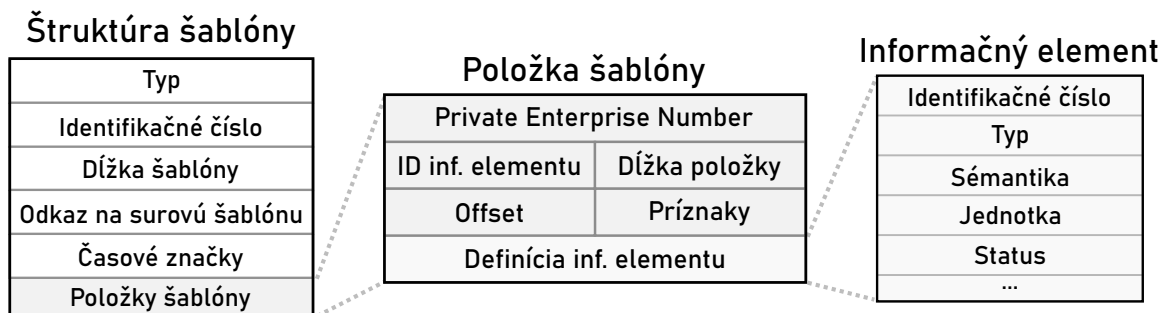
Obr. 3.4: Použitie odpadovej správy na odstránenie šablón

Terminačná správa

Posledným typom správy je tzv. terminačná správa, ktorá je de facto poslednou prechádzajúcou kolektorom. Prijatie terminačnej správy modulom oznamuje ukončenie činnosti kolektoru, ktoré je spojené s uvoľňovaním alokovaných zdrojov a ukončením práce modulov.

3.3 Správa šablón

Správa šablón je realizovaná preprocesorom, ktorý z prijatých správ extrahuje definície šablón. Informácie o šablónach definície následne ukladá do vnútornej štruktúry používanej v kolektore IPFIXcol2, ktorú je možné vidieť na obrázku 3.5. Štruktúra šablóny obsahuje základné informácie, ako napríklad jej typ (šablóna dátových záznamov alebo prevádzkových parametrov), ID šablóny, dĺžku a odkaz na surovú šablónu. Časové značky popisujú, kedy bola šablóna prvýkrát definovaná, kedy bola naposledy obnovená alebo, v prípade protokolu UDP, v akom čase dôjde ku vypršaní jej platnosti. Štruktúra je ukončená zoznamom položiek, ktoré šablóna definuje.



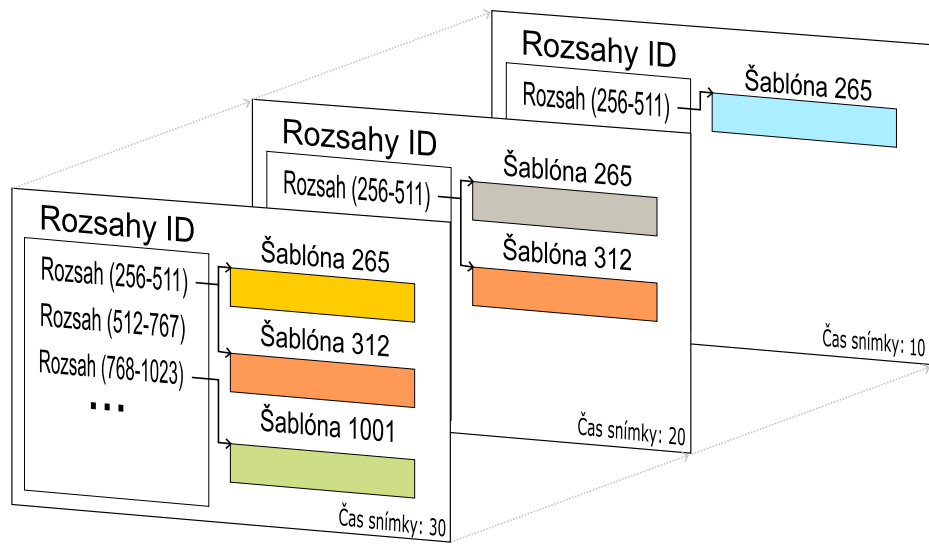
Obr. 3.5: Spôsob uloženia informácií o šablóne kolektoru IPFIXcol2

Štruktúra položky šablóny je okrem základných informácií definovaných štandardom (obrázok 2.6) obohatená o políčka *offsetu* alebo *príznamov* položky. Offset udáva polohu danej položky od začiatku šablóny a príznaky určujú dodatočné informácie (napr. viacnásobný alebo posledný výskyt, kľúčová položka, atď). Položka šablóny tiež zahrňuje odkaz na príslušný informačný element, ktorý ju popisuje. Informačné elementy sú v kolektore spravované manažérom informačných elementov, ktorý po spustení kolektoru získa definície elementov zo zvoleného konfiguračného súboru.

Šablóny sú uchovávané v **manažérovi šablón**, ktorý poskytuje fundamentálne operácie na prácu so šablónami, akými sú vkladanie, mazanie alebo vyhľadávanie na základe identifikačného čísla. Z dôvodu obmedzeného počtu definícií šablón, ktorý vychádza z pevne stanoveného rozsahu identifikačných čísel, je možné ich uloženie v dvojúrovňovej adresovacej tabuľke. Prvá úroveň tejto tabuľky rozdeľuje množinu všetkých identifikačných čísel na rovnako veľké intervaly, ktoré obsahujú v druhej úrovni zoznam definícií šablón. Tento spôsob uloženia je možné vidieť na obrázku 3.6. Kolektor IPFIXcol2 navyše adresuje problémy správy šablón spojené s ich redefinovaním alebo spracovaním oneskorených správ.

Pri zmene definície šablóny sa v zretazenej linke môžu nachádzať dátové správy, ktorých záznamy boli popísané pomocou pôvodnej šablóny. Druhým problémom je interpretácia oneskorených správ, ktorých záznamy sa môžu odkazovať na medziasom zmenené šablóny. Riešenie, ktorým manažér šablón disponuje, je prepájanie definícií šablón s časom ich vytvorenia. Tento proces je realizovaný pomocou **snímok šablón** (angl. *snapshots*). K ich vytváraniu dochádza pri zmene definície šablóny, pričom ostáva k dispozícii aj pôvodná šablóna, ktorá môže byť odkazovaná správami v zretazenej linke. Mechanizmus vytvárania snímok zároveň adresuje problém interpretácie oneskorených správ, kedy je na základe času exportu správy nájdená príslušná snímka obsahujúca pôvodnú šablónu. Odkaz na snímku sa prenáša spolu so šablónou v položke záznamu dátovej správy (obrázok 3.2). Príklady

redefinície šablóny (s číslom 265) s postupným vytváraním snímok je zachytený na obrázku 3.6.



Obr. 3.6: Snímky manažéru v priebehu procesu redefinície šablón

3.4 Zásuvné moduly

V súčasnej dobe majú správcovia sietí využívajúci IPFIXcol2 na výber z niekoľkých modulov, ktorých prehľad sa nachádza v tabuľke 3.1. Konfigurácia modulov prebieha pomocou konfiguračného XML súboru, ktorý je spracovaný pri spustení kolektoru. Po načítaní nastavení jednotlivých modulov dochádza k ich inštanciacii jadrom a kolektor začína spracovávať prichádzajúce IPFIX správy.

Typ modulu	Názov	Stručný popis
Vstupný	TCP, UDP	Komunikácia s exportérom v reálnom čase (online)
	FDS, IPFIX	Načítavanie dát zo súborov (offline)
Vnútorňý	anonymizačný profilovací filtrovací	Anonymizácia citlivých údajov, profilovanie a filtrovanie tokov
Výstupný	FDS, IPFIX	Ukladanie informácií o tokoch do súborov
	json	Viacero možností exportu správ (server, súbor, text)
	forwarder	Preposielanie správ ďalším kolektorom
	viewer, timecheck	Textové zobrazenie správ, kontrola čas. razítok
	lnfstore	Ukladanie do súborov vo formáte nástroja <i>nfdump</i>
	unirec	Export dát vo formáte nástroja <i>Nemea</i>

Tabuľka 3.1: Moduly kolektoru IPFIXcol2

Počas navrhovania kolektoru bol dôraz kladený na možnosť vytvárania nových modulov a ich jednoduchú integráciu v rámci kolektoru. Definícia nového modulu začína štruktúrou `ipx_plugin_info`, ktorá ho identifikuje spomedzi ostatných častí kolektoru. Každý modul musí implementovať minimálne tri základné funkcie. Prvou z nich je inicializačná funkcia (`ipx_plugin_init`). Jej úlohou je zabezpečiť vytvorenie privátnych dát (napr. databázového spojenia), s ktorými modul pracuje v priebehu svojej činnosti. Najdôležitejšou funkciou, ktorú musí modul implementovať, je funkcia zabezpečujúca jeho hlavnú činnosť (`ipx_plugin_process`), ktorou je práca s dátovými záznamami o tokoch. V rámci tejto funkcie sú realizované činnosti charakterizujúce daný modul. Poslednou povinnou funkciou je ukončenie činnosti modulu (`ipx_plugin_destroy`) a uvoľnenie všetkých zdrojov, ktoré ním boli alokované.

Kapitola 4

Návrh obohacovania IPFIX správ

Z popisu práce kolektoru v sekcii 2.4 vyplýva, že kolektor by mal okrem základnej činnosti spojenej s ukladaním záznamov podporovať pokročilejšie operácie, ktorými sú napríklad filtrovanie, profilovanie, alebo obohacovanie tokov. Tieto funkcie, ktoré boli do istej miery súčasťou pôvodného návrhu kolektoru IPFIXcol, však nebolo možné pri prechode na novú verziu aplikovať v pôvodnej implementácii. Na základe tejto skutočnosti je nutné prispôbiť ich novému návrhu, čomu sa venovala bakalárska práca Michala Sedláka [13] rozširujúca kolektor IPFIXcol2 o možnosti filtrovania a profilovania tokov. Podobné ciele si kladie aj táto práca, ktorá je zameraná na obohacovanie tokov dátami z externých zdrojov. Nový návrh obohacovania tokov je založený na rozšírení jadra kolektoru dvomi komponentmi. Prvým z nich je modifikátor, ktorý realizuje obohacovanie dátových záznamov a taktiež zabezpečuje správu ich šablón. Druhým komponentom jadra je tvorca nových dátových správ s modifikovanými záznamami, ktoré sú následne odosielané do zrefazenej linky. Jednoduché vytváranie modulov následne ponúka možnosť rozšírenia nového kolektoru o vnútorné moduly obohacujúce správy o čísla autonómnych systémov (ASN) a geografické informácie zdroja a cieľa toku (GeoIP).

Táto kapitola sa zaoberá popisom nového návrhu obohacovania správ kolektoru IPFIXcol2. V jej úvode sú popísané autonómne systémy a geolokácia spolu s ich významom pri analýze dát. Nasleduje stručné zhrnutie obohacovania správ pôvodnej verzie kolektoru obsahujúce zhodnotenie jeho pozitívnych a negatívnych stránok, ktoré sú adresované v novom návrhu. Ďalej je jej súčasťou samotné obohacovanie záznamov kolektoru realizované pomocou nových komponentov jeho jadra. V závere kapitoly sa nachádza stručná rekapitulácia činností nových vnútorných modulov realizujúcich úpravu záznamov prechádzajúcich kolektorom.

4.1 Autonómne systémy a geolokácia

Pred zhodnotením návrhu obohacovania správ pôvodnej verzie kolektoru je nutné stanoviť si základné informácie, ktorými budú záznamy rozšírené. Na úvod je vhodné inšpirovať sa jedným z modulov pôvodného kolektoru, ktorý obohacoval toky správ geolokačnými údajmi popisujúcimi zdroj a cieľ komunikácie. Ďalšou položkou obohacovania môžu byť čísla autonómnych systémov, pričom oba druhy týchto položiek predstavujú cenné informácie z pohľadu analýzy prevádzky na sieti.

Autonómne systémy (AS) sú definované ako skupina IP sietí pod správou jedného operátora s jasne definovanou smerovacou politikou [5]. Typicky sa AS spája s poskytovate-

lom internetových služieb (ISP) alebo inými organizáciami zabezpečujúcimi služby podobného charakteru. AS majú využitie najmä pri smerovaní paketov putujúcich cez internet a sú využívané smerovacími protokolmi, ktoré delíme na dve skupiny:

- **Interior gateway protocol (IGP)** – skupina protokolov zabezpečujúca smerovanie v rámci jedného AS (napr. OSPF, IS-IS, RIP)
- **Exterior gateway protocol (EGP)** – protokol BGP zabezpečujúci smerovanie medzi viacerými AS

AS sú identifikované pomocou unikátnych čísel, ktoré sú spravované organizáciou IANA¹. Analyzovanie informácií o tokoch spojených s autonómnymi systémami dáva správcovi sietí lepší prehľad o prúde toku dát cez sieť. Táto analýza sa môže následne využívať k zlepšovaniu kvality služieb poskytovaných klientom, napríklad pri zisťovaní príčin výpadkov siete alebo kritických úsekov infraštruktúry. Taktiež majú význam pri zvyšovaní bezpečnosti, keďže umožňujú odhadnúť potenciálne škodlivú komunikáciu na základe reputačných databáz AS, ktoré určujú dôveryhodnosť komunikácie smerujúcej z danej siete.

Podobné využitie má **IP geolokácia**, ktorá určuje geografický zdroj a cieľ toku na základe IP adres, ktoré sa v ňom nachádzajú [9]. Pri geolokácii sa môžu zisťovať rôzne atribúty pôvodu dát, pričom medzi najčastejšie vyhľadávané patria: kód krajiny/regiónu, názov mesta alebo GPS súradnice. Výsledky získané touto metódou nemusia byť vždy spoľahlivé a všeobecne platí, že pri zisťovaní polohy na úrovni súradníc sa presnosť výsledku znižuje. V takomto prípade súradnice nepopisujú presné miesto komunikujúceho zariadenia, ale iba jeho približnú polohu. Na základe tejto skutočnosti je v praxi často preferovaná lokalizácia tokov na úrovni krajiny alebo mesta. Výhodou geolokácie je možnosť zvýšenia bezpečnosti vo forme obmedzenia prístupu ku sieti pre zariadenia na základe ich polohy. Taktiež môžu organizácie využívajúce GeoIP zlepšovať kvalitu poskytovaných služieb, kedy v závislosti od počtu užívateľov pripojených z rovnakej oblasti môže napríklad dôjsť ku zavedeniu podporných lokálnych serverov poskytujúcich danú službu pre túto oblasť.

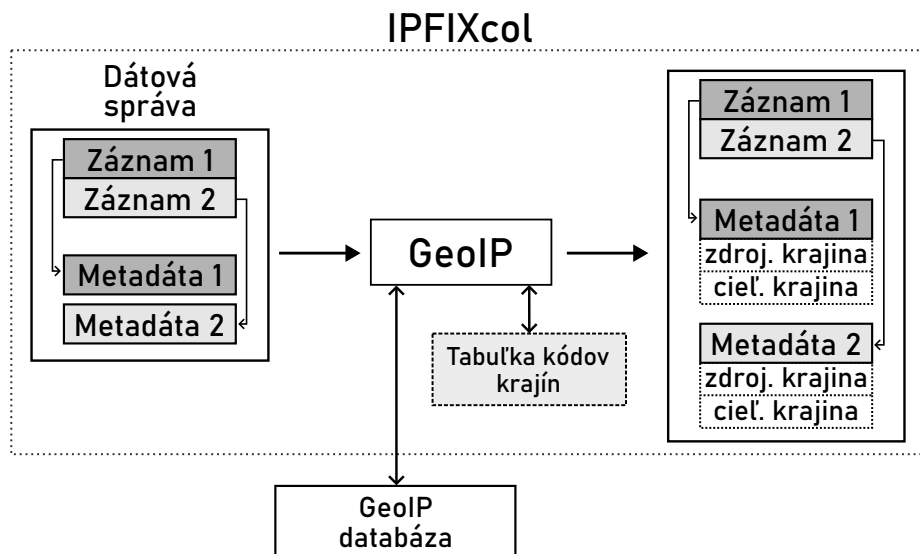
4.2 Zhodnotenie pôvodného návrhu

Pôvodný návrh kolektoru IPFIXcol sa čiastočne venoval aj problematike obohacovania tokov o dáta z externých zdrojov. Príkladom je GeoIP modul, ktorý pridával do dátových záznamov číselné kódy² krajín na základe zdrojovej a cieľovej IP adresy toku. Úskalia pôvodného návrhu súvisia so spôsobom, akým kolektor pracoval s prijatými IPFIX správami. Podobne ako pri novom kolektore sa aj v pôvodnej verzii využívajú dátové správy, ktorých dôležitým prvkom pre obohacovanie tokov je dodatočná štruktúra **metadát**.

V dátovej správe je každý záznam spojený s metadátami obsahujúcimi informácie spojené s autonómnymi systémami, geografickými údajmi alebo profilmi charakterizujúcimi daný záznam. Pre uvedenie príkladu práce s touto štruktúrou sa môžeme pozrieť na činnosť GeoIP modulu na obrázku 4.1. Jeho práca začína iteráciou cez záznamy dátovej správy. Pre každý záznam je nájdená zdrojová a cieľová IP adresa, ktoré sú následne použité pri vyhľadaní geolokačných informácií v databáze. Výsledok vyhľadávania (textový reťazec označenia krajiny, napr. CZ, SK, UK) je neskôr použitý pri vyhľadaní číselného kódu v tabuľke kódov krajín. Tento kód je vložený do štruktúry metadát pre práve obohacovaný záznam. Nevýhodou tohto prístupu je skutočnosť, že tabuľka kódov krajín je priamou súčasťou zdrojového kódu modulu, čo sťažuje jej prípadnú aktualizáciu.

¹ Čísla registrovaných autonómných systémov: iana.org/assignments/as-numbers/as-numbers.xhtml

² Trojciferné čísla identifikujúce krajinu podľa štandardu ISO 3166-1



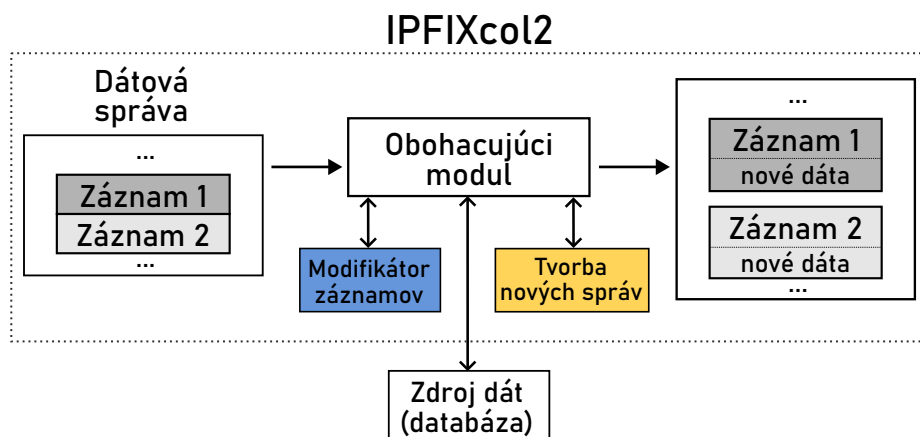
Obr. 4.1: Obohacovanie správ modulom GeoIP v kolektore IPFIXcol

Pri implementácii obohacovania správ nového kolektoru možno využiť rovnaký návrh rozšírenia dátovej správy o štruktúru metadát. Tento prístup má viacero nevýhod, pričom najhlavnejšou z nich je nízka flexibilita pridaných informácií. Ďalšou z nevýhod je fakt, že pridané informácie nie sú súčasťou surovej správy, a teda moduly kolektoru musia s týmito dátami pracovať separátne. Z daných skutočností vyplýva nepriaznivý dôsledok spočívajúci v úprave implementácie ostatných častí kolektoru pracujúcich s touto štruktúrou pri pridaní nových obohacujúcich položiek. Navyše, ak by kolektor neobsahoval aktívny ASN alebo GeoIP modul, štruktúra metadát by ostala nevyplnená, čo by museli vedieť rozpoznať výstupné moduly. Z tohto dôvodu by musel kolektor obsahovať mechanizmus označujúci platné a neplatné položky štruktúry metadát, čo by spôsobovalo zbytočné komplikácie pri pridávaní nových položiek tokov. Obohacovanie tokov hodnotami čísel AS je navyše v predošlej verzii kolektoru úplne vynechané.

4.3 Nový návrh obohacovania správ

Na základe negatívnych aspektov starého návrhu je potrebné vyvinúť nový spôsob, ktorým bude kolektor IPFIXcol2 schopný toky obohacovať. Základnou myšlienkou nového návrhu je pridávanie obohacujúcich dát priamo do surovej správy. Tento spôsob zahŕňa úpravu záznamov dátových správ, ale taktiež šablón popisujúcich tieto záznamy. Medzi výhody tohto návrhu patrí zvýšená flexibilita, ktorá umožňuje obohacovanie tokov o ľubovoľné informácie bez nutnosti úpravy ostatných častí kolektoru. Jednu z nevýhod tohto návrhu však predstavuje značná komplikovanosť úkonov spojených s udržiavaním modifikovaných šablón, čo môže mať negatívny dopad na výkonnosť kolektoru. Z toho dôvodu sú súčasťou kapitoly 5 testy hodnotiace dopady obohacovania správ na výkonnosť kolektoru.

Postup obohacovania správ zobrazený na obrázku 4.2 je založený na dvoch nových komponentoch jadra kolektoru. Prvým z nich je **modifikátor záznamov**, ktorý poskytuje obohacujúcemu modulu funkcie na úpravu dátových záznamov a šablón. Výsledkom činnosti modifikátoru sú obohatené záznamy, ktoré sú uložené v dátovej správe vytvorenej **tvorcom nových správ**. Celú činnosť obohacovania zastrešuje vnútorný **obohacujúci**



Obr. 4.2: Nový návrh obohacovania správ v kolektore IPFIXcol2

modul, ktorý prepája tieto novovytvorené komponenty a poskytuje informácie potrebné k obohateniu záznamov. Následne sú upravené správy odosielané namiesto pôvodných neobohatených do zretazenej linky, cez ktorú putujú až do výstupných modulov kolektoru.

Modifikácia dátových záznamov

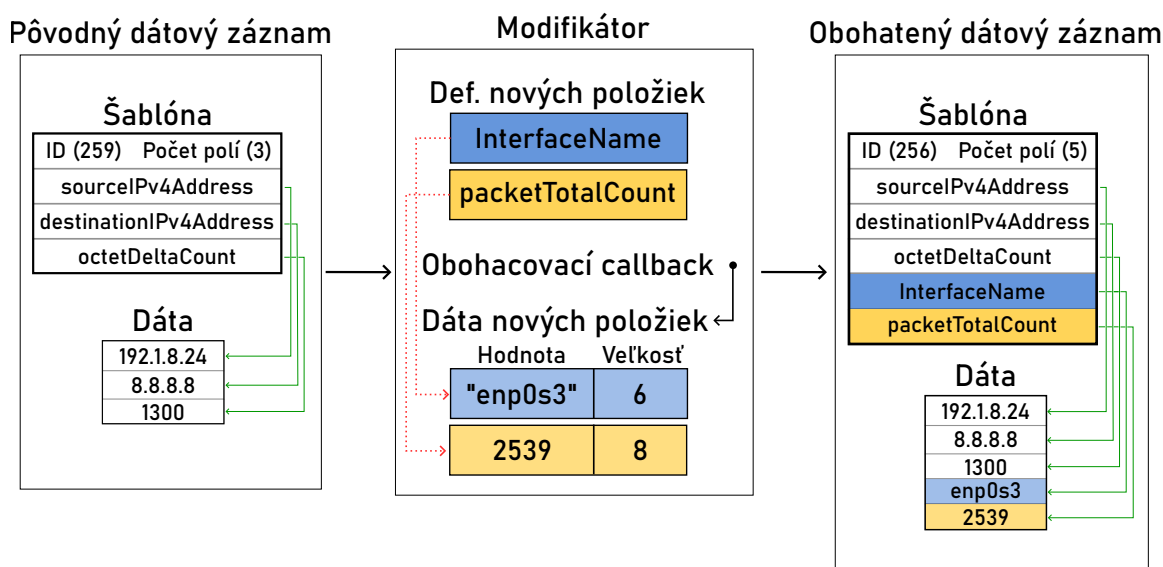
Vnútorne moduly upravujúce dátové záznamy nad nimi typicky realizujú dve činnosti. Prvou z nich je pridávanie nových položiek, pri ktorom musí vnútorný modul tieto položky definovať a zároveň poskytnúť modifikátoru funkciu na získavanie ich hodnôt. Nové položky sú následne pridané do každého dátového záznamu prechádzajúcim vnútorným modulom. Druhou činnosťou je filtrovanie vybraných položiek vyskytujúcich sa v prijatých tokoch, ktoré má za následok ich odobratie zo záznamu. Pri tejto činnosti musí vnútorný modul predať modifikátoru informáciu o pozícií týchto položiek určených na filtrovanie. Modifikátor následne zastrešuje prácu spojenú so samotným obohacovaním a filtrovaním položiek tokov.

Nový komponent modifikátoru pracuje na úrovni jednotlivých dátových záznamov získaných z prijatých správ. Základom jeho činnosti je definovanie obohacovacej a filtrovacej funkcie, ktoré sú využívané v priebehu jeho činnosti ako tzv. *callback* funkcie. Tieto funkcie sú implementované vnútornými modulmi využívajúcimi modifikátor k úprave dátových záznamov. Obohacovacia funkcia slúži k získavaniu hodnôt nových položiek pre každý spracovávaný záznam toku. Okrem nej však musí vnútorný modul poskytnúť modifikátoru taktiež definície nových položiek, ktoré majú byť do záznamu pridané. Ich formát je totožný s definíciou položky šablóny štandardu IPFIX 7011 (obrázok 2.6). V prípade filtrovania je proces modifikácie jednoduchší a realizuje sa pomocou filtrovacej funkcie, ktorá v rámci šablóny definuje pozície položiek určených na odstránenie.

Na obrázku 4.3 je zobrazený ilustračný príklad obohacovania zjednodušeného dátového záznamu položkami `InterfaceName` a `packetTotalCount`. Prvým krokom v procese obohacovania záznamu je rozšírenie šablóny novými položkami realizované modifikátorom po prijatí dátového záznamu. Následne je na základe obohacovacej funkcie naplnená vnútorná štruktúra modifikátoru obsahujúca získané hodnoty nových položiek. Pre každú novú položku je okrem hodnoty získaná aj jej veľkosť, ktorá zároveň určuje jej použitie v upravenom toku nasledovným spôsobom:

1. veľkosť položky ≥ 0 : hodnota položky je platná a použije sa pri obohatení toku
2. veľkosť položky < 0 : hodnota položky je neplatná (napr. v prípade, že ju z dôvodu absencie záznamu v databáze nebolo možné určiť) a použije sa typická hodnota reprezentujúca prázdnu položku (v prípade číselného typu položky hodnota nula, v prípade položiek s variabilnou dĺžkou prázdne pole)
3. veľkosť položky = rezervovaná hodnota: položka nie je súčasťou upraveného záznamu (napr. z dôvodu existencie danej položky v aktuálne obohatenom toku)

Hodnoty nových položiek sú modifikátorom následne vložené z vnútornej štruktúry do nového dátového záznamu.

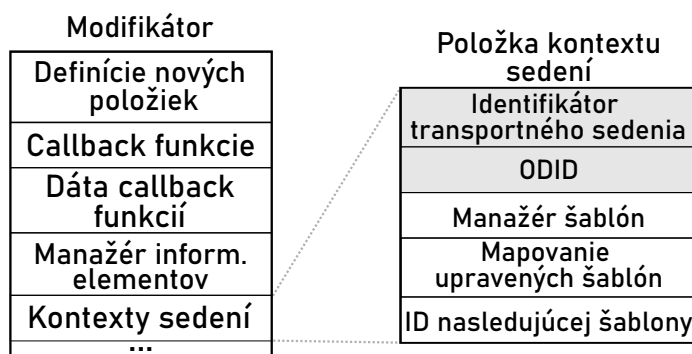


Obr. 4.3: Príklad obohatenia dátového záznamu

Typicky je v kolektore IPFIXcol2 za správu šablón zodpovedný preprocesor, ktorý spravová prijaté IPFIX správy. Je potrebné zdôrazniť, že so zmenou šablón popisujúcich dátové záznamy sa zodpovednosť ich správy presúva na modifikátor. Šablóny sú v modifikátore po ich úprave uložené v manažéri šablón príslušného **kontextu sedenia**. Potreba prepomenia šablóny na kontext sedenia vyplýva zo skutočnosti, že v rámci jednotlivých spojení môžu byť odlišné šablóny identifikované rovnakým číslom. Kontext sedenia, ktorý je zobrazený na obrázku 4.4, slúži na správu upravených šablón odosielaných z rovnakej pozorovacej domény. Kľúčovou položkou kontextu sedenia je dvojica identifikátoru transportného sedenia s exportérom a ODID, ktorá jednoznačne identifikuje zdroj pôvodnej šablóny. Na prvý pohľad sa môže zdať, že ku vytvoreniu kontextu sedenia dochádza bezprostredne po ustanovení nového spojenia s exportérom. Tento proces je však zrealizovaný až po prijatí prvej správy z novej pozorovacej domény daného exportéru.

Kontext sedenia obsahuje okrem manažéra šablón aj ID nasledujúcej šablóny, ktoré sa používa pri generovaní sekvencie nových identifikačných čísel upravených šablón. Príčinou generovania nových identifikátorov šablón je skutočnosť, že záznamy popísané jednou šablónou sa môžu po obohatení namapovať na viacero odlišných záznamov v závislosti od toho, či sa pôvodný záznam podarilo obohatiť modulom o nové dáta, alebo či sa modifikátor rozhodol tieto položky ignorovať. Pri filtrovaní môže taktiež nastať podobná situácia, v prípade, že sa viacero pôvodných záznamov namapuje na rovnaký záznam, ak sú niektoré

položky zo šablóny odobrané. V zriedkavých situáciách môže byť počet nových šablón väčší ako je maximálna hodnota počítadla identifikátorov (65535). V takom prípade dochádza ku reštartovaniu kontextu sedenia, ktoré zahŕňa odstránenie všetkých definovaných šablón v manažéri spolu s nastavením ID nasledujúcej šablóny na predvolenú hodnotu (256). Obohacovanie tokov následne môže bezproblémovo pokračovať ďalej.

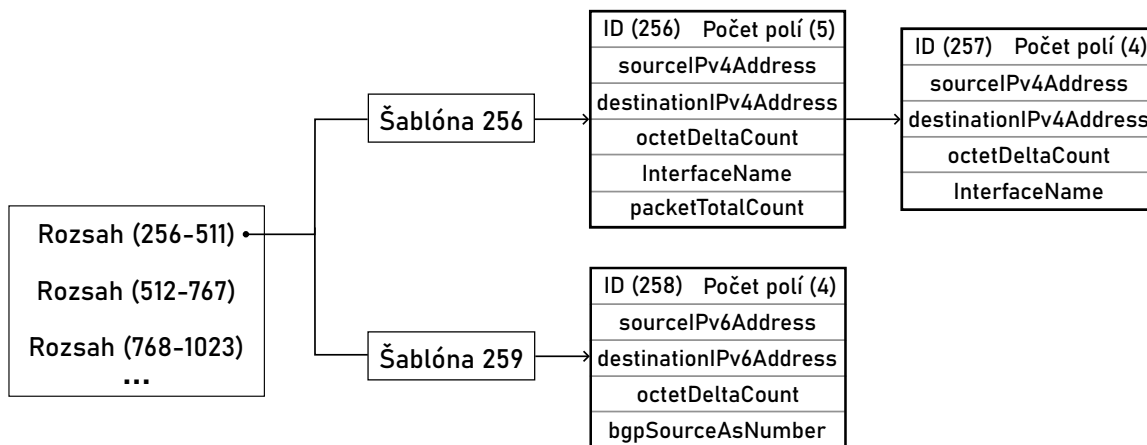


Obr. 4.4: Štruktúra kontextu sedenia modifikátoru tokov

Pred uložením modifikovanej šablóny do manažéra šablón je z dôvodu eliminácie duplikátneho ukladania dôležité skontrolovať, či sa konkrétna šablóna v manažéri už nevyskytuje. Riešenie, ktorým disponuje manažér šablón, je porovnávanie novej šablóny so šablónami uloženými v rámci aktuálnej snímky. V prípade veľkého počtu vygenerovaných šablón však môže sekvenčné prechádzanie snímky zvýšiť čas potrebný k nájdeniu modifikovanej šablóny, čo môže mať negatívny dopad na výkonnosť kolektoru. Kontext sedenia preto obsahuje štruktúru **mapovania upravených šablón** urýchľujúcu ich vyhľadávanie. Štruktúra mapy modifikovaných šablón, podobná snímke manažéra šablón (obrázok 3.6), je tvorená dvojúrovňovou adresovacou tabuľkou, pričom prvá vrstva rozdeľuje rozsah identifikátorov šablón na rovnako veľké intervaly. Druhá vrstva je indexovaná identifikačným číslom pôvodnej šablóny a obsahuje lineárne viazaný zoznam s definíciami modifikovaných šablón, ktoré vznikli zo šablóny s daným identifikátorom (obrázok 4.5). Dĺžka tohoto zoznamu sa odvíja od toho, na aký počet šablón sa môže potenciálne po obohatení namapovať pôvodná šablóna. Vo väčšine prípadov sa dĺžka zoznamu pohybuje rádovo v jednotkách položiek, čo významne napomáha zvyšovaniu rýchlosti procesu vyhľadávania modifikovanej šablóny.

Pri procese správy upravených šablón (ich uloženie v manažéri) sa komponent modifikátoru pokúsi o nájdenie rovnako definovanej šablóny v mape modifikovaných šablón. V prípade úspechu tak nemusí byť šablóna pracne vyhľadávaná v manažéri a modifikátor môže pracovať s odkazom získaným z položky mapy. V opačnom prípade je nutný sekvenčný prechod aktuálnej snímky z dôvodu možnej existencie rovnako upravenej šablóny vytvorenej zo šablóny s odlišným identifikačným číslom. V prípade, že na takúto šablónu modifikátor narazí, vytvorí v mape modifikovaných šablón na indexe pôvodnej šablóny novú položku, ktorá bude obsahovať odkaz na upravenú šablónu. Tento proces zaisťuje, že modifikovaná šablóna je následne uložená v manažéri šablón príslušného kontextu sedenia a môže byť použitá na popis obohateného záznamu. Na konci procesu obohatenia je upravený dátový záznam vrátený vnútornému modulu využívajúcemu modifikátor (ASN, GeoIP, ...).

Súčasťou správy šablón je aj reakcia na ukončenie spojenia s exportérom, pri ktorej dochádza k odstráneniu šablón definovaných daným exportérom z dôvodu nutnosti uvoľnenia pamäte potrebnej pre ich uloženie. V modifikátore dochádza po prijatí správy sedenia s príznakom ukončenia spojenia k odstráneniu príslušnej položky kontextu sedenia, ktorá je



Obr. 4.5: Mapovanie pôvodných šablón na modifikované

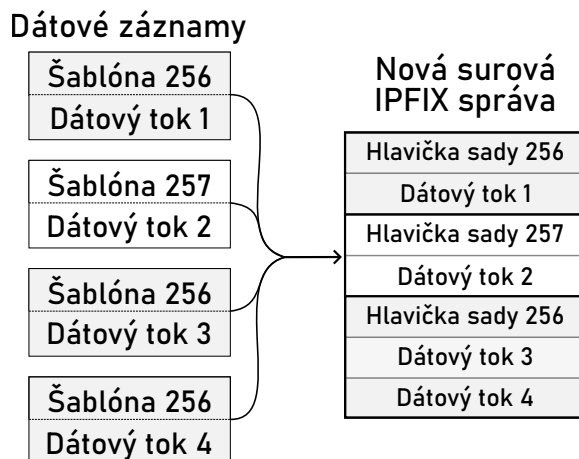
spätá s daným spojením. Zároveň sú všetky nové šablóny tohoto kontextu vložené do odpadovej správy a následne odoslané do zretazenej linky. Je dôležité podotknúť, že odstránenie položky kontextu nie je spojené s ODID, ale výhradne s identifikátorom transportného sedenia. Po ukončení spojenia s exportérom tak dochádza k odstráneniu všetkých s ním súvisiacich kontextov (nezávisle od ODID).

Vytváranie nových dátových správ

Po ukončení procesu modifikácie je nutné uložiť dátové záznamy do novovzniknutých dátových správ. Táto činnosť je realizovaná tvorcom dátových správ, ktorý je novým komponentom jadra kolektoru. Proces tvorby dátovej správy je založený na vytvorení surovej IPFIX správy s upravenými dátovými záznamami, nad ktorými následne vzniká obálka dátovej správy, ktorá je ďalej odoslaná do zretazenej linky kolektoru.

V priebehu procesu vytvárania novej surovej IPFIX správy dochádza k okopírovaniu hlavičky pôvodnej správy z dôvodu potreby zachovania sekvenčného čísla, času exportu a ODID, ktoré sú súčasťou pôvodnej správy. Tvorca nových správ ďalej poskytuje zastrešujúcemu modulu, t.j. vnútornému modulu implementujúcemu logiku obohacovania alebo filtrovania tokov (napr. ASN alebo GeoIP), funkcie na pridávanie modifikovaných dátových záznamov. V priebehu procesu pridávania záznamu je automaticky vytvorená nová dátová sada s identifikačným číslom šablóny záznamu. V prípade záznamov špecifikovaných rovnakou šablónou dochádza k ich uloženiu v rámci rovnakej dátovej sady, ako je ukázané na obrázku 4.6.

Samotné šablóny nie sú po pridaní dátového záznamu súčasťou novovytvorených správ, čím dochádza ku redukcii veľkosti surovej správy. Zároveň sa tu však vyskytuje otázka, či absencia šablón v správach nemá vplyv na fungovanie ostatných častí kolektoru. Pri práci ostatných častí kolektoru s dátovými záznamami sa využívajú výhradne spracované šablóny, ktoré sú odkazované zo záznamov v obálke dátovej správy. Moduly a časti jadra kolektoru (s výnimkou preprocesoru) tak neprístupujú priamo k šablónam surovej správy. Špeciálnym prípadom, na ktorý je dôležité upozorniť, je výstupný *forwarder* modul, ktorý sa používa na distribúciu prijatých správ na subkolektory. Pokiaľ by odosielať výlučne surové IPFIX správy, ich interpretácia by z dôvodu absencie šablón nebola možná. *Forwarder* modul však odošle definície šablón uložené v snímkach manažéra šablón skôr, než samotné správy, preto ani v tomto prípade nie je pridávanie šablón do surových správ potrebné. Po vložení



Obr. 4.6: Pridávanie dátových záznamov do novej surovej IPFIX správy

všetkých záznamov je nad novou surovou IPFIX správou vytvorená obálka dátovej správy, ktorá je následne vrátená zastrešujúcemu modulu.

4.4 Popis činnosti zastrešujúcich modulov

Poslednou časťou nového návrhu sú zastrešujúce moduly (vnútorné moduly kolektoru realizujúce obohacovanie alebo filtrovanie tokov), ktoré spájajú činnosti nových komponentov jadra kolektoru. Pri ich inštanciacii sú inicializované komponenty modifikátoru a tvorcu správ, ktoré sú využívané v priebehu ich práce. Pred vytvorením modifikátoru je nutné definovať príslušné callback funkcie, ktoré budú použité v priebehu úpravy záznamov. Tieto funkcie implementujú logiku úpravy záznamov, t.j. pri obohacovaní určujú spôsob získania hodnôt nových položiek a pri filtrovaní určujú pozície položiek určených k odstráneniu.

Pri inicializácii zastrešujúceho modulu je potrebné jeho prihlásenie na odber správ sedenia. Po prijatí správy sedenia oznamujúcej ukončenie spojenia s exportérom, je táto informácia predaná modifikátoru záznamov, ktorý môže následne uvoľniť šablóny definované v tomto spojení.

Po prijatí novej dátovej správy je jej kontext odoslaný do modifikátoru, ktorý na jeho základe aktualizuje kontext sedenia pre danú správu. Zároveň je tvorcom správ vytvorená nová surová IPFIX správa obsahujúca hlavičku pôvodnej správy. Zastrešujúci modul ďalej prechádza všetkými dátovými záznamami prijatej správy, ktoré sú odoslané modifikátoru k úprave. Upravené dátové záznamy sa vkladajú do novej správy v tvorcovi správ.

Po úspešnom spracovaní všetkých záznamov pôvodnej správy je vytvorená nová dátová správa, ktorá je odoslaná do zrefazenej linky kolektoru. Pokiaľ dôjde ku chybe v niektorej z časti jadra kolektoru, je namiesto novej správy odoslaná pôvodná, preto následne nevzniká riziko prípadnej straty dát pri výskyte chyby v obohacovaní.

Kapitola 5

Realizácia obohacovania správ a testovanie

Predposledná kapitola popisuje implementáciu nových komponentov jadra kolektoru a obohacujúcich modulov. Taktiež sú v nej zhodnotené výsledky výkonnostných testov merajúcich priepustnosť kolektoru pri použití nového návrhu obohacovania záznamov.

Implementácia nových častí kolektoru je napísaná v jazyku C a na kompiláciu kolektoru sa využíva prekladový systém *CMake*¹. Preklad kolektoru, spolu s knižnicou *libfds*, potrebnou na jeho činnosť, je popísaný v prílohe A. Dokumentácia nových častí kódu je vytvorená, rovnako ako vo zvyšných častiach kolektoru, pomocou nástroja *Doxygen*². V rámci testovania boli pre nové komponenty jadra vytvorené jednotkové testy, ktoré sa zameriavali na úpravu záznamov, nastavenie kontextu modifikátoru a vytváranie nových správ. Tieto testy boli realizované pomocou knižnice *Google Tests*³.

Testovanie výkonnosti prebiehalo na fyzickom stroji, ku ktorému mi bol poskytnutý prístup v rámci spolupráce s tímom LiberoRouter organizácie CESNET. Parametre tohoto zariadenia a prekladu je možné vidieť v tabuľke 5.1. Na stroji počas výkonnostného testovania neprebiehali žiadne iné výpočetne náročné úlohy. Samotné testovanie bolo založené na spustení kolektoru s jedným vstupným TCP modulom. Testovacie dáta boli na kolektor odosielané prostredníctvom nástroja *ipfixsend2*, ktorý je súčasťou kolektoru a slúži k jeho testovaniu. Ako výstupný modul bol použitý ukážkový (*dummy*) modul, ktorý zobrazuje štatistiky o spracovaných správach po ukončení činnosti kolektoru. Okrem týchto modulov sa v konfigurácii nachádza testovaný vnútorný modul, ktorý realizuje obohacovanie tokov. Cieľom testov je zistenie výkonnosti kolektoru, ktorá je vypočítaná ako podiel celkového počtu záznamov a času od prijatia dát vstupným TCP modulom až po ukončenie práce výstupného modulu.

Sady IPFIX záznamov zachytených z pozorovania tokov na sieti, ktoré boli použité pri výkonnostných testoch, mi boli poskytnuté Ing. Lukášom Hutákom. Prvá z nich, v testoch označovaná ako *bežná*, obsahuje IPFIX záznamy so základnými meranými vlastnosťami tokov. Druhá sada, v testoch tiež označovaná ako *aplikačná*, obsahuje navyše položky získané z aplikačných vrstiev (napr. DNS alebo HTTP). Výkonnosť kolektoru je taktiež podmienená typom položiek vyskytujúcich sa v prijatých správach. Je teda dobré podotknúť, že správy v bežnej sade obsahujú výhradne položky so statickou dĺžkou, zatiaľ čo v zázna-

¹<https://cmake.org/>

²<https://www.doxygen.nl/>

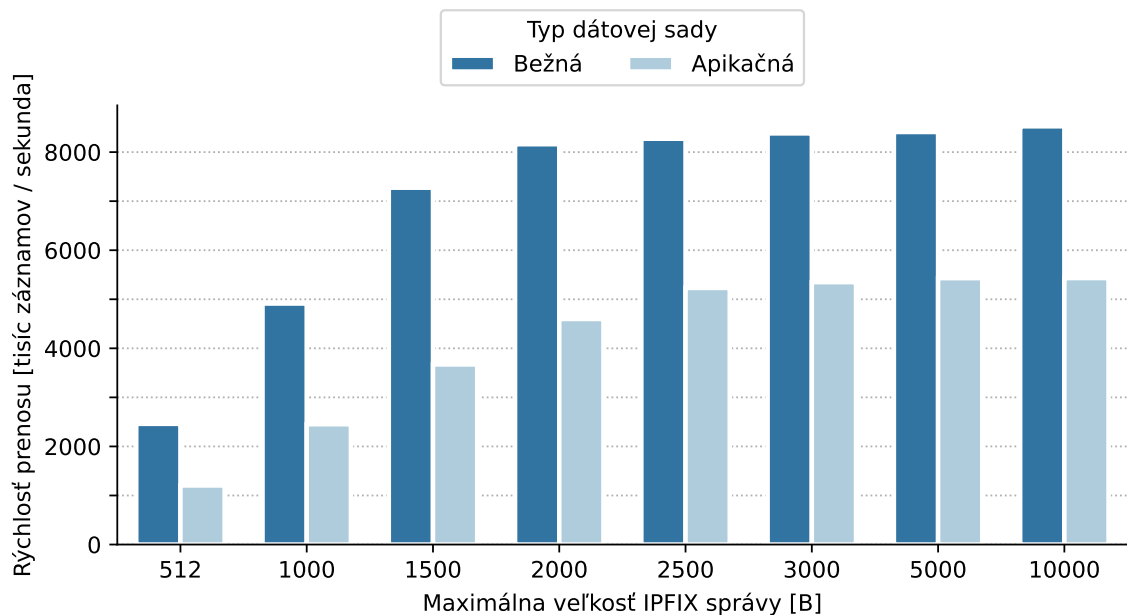
³<https://github.com/google/googletest>

moch aplikačnej sady sa vyskytujú tiež položky s variabilnou dĺžkou. V rámci týchto sád sú navyše generované IPFIX správy s postupne narastajúcimi veľkosťami, čo umožňuje zistenie priepustnosti kolektoru v závislosti od počtu záznamov vyskytujúcich sa v prijatých správach.

Procesor	Xeon (R) CPU E5-2630 v3 @ 2.40GHz
Počet jadier	8
RAM	64GiB
Operačný systém	Scientific Linux
Verzia OS	release 7.9 (Nitrogen) Kernel 3.10.0-1160.25.1.el7.x86_64
Použité nástroje	gcc 4.8.5, cmake 2.8.12.2
Typ prekladu	optimalizácia O2, bez ladiacich symbolov

Tabuľka 5.1: Informácie o použitom testovacom prostredí

K porovnaniu výkonnosti nových častí kolektoru bola zvolená konfigurácia kolektoru obsahujúca vnútorný anonymizačný modul. Pri týchto testoch bol zvolený jednoduchý anonymizačný algoritmus, v rámci ktorého je zachovaná vrchná polovica IP adresy a jej zvyšok je vynulovaný. Výber tejto konfigurácie nám poskytuje základnú predstavu o priepustnosti kolektoru pri použití modulu pracujúceho priamo s dátovými záznamami prijatých správ. Výsledky testu výkonnosti kolektoru s anonymizačným modulom je možné vidieť v grafe 5.1.



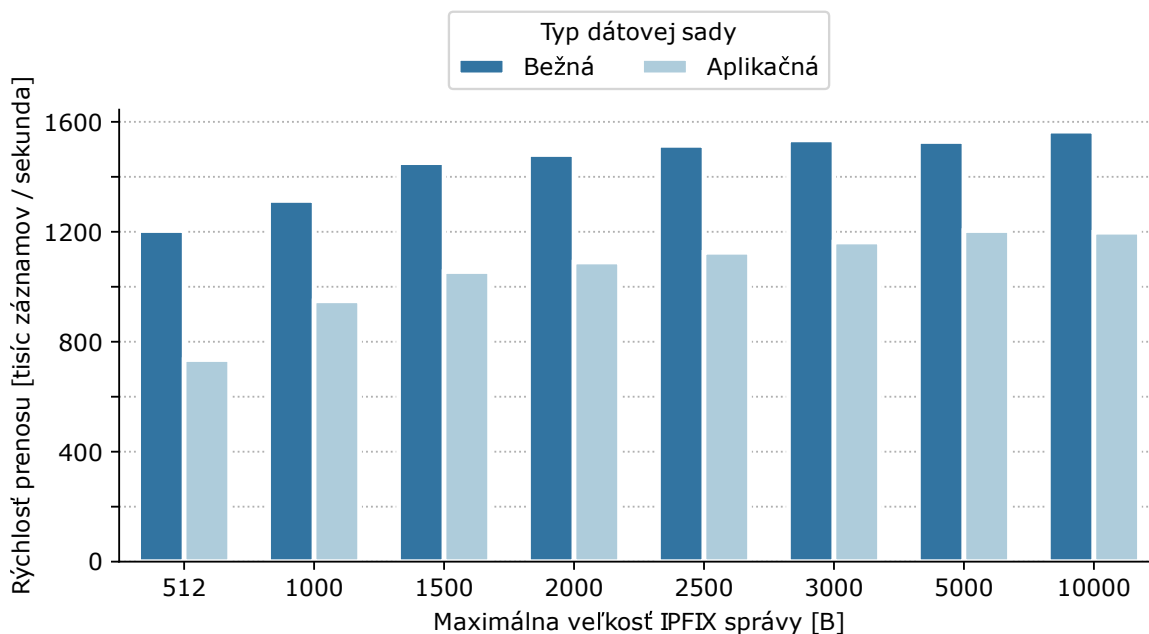
Obr. 5.1: Výkonnosť kolektoru s anonymizačným modulom

Horná hranica priepustnosti kolektoru pri anonymizácii tokov je približne 8 miliónov záznamov za sekundu, pričom pri aplikačnej sade je to o 2 milióny záznamov menej. Tento pokles výkonnosti je spôsobený výskytom položiek premenlivej veľkosti v aplikačnej sade. Zároveň si môžeme všimnúť, že výkonnosť sa zvyšuje s narastajúcou veľkosťou prijatých IPFIX správ. Táto skutočnosť je zapríčinená zvyšujúcim sa počtom dátových záznamov vo

väčších správach, čím dochádza ku zníženiu nákladov spojených s réžiou práce so samotnými správami.

5.1 Ukážkový obohacujúci modul

Za účelom testovania bol vytvorený ukážkový modul obohacujúci toky na základe zvolenej konfigurácie, ktorú je možné vidieť v prílohe B. Konfigurácia tohto modulu obsahuje definície nových položiek, ktorými môže byť celé/desatinné číslo alebo textový reťazec. Zároveň sa v konfigurácii nachádzajú hodnoty nových položiek, ktoré sú použité pri samotnom obohacovaní tokov. Táto skutočnosť umožňuje izolovane otestovať výkonnosť nových častí jadra kolektoru bez nutnosti vyhľadávania hodnôt nových položiek v databáze alebo inom zdroji dát. Toto vyhľadávanie sa v ďalších testoch preukáže ako časovo kritický úsek procesu obohacovania. Graf 5.2 zobrazuje výkonnosť kolektoru s ukážkovým obohacujúcim modulom po pridaní dvoch celočíselných položiek.



Obr. 5.2: Výkonnosť kolektoru s ukážkovým obohacujúcim modulom

Výsledky testovania ukážkového modulu zobrazujú zníženie priepustnosti kolektoru pri bežnej dátovej sade v porovnaní s anonymizačným modulom na hranicu 1,5 milióna záznamov za sekundu. Tento výrazný pokles je zapríčinený skutočnosťou, že oproti anonymizačnému modulu vykonáva obohacujúci modul pokročilejšie operácie v podobe zmeny štruktúry záznamov a ich šablón pre každý prijatý dátový záznam.

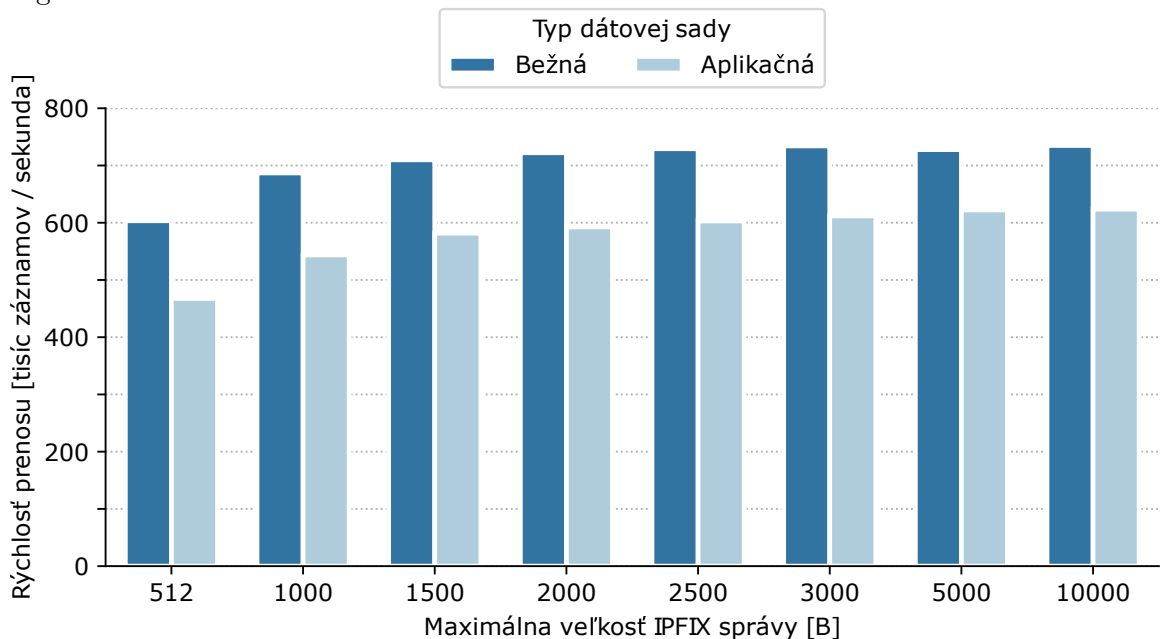
5.2 Moduly ASN a GeoIP

Dôležitou súčasťou práce modulov ASN a GeoIP, ktorá v tejto práci doposiaľ nebola popísaná, je získavanie geolokačných informácií a čísel autonómnych systémov. V súčasnosti sú v rámci implementácie spomínaných modulov na túto aktivitu využívané voľne dostupné

databázy *GeoLite2*⁴ vyvíjané organizáciou MaxMind. Pri ich výbere je možné siahnuť po dvoch variantách – textový formát CSV alebo interný formát MMDB (*MaxMind DataBase*). Z hľadiska výkonnosti boli v implementácii využité databázy interného formátu, ktoré predstavujú v porovnaní s textovým formátom optimalizovaný spôsob vyhľadávania potrebných informácií. Záznamy tohoto typu sú uložené v štruktúre binárneho stromu, pričom je optimalizovaný aj samotný proces vyhľadávania, ktorý je popísaný spolu s formátom MMDB vo verejnej dokumentácii⁵. Pri implementácii bola k práci s týmito databázami využitá knižnica *libmaxminddb*⁶.

Vytvorenie odkazu na databázový objekt prebieha v inicializačnej funkcii obohacovacieho modulu. Tento odkaz je následne predaný do modifikátoru prostredníctvom dát, ktoré sú využité pri volaní obohacovacej (callback) funkcie. Modifikátor v priebehu obohacovania toku túto funkciu invocuje a predá jej odkaz na tieto dáta. Počas jej činnosti dochádza ku vyhľadaniu potrebných informácií z uloženého databázového odkazu pomocou IP adresy získanej z aktuálne spracovávaného záznamu toku. Pokiaľ záznam už položky určené na obohatenie obsahuje, ich pridanie do záznamu je vynechané. Získané dáta sú obohacovacou funkciou vložené do vnútornej štruktúry modifikátoru, ktorý následne pokračuje v činnosti úpravy dátového záznamu. V priebehu ukončenia činnosti modulu je odkaz na databázu bezpečne odstránený.

Modul ASN využíva na vyhľadávanie čísel AS binárnu databázu *GeoLite2-ASN*. Pri konfigurácii tohto modulu je potrebné vopred definovať cestu ku tejto databáze, vďaka ktorej je následne prípadná aktualizácia jednoduchšia. Čísla autonómnych systémov definované informačnými elementami `bgpSourceAsNumber` a `bgpDestinationAsNumber`, ktorými sú toky obohacované, sa nachádzajú v spoločnom rozsahu informačných elementov organizácie IANA.



Obr. 5.3: Výkonnosť kolektoru pri použití modulu ASN

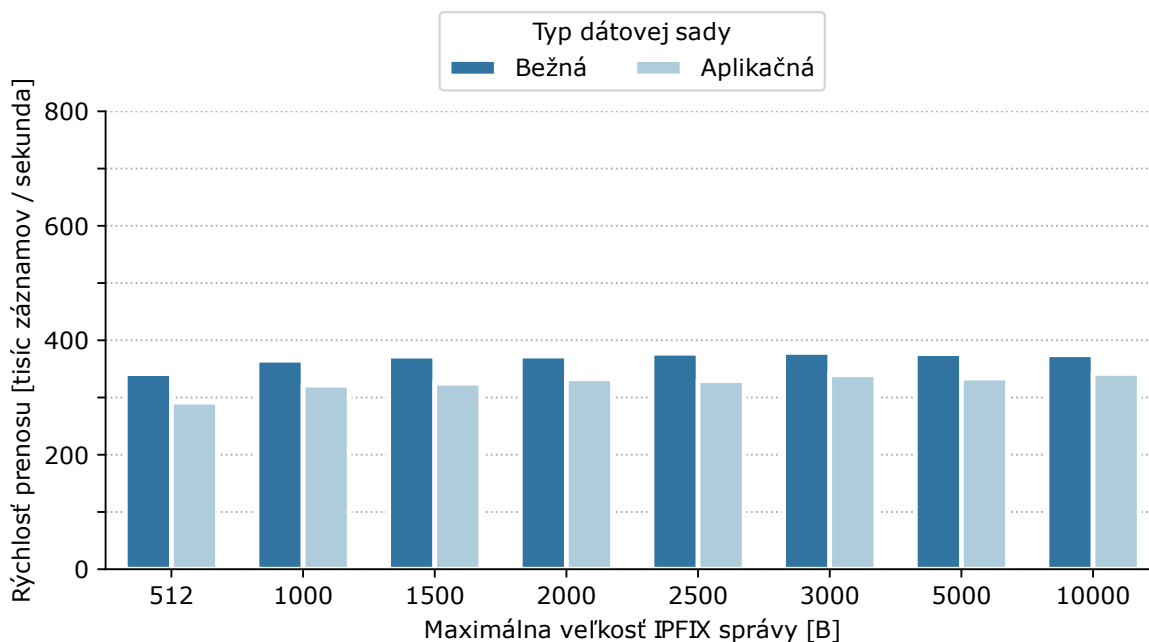
⁴K ich stiahnutiu je potrebné vytvorenie účtu. Pri testovaní boli využité nesploplatené databázy dostupné na: <https://dev.maxmind.com/geoip/geolite2-free-geolocation-data>

⁵Popis formátu MMDB: <https://maxmind.github.io/MaxMind-DB/>

⁶GitHub stránky knižnice: <https://github.com/maxmind/libmaxminddb>

Graf 5.3 zobrazuje výkonnosť kolektoru pri použití obohacovacieho ASN modulu. Ako je možné vidieť, horná hranica priepustnosti je pri bežnej dátovej sade približne 700 tisíc záznamov za sekundu, v prípade aplikačnej sady o stotisíc menej. Zníženie rýchlosti o polovicu je oproti ukázkovému modulu zapríčinené potrebou vyhľadávania čísel AS v použitej databáze.

V prípade geolokácie poskytuje MaxMind na výber dve geolokačné databázy – *GeoLite2-Country* a *GeoLite2-City*. Pri testovaní bola využitá databáza *City*, ktorá v porovnaní s databázou *Country* disponuje väčším množstvom poskytovaných geografických informácií s nepatrným rozdielom v nameranej priepustnosti. Konfigurácia tohto modulu je pokročilejšia a v porovnaní s modulom ASN sa okrem cesty ku použitej databáze definujú aj obohacujúce geolokačné položky. Modul GeoIP ponúka obohatenie toku o kód kontinentu/krajiny, názvu mesta alebo priamo GPS súradníc. Príklad konfigurácie tohto modulu je popísaný v prílohe B. Informačné elementy geolokačných položiek však nie sú súčasťou verejného rozsahu. Aby bol kolektor schopný interpretovať tieto položky, boli do knižnice *libfds* pridané definície nových informačných elementov v rámci privátneho rozsahu organizácie CESNET. Výsledky testovania kolektoru s modulom GeoIP, počas ktorého boli toky obohatené o položky kódu krajiny a názvu mesta, sa nachádzajú v grafe 5.4.



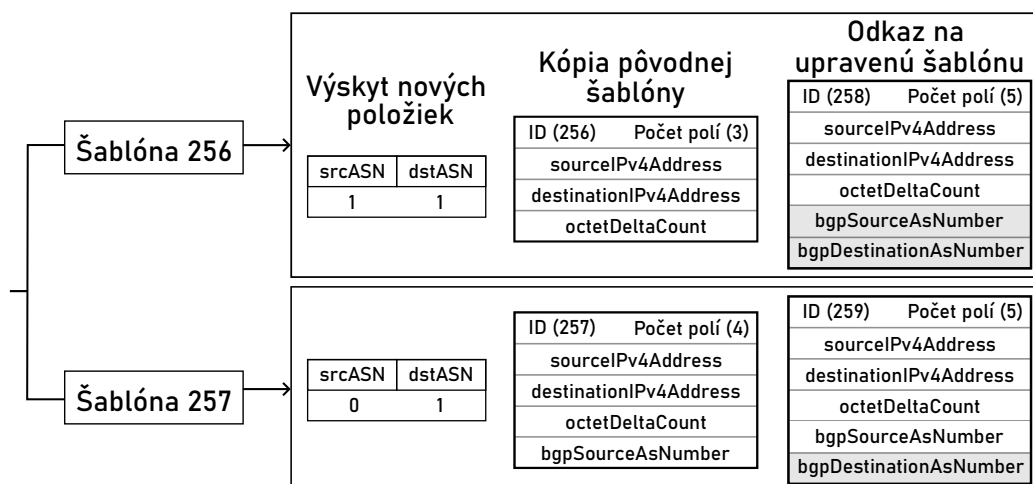
Obr. 5.4: Výkonnosť kolektoru pri použití modulu GeoIP

V porovnaní s modulom ASN dochádza ku zníženiu výkonnosti spôsobenému rozdielom veľkostí použitých databáz na hranicu 350 tisíc spracovaných záznamov za sekundu. Väčší počet záznamov databázy *City* oproti databáze *ASN* zapríčiňuje zníženie rýchlosti vyhľadávania v nej uložených informácií. Zároveň je možné všimnúť si, že rozdiel priepustnosti pri bežnej a aplikačnej sade je minimálny, čo je zapríčinené obohacovaním tokov textovými reťazcami (v porovnaní s modulom ASN, ktorý pridával iba čísla). Práca so záznamami bežnej sady sa tak stáva komplikovanejšiou z dôvodu výskytu nových položiek s variabilnou dĺžkou. Priepustnosť kolektoru zvolenej konfigurácie je navyše rovnaká pre všetky veľkosti IPFIX správ, z dôvodu absencie vplyvu režie spracovávania správ zapríčinené vyhľadávaním dát nových položiek.

Počas navrhovania obohacovania správ kolektoru IPFIXcol2 bolo zrejmé, že vykonávanie komplikovaných operácií pre každý dátový záznam prijatých správ bude mať negatívny vplyv na jeho celkovú výkonnosť. V rámci pokusov o zvýšenie priepustnosti bol proces obohacovania tokov profilovaný programom *gperftools*⁷. Výsledkom profilovania bolo zistenie, že okrem prístupu do databázy boli hlavnými príčinami poklesu priepustnosti modifikácia šablón a dátových záznamov. Proces úpravy záznamu je nutné realizovať pre každý záznam samostatne, avšak úprava šablóny môže byť uskutočnená len v niektorých prípadoch. Na základe tejto skutočnosti bola do procesu obohacovania implementovaná optimalizácia, ktorej cieľom je eliminácia úpravy šablón vedúca ku zvýšeniu priepustnosti.

5.3 Optimalizácia modifikácie záznamov

Princíp optimalizácie spočíva v prepojení pôvodnej šablóny s informáciou o výskyte nových položiek pri jej obohacovaní. Modifikátor si pri úprave dátového záznamu uloží informácie o jeho pôvodnej šablóne spolu s pridanými položkami do mapy modifikovaných šablón, ako je možné vidieť na obrázku 5.5. Ak v priebehu svojej činnosti narazí na záznam definovaný rovnakou šablónou, ktorý by bol následne obohatený rovnakými položkami, môže priamo využiť odkaz na modifikovanú šablónu nachádzajúcu sa v položke mapy. Takýmto spôsobom nemusí dochádzať k úprave šablón jednotlivo pre všetky modifikované záznamy.



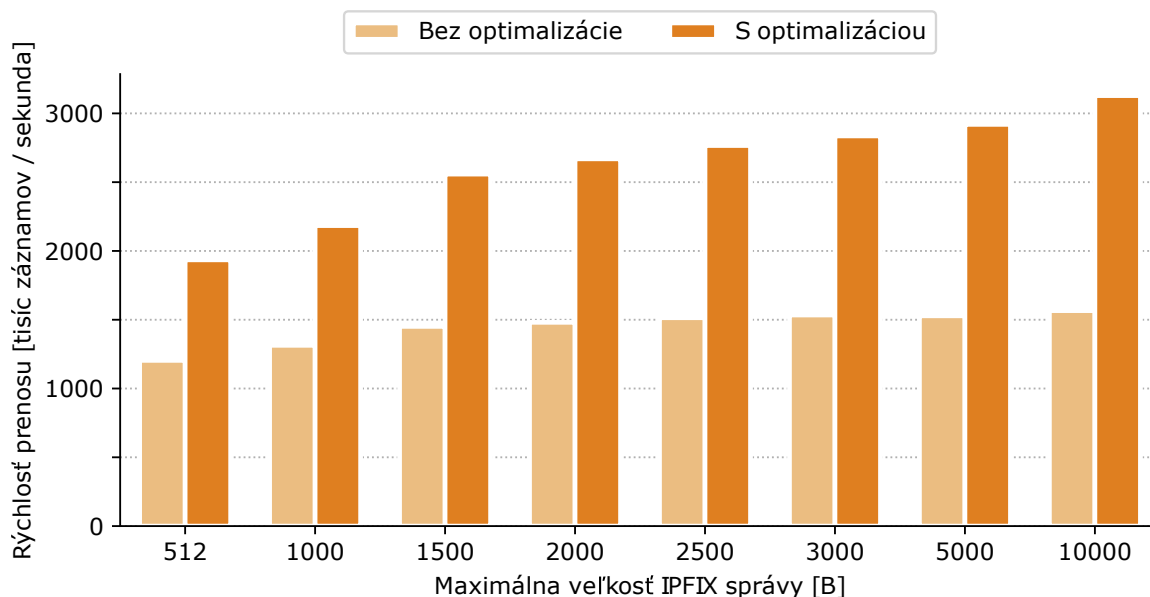
Obr. 5.5: Položky mapovania modifikovaných šablón po pridaní optimalizácie

Prítomnosť pôvodnej šablóny v položke mapy vychádza zo skutočnosti, že v priebehu činnosti kolektoru môžu byť šablóny redefinované. Ak by sa v položke mapy nenachádzala definícia pôvodnej šablóny, nebolo by možné určiť, či daná modifikovaná šablóna skutočne reflektuje zmeny vykonané v aktuálne menenej šablóne. Zároveň je dôležité poukázať na skutočnosť, že v položke mapy sa nachádza kópia pôvodnej šablóny. V tomto prípade nemôže byť použitý jednoduchý odkaz, pretože pôvodná šablóna je spravovaná preprocesorom, ktorý ju môže kedykoľvek odstrániť, a teda odkaz by sa tak mohol stať neplatným.

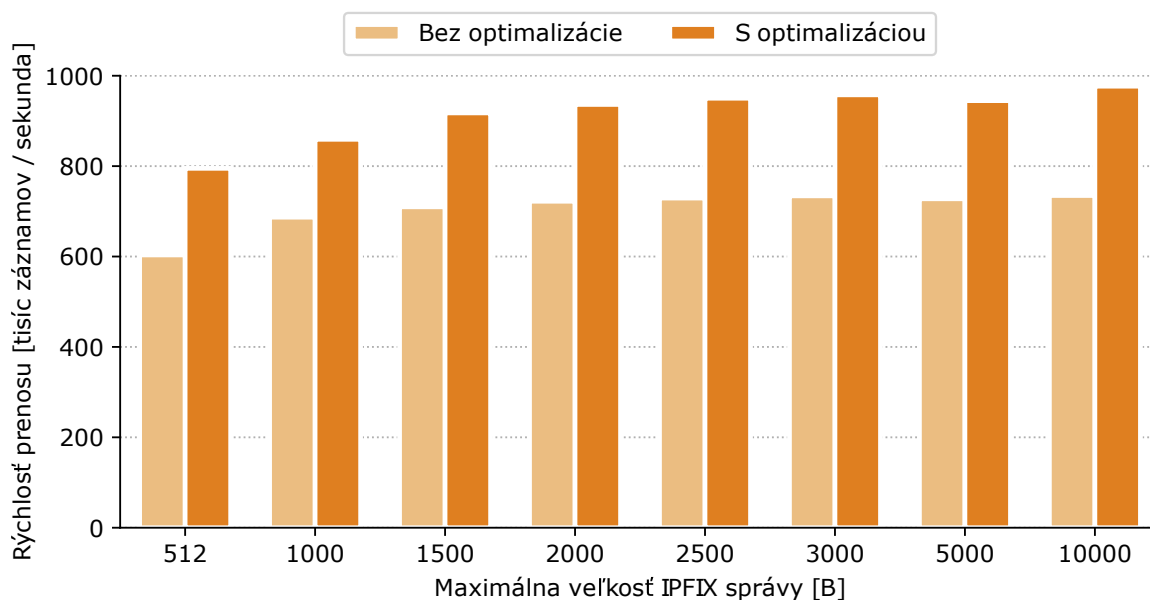
⁷<https://github.com/gperftools>

Testy s použitím optimalizovaného obohacovania záznamov

Počas testovania vytvorenej optimalizácie boli zrealizované doposiaľ vytvorené testy na ukázkovom module spolu s modulmi ASN a GeoIP. Pri týchto testoch bola použitá bežná dátová sada. Graf 5.6 zobrazuje výsledky výkonnostných testov ukázkového modulu pred a po aplikovaní navrhnutej optimalizácie. Výsledky priepustnosti kolektoru pri použití modulov ASN a GeoIP zobrazujú grafy 5.7 a 5.8.



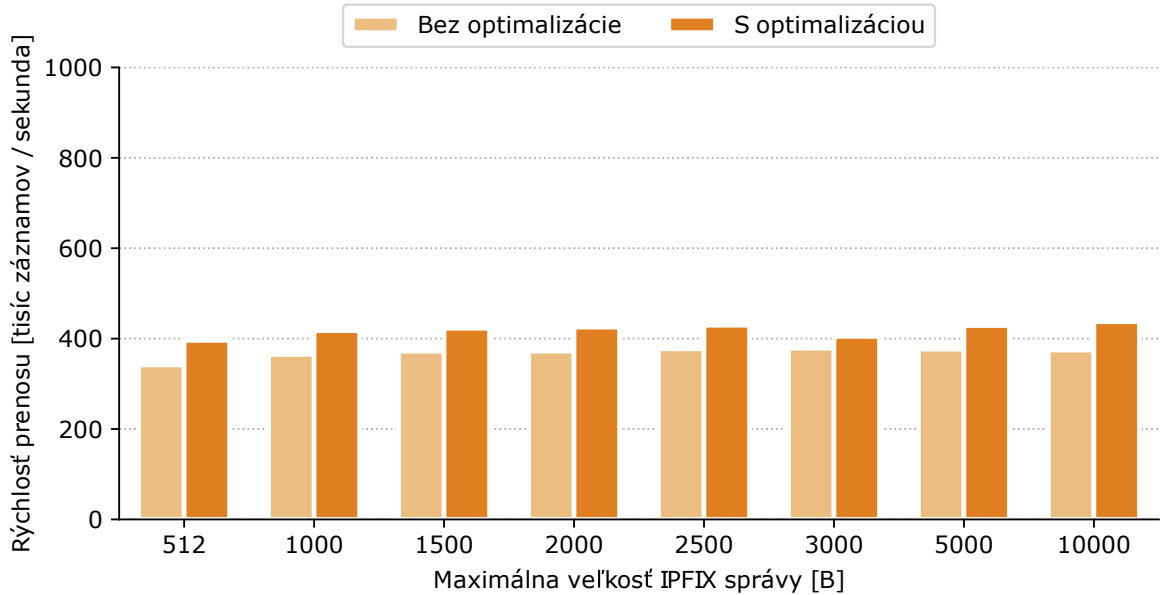
Obr. 5.6: Porovnanie výkonnosti kolektoru s ukázkovým modulom pri použití optimalizácie



Obr. 5.7: Porovnanie výkonnosti kolektoru s modulom ASN pri použití optimalizácie

Zavedenie optimalizácie do činnosti úpravy záznamov prinieslo pozitívne výsledky v prípade obohacovania tokov ukázkovým modulom, pri ktorom sa priepustnosť kolektoru mies-

tami zdvojnásobila a dosahovala maximálnu hodnotu 3 milióny spracovaných záznamov za sekundu. Pre modul ASN došlo taktiež ku zvýšeniu priepustnosti o 200 tisíc záznamov, avšak v porovnaní s ukázkovým modulom je tento nárast nižší. Dôvodom tohto menej priaznivého výsledku je skutočnosť, že v priebehu procesu modifikácie narážajú vnútorné moduly na pomalšiu rýchlosť získavania dát z použitých databáz. Tento jav je viac viditeľný pri testovaní optimalizácie modulu GeoIP, pri ktorej došlo ku zvýšeniu výkonnosti iba o necelých 50 tisíc záznamov.



Obr. 5.8: Porovnanie výkonnosti kolektora s modulom GeoIP pri použití optimalizácie

Keďže priepustnosť kolektora s obohacujúcimi modulmi predstavuje v porovnaní jeho výkonnosti bez ich využitia úzke hrdlo procesu spracovania záznamov, viaže sa použiteľnosť tohto návrhu na počet záznamov exportovaných meracím procesom v danej sieti. Testovanie nových modulov prinieslo pozitívne výsledky vzhľadom na možnosť ich reálneho využitia v monitorovacích architektúrach sietí, ktorých rýchlosť exportu tokov dosahuje hranice pol milióna záznamov za sekundu.

Z pohľadu škálovateľnosti nového návrhu pri jeho použití v rozsiahlejších sieťach však boli zistené nepriaznivé dôsledky dopadu obohacovania tokov na zníženie výkonnosti kolektora. Tento pokles sa môže premietnuť do zvýšenia oneskorenia analýzy záznamov prebiehajúcej v reálnom čase, ktorá je často kľúčovou vlastnosťou monitorovania. V horšom prípade by mohli byť toky zahodené, pretože kolektor by nedokázal dostatočne rýchlo odberať pakety odosielané z exportéru. V takýchto prípadoch sa však ponúka možnosť využitia distribuovanej architektúry, pri ktorej sú toky odosielané na hlavný kolektor, ktorý by rozkladal celkovú záťaž vo forme preposielania záznamov na subkolektory realizujúce ich obohacovanie. Využitie tejto architektúry je navyše podporované priamo v kolektore IPFIXcol2, ktorý túto činnosť vykonáva pomocou výstupného *forwarder* modulu.

5.4 Budúci rozvoj projektu

V súčasnej dobe komponent modifikácie nepodporuje obohacovanie záznamov nastavení a prevádzkových parametrov exportéru alebo pridávanie štruktúrovaných dát, ktoré neboli

považované za prioritné pri vytváraní počiatočného návrhu. V rámci ďalšieho smerovania projektu by tak mohlo dôjsť ku začleneniu týchto požiadaviek do procesu obohacovania.

Výsledky výkonnostných testov poukazujú na pokles priepustnosti kolektoru, ktorý je zapríčinený pristupovaním do databáz za účelom získavania dát použitých pri obohacovaní tokov. V záujme zvýšenia priepustnosti by aj v tomto aspekte mohli byť implementované techniky redukujúce čas spojený s týmito úkonmi. Jedno z možných vylepšení je ukládanie získaných dát do vyrovnávacej pamäte, vďaka čomu by pri viacnásobnom vyhľadávaní rovnakých záznamov mohlo potenciálne dôjsť k zníženiu času potrebného k ich nájdeniu.

Flexibilný návrh nového obohacovania záznamov ponúka možnosti jednoduchého vytvárania vnútorných modulov, ktoré by do tokov pridávali rôznorodé informácie využiteľné pri bezpečnostných analýzach alebo správe sietí. Príkladom ďalšieho rozšírenia kolektoru môže byť prenesenie implementácie vybraných vnútorných modulov pôvodnej verzie kolektoru, medzi ktoré patrí modul UID (User ID) pridávajúci do tokov informácie spojené s identitou komunikujúceho užívateľa. Súčasťou návrhu modifikácie záznamov je filtrovanie položiek dátových záznamov, ktoré je implementované v nových častiach jadra kolektoru. Jeho aplikovanie na nových moduloch však presahuje stanovené ciele vyplývajúce zo zadania tejto práce. Pridanie filtračných modulov by preto mohlo byť realizované v pokračovaní vývoja vnútorných modulov kolektoru IPFIXcol2. Tieto moduly by mohli byť použité napríklad k redukcii potrebného miesta na dlhotrvajúce uloženie tokov alebo k odstráneniu nadbytočných hodnôt z prijatých dát, ktoré nie sú ďalej spracovávané a preto nemusia byť uložené.

Kapitola 6

Záver

Témou tejto bakalárskej práce bolo štúdium problematiky obohacovania tokov dátami z externých zdrojov a jeho následné využitie v návrhu nových častí kolektoru IPFIXcol2 v súčasnosti používanom nielen na monitorovanie tokov v sieťach CESNETu. V rámci skúmania možností návrhu boli zhodnotené pozitívne a negatívne stránky obohacujúcich modulov v pôvodnej verzii tohto kolektoru, ktorých nedostatky boli adresované v novom návrhu. Návrh nového spôsobu obohacovania tokov vyústil do vytvorenia nových vnútorných modulov ASN a GeoIP, začlenených do kolektoru IPFIXcol2. V závere boli uskutočnené výkonnostné testy hodnotiace priepustnosť kolektoru pri použití týchto novovytvorených modulov.

Prvá časť tejto práce je zameraná na monitorovanie zariadení komunikujúcich v sieti s dôrazom kladeným na monitorovaciu architektúru typu NetFlow/IPFIX. Celkový popis tejto architektúry, ktorý pozostáva najmä z činnosti exportéru a kolektoru, poukazuje na súčasné trendy v tejto oblasti spočívajúce v realizácii pokročilých činností spojených s úpravou záznamov na strane kolektoru. S týmto typom architektúry je úzko spätý protokol IPFIX, ktorého znalosť je pre prácu s dátami reprezentujúcimi monitorované toky nevyhnutná. Z toho dôvodu je súčasťou práce aj podrobný popis mechanizmov a dátových štruktúr využívaných v rámci tohto protokolu.

V úvode druhej časti sa nachádza popis kolektoru IPFIXcol2, na ktorom bol realizovaný nový návrh obohacovania tokov. Prehľad tohto kolektoru začína popisom architektúry, v ktorej boli zhrnuté jeho silné stránky v podobe paralelizácie a modularity. V rámci jeho rozšírenia boli vyzdvihnuté vybrané časti jadra realizujúce činnosti spojené s prácou prijatých záznamov alebo správou šablón, ktoré boli využité pri implementácii nových častí kolektoru. Záver tejto kapitoly je zameraný na spôsob vytvárania nových modulov, na základe ktorého vznikli vnútorné moduly realizujúce obohacovanie tokov.

Nasledujúce kapitoly obsahujú návrh nového spôsobu obohacovania záznamov. V úvode sú vysvetlené prínosy obohacovania tokov novými informáciami na príkladoch čísel autonómnych systémov a IP geolokácie. V texte sa ďalej nachádza popis spôsobu obohacovania tokov týmito informáciami v pôvodnej verzii kolektoru, od ktorého sa odvíjajú základy nového návrhu. Samotný nový návrh je založený na vytvorení nových komponentov jadra kolektoru zastrešujúcich prácu spojenú s úpravou záznamov o tokoch. V rámci využitia týchto rozhraní došlo k implementácii nových vnútorných modulov ASN a GeoIP, ktoré obohacujú toky spomínanými informáciami.

V závere tejto práce boli nad novými modulmi zrealizované výkonnostné testy, na základe ktorých bol zhodnotený dopad obohacovania tokov na priepustnosť kolektoru. Výsledky týchto testov sú pozitívne, nakoľko z nich vyplýva, že obohacovanie tokov je realizovateľné v sieťach produkujúcich záznamy o tokoch v maximálnej rýchlosti až pol milióna

záznamov za sekundu. Zároveň však tieto testy poukázali na skutočnosť, že obohacovanie tokov exportovaných vyššou rýchlosťou predstavuje obmedzenie výkonnosti použitej architektúry. Pre monitorovanie týchto sietí boli diskutované postupy, ktorými môže dôjsť ku zvýšeniu výkonnosti kolektorov realizujúcich obohacovanie tokov, napríklad v podobe použitia distribuovanej monitorovacej architektúry. Implementácia vytvoreného návrhu otvára nové možnosti rozšírenia kolektoru IPFIXcol2 vnútornými modulmi, ktoré by obohacovali toky rôznymi analyticky zaujímavými informáciami.

Literatúra

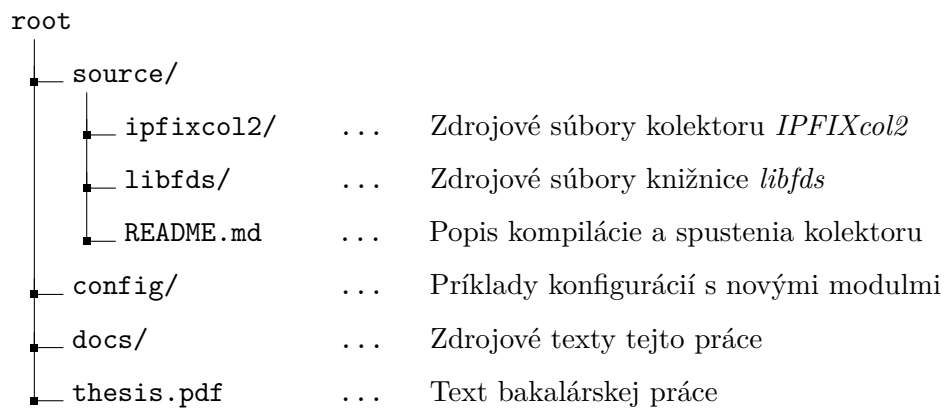
- [1] BOSCHI, E., MARK, L., QUITTEK, J. et al.: *IP Flow Information Export (IPFIX) Implementation Guidelines*. RFC 5153. RFC Editor, Apríl 2008. Dostupné z: <https://www.rfc-editor.org/rfc/rfc5153.txt>.
- [2] CLAISE, B. a TRAMMELL, B.: *Information Model for IP Flow Information Export (IPFIX)*. RFC 7012. RFC Editor, September 2013. Dostupné z: <http://www.rfc-editor.org/rfc/rfc7012.txt>.
- [3] CLAISE, B.: *Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information*. RFC 5101. RFC Editor, Január 2008. Dostupné z: <http://www.rfc-editor.org/rfc/rfc5101.txt>.
- [4] CLAISE, B., AITKEN, P. a TRAMMELL, B.: *Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of Flow Information*. RFC 7011. RFC Editor, September 2013. Dostupné z: <https://www.rfc-editor.org/rfc/rfc7011.txt>.
- [5] HAWKINSON, J. a BATES, T.: *Guidelines for creation, selection, and registration of an Autonomous System (AS)*. RFC 1930. RFC Editor, March 1996. Dostupné z: <http://www.rfc-editor.org/rfc/rfc1930.txt>.
- [6] HOFSTEDÉ, R., ČELEDA, P., TRAMMELL, B. et al.: Flow Monitoring Explained: From Packet Capture to Data Analysis With NetFlow and IPFIX. *IEEE Communications Surveys & Tutorials*. 2014, zv. 16, č. 4, s. 2037–2064. DOI: 10.1109/COMST.2014.2321898. ISSN 1553-877X.
- [7] HUTÁK, L.: *Nová generace IPFIX kolektorů*. Brno, 2018. Diplomová práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedúci práce WRONA, J. Dostupné z: <http://hdl.handle.net/11012/84955>.
- [8] MATOUŠEK, P.: *Síťové služby a jejich architektura*. Publishing house of Brno University of Technology VUTIUUM, 2014. 396 s. ISBN 978-80-214-3766-1. Dostupné z: <https://www.fit.vut.cz/research/publication/10567>.
- [9] MUIR, J. A. a OORSCHOT, P. C. V.: *Internet Geolocation: Evasion and Counterevasion*. New York, NY, USA: Association for Computing Machinery. dec 2009, zv. 42, č. 1. DOI: 10.1145/1592451.1592455. ISSN 0360-0300. Dostupné z: <https://doi.org/10.1145/1592451.1592455>.
- [10] NOCTION: *The Evolution of Network Flow Monitoring, from NetFlow to IPFIX* [online, vid. 28.02.2023]. Júl 2020. Dostupné z: <https://www.noction.com/blog/network-flow-monitoring>.

- [11] QUITTEK, J., BRYANT, S., CLAISE, B. et al.: *Information Model for IP Flow Information Export*. RFC 5102. RFC Editor, Január 2008. Dostupné z: <http://www.rfc-editor.org/rfc/rfc5102.txt>.
- [12] SADASIVAN, G., BROWNLEE, N., CLAISE, B. et al.: *Architecture for IP Flow Information Export*. RFC 5470. RFC Editor, Marec 2009. Dostupné z: <http://www.rfc-editor.org/rfc/rfc5470.txt>.
- [13] SEDLÁK, M.: *Filtrace a profilování IP toků*. Brno, 2020. Bakalářská práce. Vysoké učení technické v Brně, Fakulta informačních technologií. Vedúci práce KUČERA, J. Dostupné z: <https://www.fit.vut.cz/study/thesis/22989/>.
- [14] VELAN, P.: Practical experience with IPFIX flow collectors. In: IFIP/IEEE, ed.: *International Symposium on Integrated Network Management* [online, vid. 10.03.2023]. Január 2013, s. 1021–1026. ISBN 978-1-4673-5229-1. Dostupné z: <https://is.muni.cz/publication/1120446/ipfix-collectors-paper.pdf>.
- [15] VELAN, P. a KREJČÍ, R.: Flow Information Storage Assessment Using IPFIXcol. In: SADRE, R., NOVOTNÝ, J., ČELEDA, P. et al., ed.: *Dependable Networks and Services*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, s. 155–158. ISBN 978-3-642-30633-4.
- [16] WIKIPEDIA: *SYN flood* — *Wikipedia, The Free Encyclopedia* [online, vid. 10.03.2023]. 21. Máj 2023. Dostupné z: https://en.wikipedia.org/wiki/SYN_flood.
- [17] ŽÁDNÍK, M., KROBOT, P., KEKELY, L. et al.: *Distribučný kolektor záznamů o IP tocích: návrh a první experiment* [online, vid. 10.03.2023]. CESNET, December 2014. Dostupné z: <https://www.cesnet.cz/wp-content/uploads/2015/04/securitycloud.pdf>.

Príloha A

Obsah priloženého pamäťového média

Štruktúra odovzdaného archívu v priloženej SD karte:



Príloha B

Konfigurácia nových modulov

Konfigurácia kolektoru je realizovaná pomocou jazyka XML, prostredníctvom ktorého je možné definovať vstupné, vnútorné a výstupné moduly kolektoru. Príklady konfigurácií sú dostupné na stránke repozitára IPFIXcol2¹.

Ukážkový obohacovací modul

Ukážkový modul umožňuje definíciu položiek, ktorými sú obohatené záznamy prichádzajúcich IPFIX správ. Povinnou súčasťou definície položky je:

- PEN (*Private Enterprise Number*)
- identifikačné číslo informačného elementu
- dátový typ (string, integer, decimal)
- hodnota položky

Voliteľne je možné definovať početnosť výskytu novej položky v obohatenom zázname. Príklad konfigurácie ukážkového modulu pri jednorázovom obohacovaní tokov celočíselnou položkou:

```
<intermediate>
  <name>Dummy intermediate plugin</name>
  <plugin>dummy_enricher</plugin>
  <params>
    <field>
      <pen>0</pen>
      <id>21</id>
      <type>integer</type>
      <value>1023456</value>
      <times>1</times>
    </field>
  </params>
</intermediate>
```

¹<https://github.com/CESNET/ipfixcol2/tree/master/doc/data/configs>

Modul ASN

Konfigurácia modulu ASN obsahuje ako jediný parameter cestu k databázovému súboru *GeoLite2-ASN*, ktorý bude využívaný v priebehu obohacovania tokov. Príklad konfigurácie tohto modulu:

```
<intermediate>
  <name>ASN enrichment</name>
  <plugin>asn</plugin>
  <params>
    <path>/home/user/maxmind/GeoLite2-ASN.mmdb</path>
  </params>
</intermediate>
```

Modul GeoIP

Konfigurácia modulu GeoIP obsahuje parameter cesty k databázovému súboru *GeoLite2-City*. Okrem neho je možné definovať geolokačné položky, ktorými budú obohacované záznamy prijatých správ. V súčasnosti je podporované obohacovanie tokov nasledovnými položkami:

- *continent* – textový reťazec označenia kontinentu (NA, EU, AF ...)
- *country* – označenie krajiny podľa normy ISO 3166-1 alpha-2 (SK, CZ, UK ...)
- *city* – názov mesta v anglickom jazyku
- *latitude* – súradnice zemepisnej šírky
- *longitude* – súradnice zemepisnej dĺžky

Pri vynechaní definícií týchto položiek je použitá predvolená kombinácia obohatenia toku kódom krajiny a názvom mesta. Príklad konfigurácie modulu GeoIP s položkami kódu kontinentu a názvu mesta:

```
<intermediate>
  <name>GeoIP enrichment</name>
  <plugin>geoip</plugin>
  <params>
    <path>/home/user/maxmind/GeoLite2-City.mmdb</path>
    <fields>
      <field>continent</field>
      <field>city</field>
    </fields>
  </params>
</intermediate>
```