

Technická univerzita v Košiciach
Fakulta elektrotechniky a informatiky
Katedra počítačov a informatiky

**Príspevok k riešeniu problematiky neintruzívneho merania
parametrov QoS**

Vedúci diplomovej práce:
Ing. František Jakab

Diplomant:
Alžbeta Kleinová

Konzultant diplomovej práce:
Ing. František Jakab

Košice 2005

Čestné prehlásenie

Prehlasujem, že som diplomovú prácu vypracovala samostatne s využitím uvedenej odbornej literatúry.

V Košiciach dňa 2.5.2005

.....
vlastnoručný podpis

Na tomto mieste bude vložené zadanie diplomovej práce

Pod'akovanie

Ďakujem Ing. Františkovi Jakobovi za pripomienky a odbornú pomoc pri vypracovaní diplomovej práce.

Za podnetné návrhy pri vytváraní flash aplikácie by som chcela poďakovať Ľubošovi Koščovi, Miroslavovi Potockému a Mariánovi Ceľuchovi.

Moja vďaka patrí aj mojej rodine za podporu počas celého štúdia na vysokej škole.

Názov práce : Príspevok k riešeniu problematiky neintruzívneho merania parametrov QoS

Katedra : Katedra počítačov a informatiky, TU FEI Košice

Autor : Alžbeta Kleinová

Vedúci DP : Ing. František Jakab

Konzultant DP : Ing. František Jakab

Dátum : 2.5.2005

Kľúčové slová : Monitorovací nástroj, IPFIX, NetFlow, tok, množina toku, RRD nástroj, QoS, jednosmerné oneskorenie, kolísanie oneskorenia, RTT

Anotácia : Diplomová práca je zameraná na problematiku monitorovacích nástrojov. Zaoberá sa implementáciou štandardu IPFIX na príklade protokolu NetFlow ako aj koncepciou architektúry základného meracieho nástroja BasicMeter. Praktická časť je venovaná vytvoreniu flash aplikácie pre animačný model architektúry.

Thesis title : Contribution to the Non-Intrusive Measurement of QoS

Department : Department of Computers and Informatics, TU FEI Košice

Author : Alžbeta Kleinová

Supervisor : Ing. František Jakab

Tutor : Ing. František Jakab

Date : 2.5.2005

Keywords : Monitoring tool, IPFIX, NetFlow, Flow, FlowSet, RRD tool, QoS, One-way delay, Jitter, RTT

Annotation : Diploma thesis is concentrated on the problematic of the monitoring tools. It deals with the implementation of the IPFIX standard on the NetFlow example and also with the architecture conception based on the measuring tool BasicMeter. Practical part of diploma thesis is devoted to creating flash application for the animated model of the architecture.

Obsah

Úvod	1
1. Motivácia	3
2. Formulácia úlohy.....	4
3. Analýza a porovnanie vlastností monitorovacích nástrojov pre neintruzívne meranie prevádzky v IP sieťach	5
3.1 MRTG (The Multi Router Traffic Grapher).....	7
3.2 NAGIOS	9
3.3 NETFLOW MONITOR	11
3.4 SMOKEPING	12
3.5 CACTI	14
4. Implementácia štandardu IPFIX na príklade protokolu NetFlow	17
4.1 Základné pojmy.....	17
4.2 Pasívne merania	19
4.2.1 Architektúra pasívnych meraní	19
4.3 Štandard IPFIX	21
4.3.1 Kolektor	22
4.3.2 Bod merania	23
4.3.3 Merací proces	23
4.3.4 Exportovací proces	24
4.3.5 Zhromažďovací proces	24
4.3.6 Pozorovacia doména.....	25
4.4 IPFIX správa	26
4.5 NetFlow verzia 5	29
4.6 NetFlow verzia 9.....	30
5. Konceptia architektúry základného meracieho nástroja "BasicMeter"	32
5.1 Architektúra meracej platformy	32
5.2 Analýza architektúry BasicMetra.....	33
5.3 Analýza architektúry kolektora.....	37
6. Návrh a implementácia „animačného modelu architektúry“ meracieho nástroja v prostredí Macromedia Flash MX	39
6.1 Úvod	39
6.2 Výber SW nástrojov pre realizáciu animačného modelu	39
6.3 Návrh animačného modelu architektúry BasicMetra.....	40
6.3.1 Meracia platforma	41
6.3.1.1 Analýza jednotlivých častí schémy	42
6.3.1.2 Analýza funkcionality a jednotlivých komponentov architektúry	43
6.3.2 Meranie QoS parametrov.....	50
7. Zhodnotenie dosiahnutých výsledkov riešenia diplomovej práce	55
8. Zoznam používaných skratiek.....	56
9. Zoznam použitej literatúry	58
10. Zoznam príloh	62
11. Zoznam obrázkov a tabuliek	63

Úvod

Monitorovanie siete je v súčasnosti veľmi dôležitým znakom pri ochrane a zabezpečovaní počítačovej siete. Je potrebné vyznať sa v dostupných monitorovacích nástrojoch a poznať princípy, protokoly a štandardy, na základe ktorých sú monitorovacie nástroje postavené.

Prvá kapitola obsahuje len súhrnné informácie o problematike. Druhá kapitola je zameraná na analýzu a porovnanie monitorovacích nástrojov. Poukazuje na výhody a nevýhody použitia najznámejších monitorovacích nástrojov z hľadiska ich využitia v počítačovej sieti.

Základom diplomovej práce je návrh animačného modelu meracieho nástroja. Keďže celá koncepcia je založená na implementácii štandardu IPFIX, je tejto problematike venovaná tretia kapitola. Táto kapitola rozoberá najmä implementáciu štandardu IPFIX na príklade protokolu NetFlow verzie 9. Sú tu tiež vysvetlené základné pojmy tak, ako sú chápané v rámci tejto diplomovej práce. V tretej kapitole je tiež rozpracovaná aj problematika neintruzívnych meraní, pretože práve tieto merania tvoria základ, na ktorom je postavený celý merací nástroj. Podrobnejšie sú rozobrané aj jednotlivé časti architektúry štandardu IPFIX a aj samotný protokol NetFlow verzia 9. V diplomovej práci je poukázané na jeho prednosti oproti predchádzajúcim verziám.

Štvrtá kapitola je venovaná problematike základného meracieho nástroja BasicMeter. Je v nej zahrnutá architektúra meracej platformy, analýza architektúry BasicMetra a aj kolektora. Každá architektúra je znázornená v blokovej schéme.

Hlavnou časťou tejto diplomovej práce je piata kapitola, ktorá je venovaná návrhu a implementácii animačného modelu architektúry, ktorý bol vytvorený v prostredí Macromedia Flash MX. Je tu analyzovaný výber SW nástroja pre realizáciu animačného modelu. V tejto kapitole je ďalej rozpracovaný návrh animačného modelu architektúry BasicMetra v prostredí Macromedia Flash MX. Posledná časť tejto kapitoly je venovaná problematike merania QoS parametrov. V poslednej kapitole sú zhrnuté dosiahnuté výsledky diplomovej práce.

Celá diplomová práca je teda akýmsi zhrnutím všetkých doterajších poznatkov získaných pri návrhu meracieho nástroja. Teoretická časť diplomovej práce poskytuje prehľad všetkých častí meracieho nástroja a ich vzájomnej závislosti.

1. Motivácia

Dôvodom pre vytvorenie tejto diplomovej práce bolo nájsť vhodný spôsob ako zhrnúť viacročné poznatky o meracom nástroji BasicMeter do jedného celku. Zhrnutie do jedného celku má význam z hľadiska jednoduchšej prezentácie dosiahnutých výsledkov v tejto oblasti. Bol teda vytvorený animačný model, ktorý má slúžiť na prezentačné účely, ale aj ako vzdelávací prostriedok pre ďalšie generácie, ktoré sa rozhodnú pokračovať v zdokonaľovaní a rozširovaní meracieho nástroja BasicMetra.

Animačný model detailne popisuje princípy fungovania meracieho nástroja. Popisuje všetky jeho jednotlivé časti. Zahrňuje teoretické poznatky, na ktorých je daný merací nástroj postavený a systematicky zobrazuje spôsob merania QoS parametrov.

2. Formulácia úlohy

Cieľom diplomovej práce je navrhnuť animačný model architektúry celej koncepcie meracieho nástroja, jej jednotlivých častí a meraní parametrov QoS v prostredí Macromedia Flash MX. Realizácia návrhu animačného modelu bude vychádzať zo všetkých doterajších poznatkov o meracom nástroji.

Čiastkové úlohy diplomovej práce:

- vytvorenie animačného modelu meracej platformy na báze BasicMetra
- návrh zobrazenia meracej platformy
- analýza architektúry meracieho nástroja BasicMetra a kolektora
- výber vhodného prostredia pre realizáciu animačného modelu, ktorý by sprehľadnil architektúru meracieho nástroja
- znázornenie jednotlivých meraní parametrov QoS

3. Analýza a porovnanie vlastností monitorovacích nástrojov pre neintruzívne meranie prevádzky v IP sieťach

V súčasnosti existuje veľké množstvo monitorovacích nástrojov. Avšak predtým ako začneme s monitorovaním siete, je dôležité poznať sieť, ktorú chceme monitorovať. Je potrebné poznať jej fyzickú a logickú topológiu ako aj protokoly, ktoré sú v danej sieti využívané.

Techniky monitorovania sú založené na získavaní štatistických informácií z rôznych zariadení na sieti. Je užitočné tieto informácie zbierať na jednom mieste, odkiaľ sa potom dá vyhodnocovať činnosť siete.

Primitívnou technikou pre monitorovanie siete je posielanie tzv. "echo" datagramu. Tento datagram požaduje okamžitú odozvu. Väčšina TCP/IP implementácií obsahuje službu ping, pracujúcu týmto spôsobom.

Existujú dva hlavné dôvody, prečo je nevyhnutné monitorovať sieť. Jedným je možnosť predpokladať zmeny pre budúcnosť, druhým odhaliť nečakané zmeny v stave siete. Nečakané zmeny môžu zahŕňať problémy ako chybný smerovač alebo prepínač, hackera pokúšajúceho sa získať nelegálny prístup do siete alebo chybu v komunikačných linkách. Administrátor siete, v ktorej je možné monitorovať jej stav, môže týmto problémom predchádzať. V sieti bez možnosti monitorovania, administrátor môže len reagovať na problémy, ktoré už v sieti vznikli.[27]

Jednou zo základných aktivít sieťových administrátorov je teda monitorovanie siete. Je to proces kontrolovania počítačov, systémov a služieb, z ktorých sa sieť skladá. Hoci nikdy nie je možné dopredu vedieť aká udalosť v sieti nastane, je možné sa na ňu dopredu pripraviť. Nikdy nie je isté, či zdroj napájania nie je odpojený alebo či nenastal výpadok servera, keď poklesla priepustnosť siete alebo keď je sieť hackovaná. Vždy je ale možné pripraviť sa na výskyt udalostí, ktoré by sa mohli vyskytnúť. Efektívne monitorovanie siete je také, ktoré dokáže upozorniť na moment, že sa situácia v sieti zmenila tak, aby bolo možné reagovať okamžite a umožniť tak skrátenie času poruchového stavu.

Monitorovací systém by mal poskytnúť nielen informácie o probléme, ale aj informácie ako zlepšiť sieť. Dobrý systém by mal byť schopný generovať log súbory a vytvárať grafy, ktoré by zobrazovali detaily danej siete. S týmito údajmi je potom možné zistiť optimálne nastavenia siete.

Najlepším spôsobom ako riadiť monitorovanie siete je pomocou softvérového balíka. Softvérový balík predstavuje komplexný súbor služieb a programov, ktoré sú potrebné pre monitorovanie siete bez ďalších externých programov a služieb. Dobrý softvérový balík môže pomôcť pri odhaľovaní a odstraňovaní problémov, ktoré v danej sieti už vznikli alebo môžu vzniknúť [16].

Na monitorovanie siete existuje veľké množstvo rôznych monitorovacích nástrojov. Každý nástroj poskytuje špecifický prístup ako riešiť problém, ktorý v sieti vznikol. V každom monitorovacom nástroji je zahrnutých množstvo funkcií, mechanizmov a údajov, ktoré je potrebné poznať. Je dôležité vedieť, ktorý monitorovací nástroj by bolo vhodné použiť pre vyriešenie daného problému v sieti.

Existujú monitorovacie nástroje, ktoré monitorujú a kontrolujú stav siete pre zistenie zmien v súborovom systéme, sledujú podozrivé aktivity ako napr. nezvyčajné alebo neočakávané otvorenie súborov, úspešné a neúspešné nahlásenie administrátora, neočakávané prerušenie alebo reštartovanie systému, nezvyčajné aktivity modemu, nezvyčajné alebo nadmerné počty emailov a pod.

Ďalšou skupinou monitorovacích nástrojov sú nástroje, ktoré sa zameriavajú na monitorovanie a kontrolovanie sieťovej prevádzky a sieťového spojenia. Ide napr. o spôsob pripojenia, pričom sa zisťuje odkiaľ a kedy bolo dané spojenie vytvorené, pripojenia do alebo z nezvyčajného miesta, neautorizované sieťové sondy, systematické skenovanie portov, prevádzka siete zameraná proti nastaveniam firewallu a nezvyčajné súborové prenosové aktivity.

Charakteristickou skupinou sú aj monitorovacie nástroje zamerané na autentifikáciu a autorizáciu informácii. Nástroje, ktoré monitorujú a kontrolujú aktivitu používateľov, ako napr. prihlasovanie používateľa a jeho opakované prihlásenie, nesprávne zadaný login používateľa, prihlásenie z nezvyčajného miesta alebo v neobvyklom čase, neautorizovaný spôsob prístupu k vyhradeným informáciám.

Existujú aj nástroje, ktoré preverujú dátovú, súborovú a softvérovú integritu. Príkladom takýchto monitorovacích nástrojov sú nástroje, ktoré monitorujú operačné

systémy a konfiguráciu nástroja pre možné spôsoby využitia, ako napr. nesprávne nastavené prístupové kontrolné zoznamy (access control lists) na systémovom nástroji atď. Do tejto skupiny patria aj nástroje, ktoré detekujú neočakávané zásahy do obsahu alebo zabezpečujú ochranu súborov.[5]

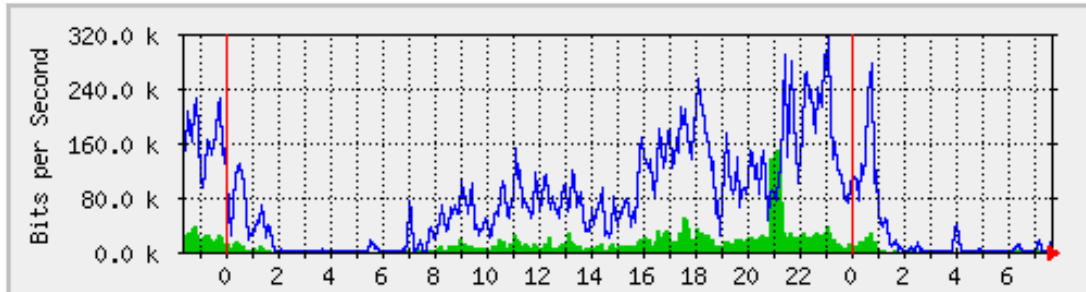
V ďalšej časti diplomovej práce je uvedená analýza vybraných dostupných monitorovacích nástrojov a ich vlastností.

3.1 MRTG (The Multi Router Traffic Grapher)

MRTG je nástroj na monitorovanie záťaže siete. Výsledkom MRTG sú HTML stránky. MRTG využíva zdrojový jazyk Perl a C. MRTG je podporovaný operačným systémom Linux ako aj Windows. Používa SNMP protokol, aby zistil zaťažiteľnosť všetkých zariadení, ale aj všetky informácie získané cez SNMP. Z týchto informácií získava dáta, z ktorých potom vytvára grafy.

MRTG umožňuje vytvárať grafy denné, za posledných 7 dní a za posledných 12 mesiacov. Tieto zápisy sú automaticky upravované tak, aby nedošlo k prekročeniu množstva dát za časovú jednotku, ale aj napriek tomuto obmedzeniu MRTG obsahuje všetky dáta týkajúce sa zaťažiteľnosti siete za posledné dva roky. Môže monitorovať viac ako 200 liniek súčasne. MRTG môže monitorovať akúkoľvek variáciu SNMP protokolu. Je dokonca možné použiť aj externé programy na zhromažďovanie dát, ktoré majú byť monitorované. MRTG sa používa na monitorovanie zaťažiteľnosti systému, dosiahnuteľnosti modemu atď. V jednom grafe je potom možné skombinovať dva alebo viaceré zdroje. Tieto grafy sú potom dané na webovú stránku, kde je ich možné prehliadať. [23]

Na Obr. 3.1 je zelenou farbou znázornená vstupná záťaž v bitoch za sekundu, modrou farbou výstupná záťaž v bitoch za sekundu.



Obr. 3.1: MRTG graf.

Výhody MRTG:

- je rozšíriteľný na viacero monitorovacích úloh
- podporuje množstvo programov, ktoré sú schopné monitorovať ďalšie činnosti
- výstup je ľahko pochopiteľný takmer pre každého
- poskytuje denný, týždenný, mesačný a ročný pohľad na zozbierané dáta

Nevýhody MRTG:

- každé spustenie zahŕňa vzorku dodaných dát a obnovenie webovej stránky
- neexistuje žiadna archivácia dát každej vzorky v čase
- ochrana je zabezpečená použitím Web Server mechanizmu
- na monitorovanie veľkého počtu zariadení (1000) je lepšie mať dostatočne rýchly disk
- na použitie externých monitorovacích skriptov potrebuje rýchly procesor a veľa pamäte

MRTG môže byť použitý v situáciách, kde existujú viacnásobné skupiny používateľov, ktorí prístupujú k dátam, ktoré potrebujú byť navzájom oddelené. Umožňuje hľadanie medzi zozbieranými dátami. MRTG zjednodušuje dáta, ktoré zozbieral a udržiava iba rok staré dáta.

MRTG je schopný vyberať vzorky odlišných dát buď na základe zariadenia alebo rozhrania. MRTG vyžaduje rozdielnú konfiguráciu pre každé premenlivé meranie na každom zariadení. Umožňuje volať externé programy, za predpokladu, že tieto programy realizujú výstup informácie vo formáte, ktorému MRTG rozumie; MRTG môže zakresľovať dané dáta do grafu.[3]

3.2 NAGIOS

Nagios je voľne šíriteľný monitorovací systém pre stanice, služby, siete, navrhnutý tak, aby podával informácie o problémoch v sieti predtým, než klienti, koncoví používatelia siete niečo zistia. Bol navrhnutý pre operačný systém Linux, ale pracuje rovnako dobre aj pod unixovskými variantmi. Obslužný program aktivovaný na základe monitorovania systémových požiadaviek (daemon) je zameraný na kontrolu hostov. Služby sú špecifikované na základe použitia externých pluginov, ktoré vracajú informácie pre NAGIOS. Súčasný stav informácií ako aj aktuálne správy sú zobrazované prostredníctvom web prehliadača.

Ak nastanú problémy, daemon môže poslať notifikáciu administrátorovi rozličným spôsobom (email, okamžitá správa, SMS, ...). Ak server zlyhá a dôležité služby nepripúšťajú žiadne zistenie príčiny, alebo ak výkon dát dosiahne kritickú hodnotu, potom je potrebné upozorniť administrátora siete prostredníctvom emailu alebo SMS, aby daný problém riešil, predtým než si ho používatelia zistia.

Nagios pomáha administrátorom rozpoznať problémy v sieti predtým ako chyba vznikne (proaktívny prístup), rýchlo ju odstrániť a významne zvýšiť dostupnosť siete pre uspokojenie používateľov. [10], [18]

NAGIOS poskytuje množstvo služieb, čo ho robí silným monitorovacím nástrojom:

- monitoruje sieťovú prevádzku (SMTP, POP3, HTTP, NNTP, PING, atď.)
- monitoruje zaťažiteľnosť procesora, využiteľnosť disku a pamäte, bežiacie procesy, atď.
- monitoruje environmentálne faktory ako je teplota

- jednoduchý návrh pluginu umožňuje používateľom ľahko rozvíjať ich vlastnú kontrolu hosta a prevádzky
- schopnosť definovať hierarchiu hostov v sieti, umožňujúcu detekciu hostov a rozlišovanie medzi hostami, ktoré sú neaktívne a tými, ktoré sú nedosiahnuteľné
- podporuje implementáciu redundantných monitorovacích serverov
- umožňuje rozoznať problémy prostredníctvom prehliadača
- jednoduchá autorizačná schéma umožňuje obmedziť používateľom prístupové práva prostredníctvom prehliadača
- zachovanie stavu hosta a prevádzky cez reštartovanie programu

Výhody NAGIOSu:

- monitoruje Windows, Unix aj Linux
- monitoruje ľubovoľné typy služieb
- dobre testovateľný vo veľmi zložitých scenároch
- pružne prispôsobiteľný a rozšíriteľný
- jasne usporiadané rozhranie prehliadača
- žiadne licenčné poplatky

Nevýhody NAGIOSu:

- vyžaduje určený server s veľkým množstvom pamäte
- konfigurácia je manuálny proces
- nie je to jednoduchý softvérový balík na konfigurovanie
- má jednoduchú inštaláciu, ale vyžaduje si veľa času
- neexistuje žiadna dobrá metóda založená na webových stránkach pre vytváranie monitoringu

NAGIOS aj MRTG sú silnými, voľne šíriteľnými monitorovacími nástrojmi, ktoré sú podporované operačným systémom Linux a môžu byť použité na monitorovanie zmiešaných sieťových platforiem, na ktorých súčasne existujú UNIX aj Windows. Kým MRTG slúži na monitorovanie najmä šírky pásma (bandwidth), ale môže sa tiež použiť na monitorovanie teploty, disku, procesora (CPU – Central

Processing Unit) atď., NAGIOS sa používa ako signalizačný systém pre činnosti ako sú vysoké zaťaženie, použitie diskového priestoru, neúspešné overenie dostupnosti siete (ping). MRTG sa tiež môže používať ako výstražný signál, ale nemá také dobré vlastnosti ako NAGIOS.[13]

3.3 NETFLOW MONITOR

Zatiaľ čo NAGIOS a MRTG sú voľne dostupnými monitorovacími nástrojmi, NetFlow Monitor využíva protokol Netflow firmy Cisco.

NetFlow Monitor je monitorovacím nástrojom na spracovanie a vyhodnocovanie „NetFlow Exports“. Všetky dáta (len niektoré polia z hlavičiek paketov) prechádzajúce cez smerovač alebo prepínač [17], sú odchytené v smerovači a po expirácii sú toky spolu spojené do „NetFlow Export“ UDP datagramov (User Datagram Protocol) pre vstup do zberača (collector).

NetFlow Monitor poskytuje monitorovanie skutočného času záťaže, rýchlu filtráciu toku (smart flow filtration), štatistické vyhodnotenie, mnohokritériový výber toku dát, používa zdrojové a cieľové IP adresy, protokoly, atď.

V rámci monitorovacieho nástroja NetFlow Monitor je tok (flow) chápaný ako neriadený tok dát definovaný siedmimi poliami: zdrojová IP adresa, cieľová IP adresa, L3 protokolový typ, zdrojový port, cieľový port, ToS bajt (DSCP), vstupné logické rozhranie. Ostatné polia v NetFlow sú určené pre prvý tok paketov alebo sú usporiadané do TCP flagov alebo sú zhrnuté do bajtov a paketov [17].

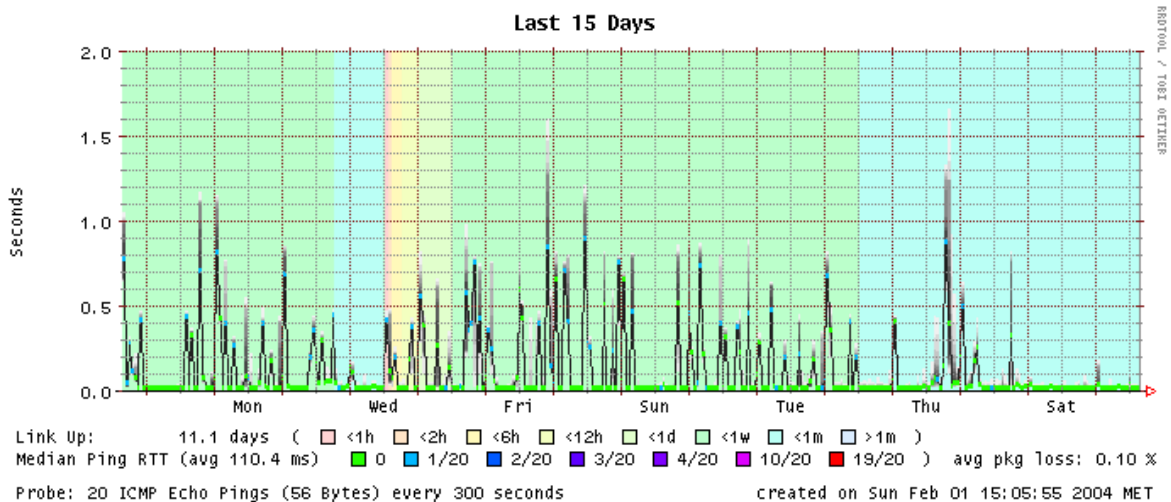
Súčasná verzia monitorovacieho nástroja NetFlow Monitor poskytuje rozšírený výpočet so spracovaním s maximálnou rýchlosťou prevádzky 40 Gb/s[17].

3.4 SMOKEPING

SmokePing sa používa na meranie oneskorenia, oneskorenia pri šírení a na meranie straty paketov v sieti. SmokePing používa RRD (Round Robin Database) nástroj na dlhodobé zachovanie uložených dát a vykreslenie grafov pre každé sieťové spojenie (Obr.3.2).

SmokePing sa tiež zaoberá zariadeniami, ktoré často menia svoje IP adresy, čím umožňujú vzdialeným hostom privolať pozornosť SmokePing na svoju novú adresu. V takomto nastavení SmokePing monitoruje ako dlho by si mohol vzdialený systém udržať svoju IP adresu. Na zabezpečenie toho, že nie je monitorovaný nesprávny host sa SmokePing pokúša okopírovať každý zo svojich dynamických IP cieľov prostredníctvom SNMP.

SmokePing je napísaný v jazyku Perl. Obsahuje daemon metódu zodpovednú za zbieranie dát a CGI skript, ktorý zobrazuje dáta na web. SmokePing má plugin architektúru, aby ľahko pridával nové schopnosti merania oneskorenia do SmokePingu. Pracuje na všetkých Unix platformách.[22]



Obr. 3.2: Graf SmokePing.

RRD nástroj

RRD je systém na ukladanie a zobrazovanie skupín dát za časovú jednotku. Skratka RRD je odvodená zo začiatočných písmen spojenia Round Robin Database. RRD ukladá dáta spôsobom, ktorý nenarastá za hranice časovej jednotky a spracované dáta prezentuje použitím grafov, aby uplatnil určitú hustotu dát. Môže byť používaný prostredníctvom jednoduchých skriptov jazyka Shell alebo Perl, alebo prostredníctvom frontends, ktoré komunikujú so sieťovými zariadeniami a vytvárajú používateľsky prístupné rozhranie.

Keďže nie je možné prijať dáta vždy v čase stanovenom používateľom, RRD nástroj dovoľuje používateľovi obnoviť log súbor (záznam o stave) v ľubovoľnom čase. Automaticky interpoluje hodnotu dátového zdroja na poslednom časovom úseku a zapisuje túto hodnotu do log súboru. Hodnoty dát rovnakého ustáleného nastavenia sú uložené v archíve Round Robin (RRA – Round Robin Archives). Je to veľmi účinný spôsob ukladania dát v určitom časovom rozsahu, pokiaľ sa používa známy rozsah pamäťového priestoru.[21]

Nevýhody RRD nástroja:

- nevytvára webové stránky alebo obrázky
- administrátor musí inštalovať skript na vytváranie obrázkov alebo musí nejaký sám napísať
- ťažko zlučuje dátové súbory
- nie je schopný dlho kopírovať riadky textu ako napr. MRTG

Výhody RRD nástroja:

- usporiadanie vstupných dát poskytuje dlhodobé ukladanie dát na redukovanom priestore disku
- binárne dátové súbory nenarastajú v čase

Po vytvorení RRD dátového súboru, používateľ špecifikuje ako dlho príslušný dátový bod v usporiadanom dátovom toku ostane v súbore.[34]

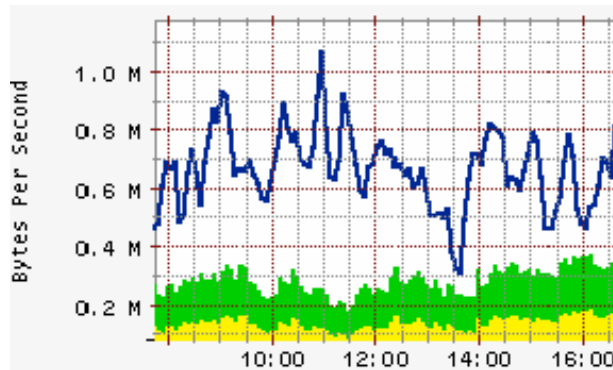
SmokePing je teda monitorovacím nástrojom, ktorý je využívaný predovšetkým na meranie najmä oneskorenia (latency). Používa RRD nástroj na udržiavanie dlhodobo uložených dát a na vykreslenie grafov, zobrazujúcich informácie každú minútu o stave každého sieťového pripojenia.

V porovnaní s MRTG má SmokePing vylepšenú webovú konfiguráciu, pretože popri konfiguračných súboroch, existuje tiež webová šablóna. Webová šablóna je jednoduchá webová stránka bez akéhokoľvek obsahu, ktorú možno aj editovať pomocou programu Photoshop a zobrazíť tak na stránke napr. vlastné logo, a do ktorej je možné umiestniť vlastné obrázky a obsah použitím programov ako napr. Dreamweaver alebo Front Page.

Najväčšou výhodou monitorovacieho nástroja SmokePing je to, že množina paketov získaná príkazom ping je vyjadrená jednoduchým grafom ako funkcia času týchto paketov[19].

3.5 CACTI

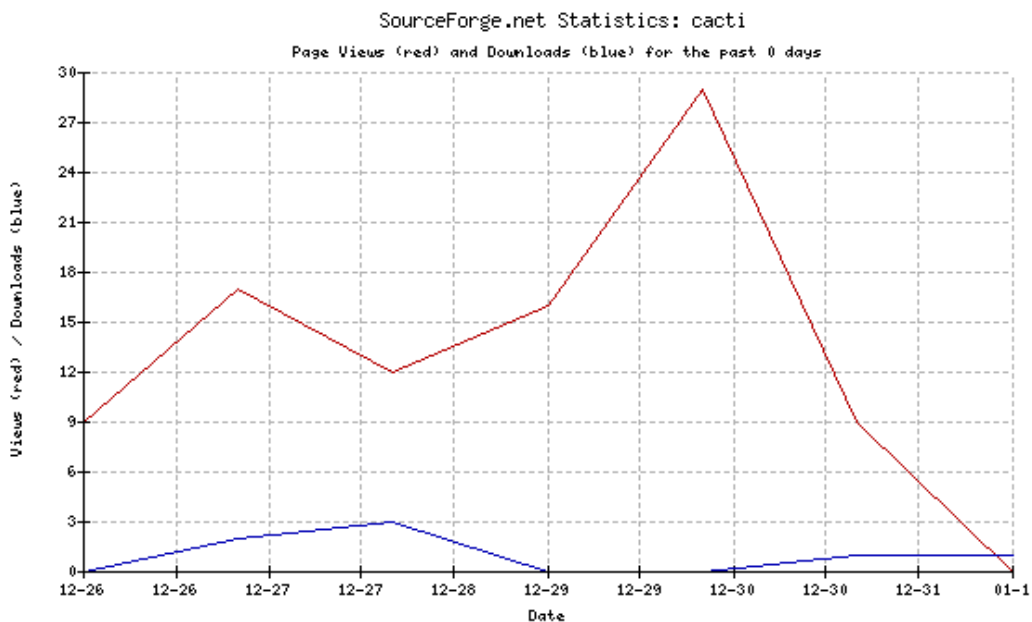
Jedným nedostatkom RRD nástroja je, že sám neposkytuje úplný súbor pre používateľa, ktorý by chcel vytvárať grafy pre vlastnú sieť. Na to je potrebný monitorovací nástroj Cacti. Cacti je kompletným riešením pre vytváranie sieťových grafov. Je navrhnutý tak, aby vedel použiť RRD nástroj pre ukladanie dát a ich následné vykreslenie vo forme grafov. Dáta sú ukladané do MySQL databázy (Obr. 3.3). Monitorovací nástroj Cacti využíva SNMP protokol.



Obr. 3.3: Graf Cacti.

Jeho výnimočnosť spočíva v používaní šablón, ktoré umožňujú zduplikovanie toho istého grafu. Je podporovaný operačným systémom Linux a Windows NT/2000. Programovacím jazykom je php.[4]

Webová stránka [24] poskytuje štatistické údaje o zaťažení stránky pri prezeraní a sťahovaní vo forme tabuľky a grafu. Je možné zobrazit' požadované údaje za posledných sedem dní, za posledný mesiac, alebo mesačne (Obr. 3.4). Modrou čiarou je vykreslená zaťažiteľnosť stránky počas jej prezerania a červenou pri ukladaní dát na disk.



Obr. 3.4: Štatistické zobrazenie.

Cacti okrem udržiavania grafov (Graphs), zdrojov dát (Data Sources) a archívov Round Robin (RRA – Round Robin Archives) v databáze, riadi aj zhromažďovanie dát.

Zdroj dát

Aby dáta mohli byť zhromažďované, Cacti umožňuje nastavenie cesty k akémukoľvek vonkajšiemu zdroju, z ktorého sa dáta, ktoré používateľ potrebuje na naplnenie databázy použijú. Ak sa vytvorí zdroj dát, je automaticky obnovovaný v 5 minútových intervaloch. Každý zdroj dát môže byť použitý na zhromažďovanie miestnych ale aj vzdialených dát.

Grafy

Ak je definovaný jeden alebo viac zdrojov dát, potom pomocou RRD nástroja môže byť vytvorený graf využívajúci tieto dáta. Cacti dovoľuje vytvoriť akýkoľvek graf, používajúci štandardné typy grafov RRD nástroja.

Cacti je vhodným monitorovacím nástrojom pre stredné až veľké LAN siete. Je dostatočne rýchlym monitorovacím nástrojom, pretože je napísaný aj v jazyku C a jeho výhodou je to, že využíva kompilovateľné SNMP knižnice. Cacti pomáha monitorovať použiteľnosť diskového priestoru v čase a umožňuje aj pozrieť s akou intenzitou je priestor používaný. V Cacti je implementovaná základná SNMP podpora, čo umožňuje tomuto monitorovaciemu nástroju vytvárať grafy podobné grafom generovaných pomocou MRTG.[32]

4. Implementácia štandardu IPFIX na príklade protokolu NetFlow

IPFIX (Internet Protocol Flow Information eXport) je štandardom, ktorý bol navrhnutý Technickou štandardizačnou skupinou pre internet (IETF – Internet Engineering Task Force). IETF vybrala pre štandard IPFIX protokol NetFlow verzie 9 firmy Cisco. IPFIX predstavuje základ pre štandardizovaný export toku IP dát. Tento štandard má uľahčiť sieťovým administrátorom získať operačné informácie, ktoré potrebujú na riadenie ich siete.[33]

4.1 Základné pojmy

Tok IP prevádzky (IP Traffic Flow) alebo **tok (Flow)** je definovaný ako sada IP paketov prechádzajúcich pozorovacím bodom v sieti počas určitého časového intervalu. Všetky pakety patriace k špeciálnemu toku majú množinu spoločných vlastností pochádzajúcich od dát zahrnutých v pakete a od spracovania paketov v pozorovacom bode. Každá vlastnosť je definovaná ako výsledok funkcie aplikovanej na niektorú z častí paketu. Ide o nasledujúce časti:

1. jedna alebo viac položiek hlavičky paketu (napr. cieľová IP adresa)
2. jedna alebo viac vlastností samotného paketu (napr. dĺžka paketu)
3. jedno alebo viacero polí pochádzajúcich od spracovania paketov (napr. AS číslo)

Tieto tri časti toku sa nazývajú kľúč toku (flow key).

Tok je definovaný siedmimi jedinečnými kľúčmi [7]:

- zdrojová IP adresa
- cieľová IP adresa
- zdrojový port
- cieľový port
- typ protokolu 3.vrstvy (Layer3 protocol type)

- ToS bajt (DSCP)
- vstupné logické rozhranie

Ak sa tok odlišuje aspoň jedným poľom od iného toku, označuje sa tento tok ako nový tok.

Zánik toku (Flow Expiration)

Tok je považovaný za neaktívny, ak už žiaden paket tohoto toku nebol zaznamenaný na pozorovacom bode v danom časovom intervale. [28], [30] Tok môže byť vyexportovaný v týchto prípadoch :

1. Ak exportér zistí koniec toku, potom by mal vyexportovať záznamy tokov (Flow Records).
2. Ak sa tok stane neaktívnym pre určitú periódu času. Tento časový interval by mal byť nastaviteľný meracím procesom. Prednastavená hodnota je 15 sekúnd.
3. Ak už prekračuje dĺžku trvania stanoveného intervalu – je teda dlhodobo bežiacim tokom. Prednastavená hodnota je 30 minút (1800 sekúnd).
4. Ak IPFIX zariadenie zaznamená nedostatok zdrojov, potom tok môže byť predčasne vyexportovaný.

Typ toku (Flow Type) je funkcia, ktorej vstupom je množina kľúčov toku a výstupom môže byť jeden alebo viacero tokov, ktoré závisia na kombinácii hodnôt pre množinu kľúčov toku.

Záznam toku (Flow Record) pozostáva z informácií o špecifickom toku, ktorý bol pozorovaný v pozorovacom bode. Obsahuje merané vlastnosti toku (napr. celkový počet bajtov všetkých paketov toku) a charakteristické vlastnosti toku (napr. zdrojová IP adresa). [30]

4.2 Pasívne merania

Celá koncepcia meracieho nástroja je založená na neintruzívnych (pasívnych) meraniach. Sú to merania, pri ktorých sa merajú charakteristiky počítačovej siete a parametre merania kvality služieb. Pasívne merania sú vykonávateľné len na základe aktuálnej prevádzky v sieti. Výsledky sú preto dobre interpretovateľné a využiteľné v praxi. Pri tomto type meraní nie sú prvky v sieti zaťažované dodatočnou prevádzkou. Nie je posielaný žiadny test prevádzky. To sa považuje za výhodu pasívnych meraní. Keďže neexistuje dodatočná prevádzka, neexistuje ani možnosť prípadného ovplyvnenia výsledkov merania.

Nevýhodou pasívnych meraní je to, že sú to neriaditeľné experimenty, pretože nevytvárajú špeciálnu testovaciu prevádzku. Danú prevádzku nie je možné ovplyvniť a nedajú sa teda prenášať ani riadiace dáta. Z tohto dôvodu je potrebné prídavné riadenie prevádzky pre prenos výsledkov merania. Pri meraní časových charakteristík (jednosmerného oneskorenia – one-way delay, kolísania oneskorenia – jitter) je potrebná externá synchronizácia hodín v jednotlivých meracích bodoch.

Pasívne merania sa používajú napr. na počítanie paketov a na určenie pridružených metrík ako napríklad intenzita. [25]

4.2.1 Architektúra pasívnych meraní

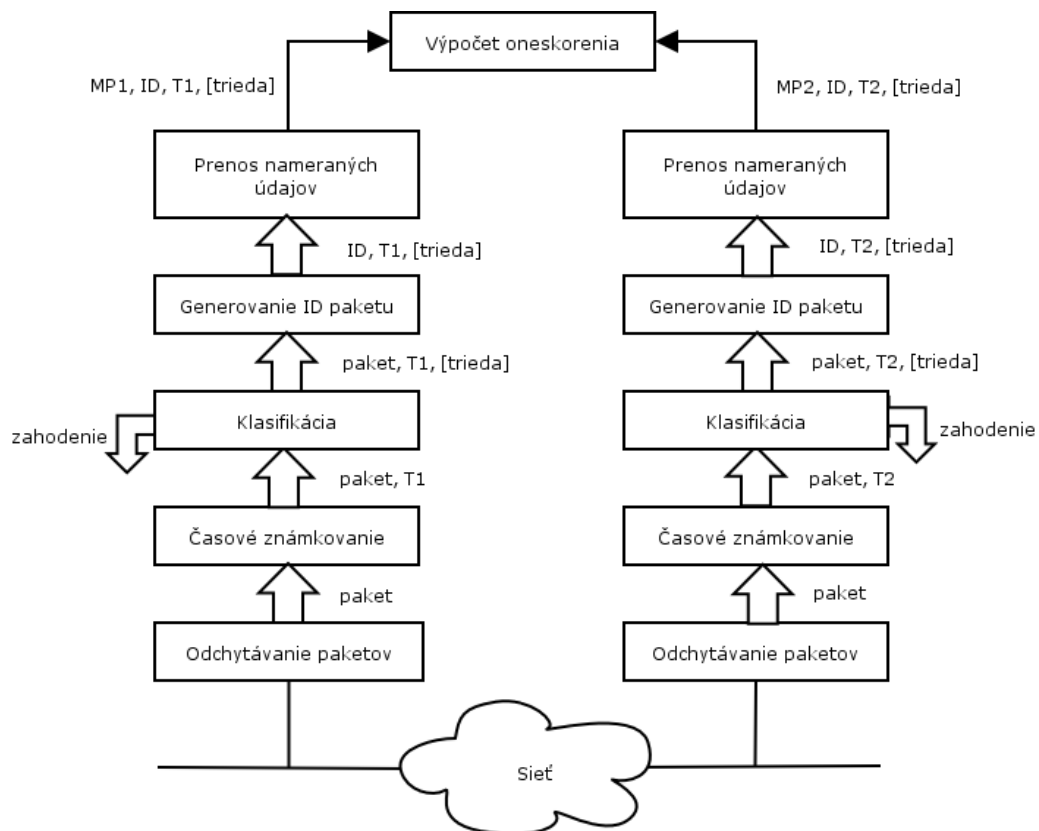
Pre implementáciu pasívnych metód merania časových charakteristík kvality služieb (QoS) - najmä oneskorenia, je potrebné zabezpečiť dva meracie body (MP1, MP2) na odmeranie času prechodu paketu.

Spôsob merania je založený na generovaní časových značiek a identifikátora ID paketu v oboch meracích bodoch a odoslani tejto informácie do kontrolného miesta na výpočet QoS parametrov. Určenie ID paketu je dôležité pre pričlenenie časových známok z rôznych meracích bodov k odpovedajúcemu paketu.

Procesy, ktorými paket prechádza, kým neodovzdá potrebné informácie do kontrolného miesta, kde sa vykoná výpočet sú:

- odchyťvanie paketov – je základom pre uskutočnenie merania, kedy potrebujeme odchytiť určité množstvo bajtov z paketu
- časové známkovanie (timestamping) – reprezentuje časové známky ako absolútne hodnoty času
- klasifikácia – je potrebná ak budeme merať len vybrané pakety. Výber paketov je vhodný pre zmenšenie času, ktorý je potrebný pri spracovaní ďalšími procesmi a pre zmenšenie množstva výsledných dát merania.
- generovanie ID – slúži na rozpoznávanie paketov
- prenos nameraných údajov – je potrebný pre zozbieranie výsledkov merania z jednotlivých meracích bodov

Architektúra pasívnych meraní je na Obr. 4.1.



Obr. 4.1: Architektúra pasívnych meraní.

4.3 Štandard IPFIX

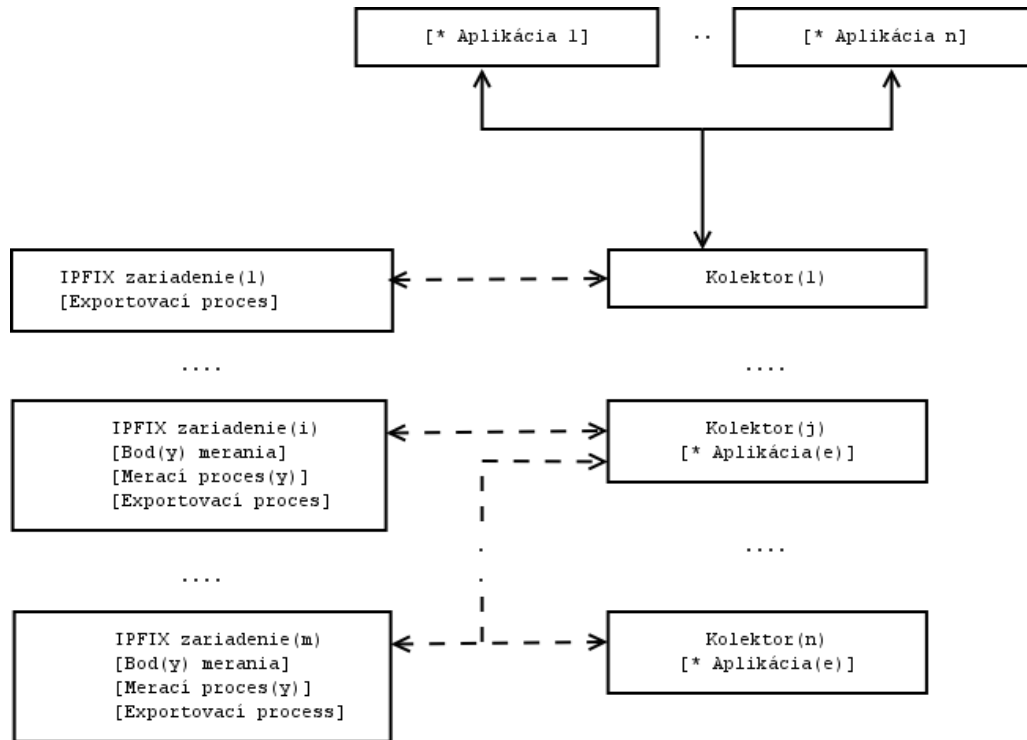
Vývoj štandardu IPFIX, ktorý začal približne pred dvoma rokmi sa zhoduje s vývojom verzie 9 protokolu NetFlow. NetFlow a teda aj IPFIX môžu exportovať informácie takmer akéhokoľvek dátového typu, čím sa zjednoduší monitorovanie aplikácií, ako napr. skupinové vysielanie (multicasting) [33].

IPFIX definuje formát, pri ktorom informácie o IP toku môžu byť prenášané z exportéra do kolektora. Aplikácie, ktoré podporujú IPFIX, dokážu zobrazit' štatistiky prijaté z akéhokoľvek smerovača, ktorý podporuje štandard IPFIX.

IPFIX je formátom pre export dát založený na šablónach. Použitie šablón je vhodné pre sieťových administrátorov, pretože nemusia modifikovať svoj softvér, aby podporoval nový formát zakaždým, keď sa administrátori rozhodnú pozrieť na prevádzkové štatistiky. [14]

Jednotlivé časti, ktoré popisuje štandard IPFIX sú rozobrané v nasledujúcich kapitolách.

Architektúra IPFIX štandardu je na Obr.4.2. Hviezdičkou (*) sú označené tie časti, ktoré nie sú súčasťou IPFIX štandardu. Rozhrania zobrazené čiarkovanou čiarou sú definované v IPFIX štandarde, rozhrania zobrazené plnou čiarou definované nie sú.



Obr. 4.2: Architektúra IPFIX.

4.3.1 Kolektor

Kolektor prijíma záznamy tokov (flow records) z jedného alebo viacerých exportérov. Môže upraviť alebo uložiť prijatý záznam toku (flow record). Kolektor je subsystém, ktorý je vo vzájomnej interakcii s jedným alebo viacerými IPFIX zariadeniami. Funkcie kolektora môžu zahŕňať[20]:

- identifikáciu, akceptovanie a dekodovanie vyexportovaných paketov z exportovacieho procesu a z pozorovacej domény (observation domain)
- spustenie IPFIX protokolu
- uloženie riadiacich informácií (control information) a záznamov tokov (flow records) prijatých z IPFIX zariadenia
- oznámiť stav a problémy IPFIX zariadenia

Kolektor prijme definíciu šablóny od exportéra ešte pred prijatím záznamov tokov. Záznamy tokov môžu byť dekodované a lokálne uložené na zariadeniach. V prípade, že definície šablón neboli prijaté v čase prijatia záznamu toku, kolektor by mal udržať záznam toku pre neskoršie dekodovanie dovedy, kým nebude prijatá definícia šablóny.

Kolektor nesmie predpokladať, že FlowSet dát a pridružené ID šablón sú exportované v tom istom vyexportovanom pakete.

Kolektor nesmie predpokladať, že iba jedna šablóna FlowSetu je prítomná vo vyexportovanom pakete. V zriedkavom prípade, vyexportovaný paket môže obsahovať niekoľko šablón FlowSetov.

Šablóny existujú len určitý časový interval. Životnosť šablóny by mala byť odpočítaná (zistená) na kolektore na základe času, kde posledná šablóna FlowSetu bola prijatá z exportéra. Kolektor sa nesmie pokúšať dekodovať záznamy tokov s vypršanou platnosťou šablóny. Kolektor by mal udržiavať takýto zoznam: <exportér, exportné rozhranie, ID šablóny, definícia šablóny, posledné prijatie>.

Ak je prijatá nová definícia šablóny (napr. v prípade reštartovania exportéra), existujúca definícia by mala byť okamžite nahradená.

4.3.2 Bod merania

Bod merania je miesto v sieti, kde môžu byť pozorované IP pakety. Môže to byť linka, ku ktorej je pripojená sonda, zdieľané médium ako napr. LAN založené na Ethernete, jednoduchý port smerovača alebo množina rozhraní na smerovači.

4.3.3 Merací proces

Merací proces generuje záznamy tokov (Flow Records). Vstupom do meracieho procesu sú hlavičky paketov zaznamenané v bode merania. Merací proces pozostáva z množiny funkcií, ktorá obsahuje záznamy tokov (Flow Records) o ich zachytávaní, generovaní časových známkov, vzorkovaní, klasifikovaní a spracovaní hlavičky.[20]

4.3.4 Exportovací proces

Exportovací proces odosiela záznamy tokov na jeden alebo viac zhromažďovacích procesov. Záznamy tokov sú generované jedným alebo viacerými meracími procesmi. Exportovací proces musí byť schopný poskytovať nasledujúce informácie o každom meranom toku:

- číslo verzie internetového protokolu
- zdrojová IP adresa
- cieľová IP adresa
- typ IP protokolu (TCP,UDP,ICMP,...)
- v prípade typu protokolu TCP alebo UDP – zdrojový port
- v prípade typu protokolu TCP alebo UDP – cieľový port
- počítadlo paketov
- počítadlo bajtov
- slabika typu služby (Type of Service – ToS)
- v prípade IP verzie 6 – návestie toku
- v prípade podpory špeciálnych multiprotokolových návestí (MPLS) – prvé návestie
- časová známka prvého paketu
- časová známka posledného paketu
- jednoznačný identifikátor bodu merania
- jednoznačný identifikátor exportovacieho procesu

4.3.5 Zhromažďovací proces

Zhromažďovací proces (collecting process) prijíma záznamy tokov z jedného alebo viacerých exportovacích procesov. Exportovací proces môže vykonávať ďalšie spracovanie záznamov tokov [12].

Zhromažďovací proces by mal prijímať záznamy dát bez spojenia so záznamom šablón. Ak záznamy šablón neboli prijaté v čase prijatia záznamov dát, zhromažďovací

proces by mal uložiť záznamy dát na krátky časový interval a dekodovať ich potom ako budú prijaté záznamy šablón. Časový interval uloženia záznamov dát musí byť menší ako životnosť šablóny.

Životnosť šablóny zhromažďovacieho procesu je obmedzená na pevne stanovené obnovenie vypršaného časového intervalu (fixed refresh timeout). Zhromažďovací proces musí byť spojený so životnosťou každej prijatej šablóny prostredníctvom UDP. Šablóny, ktoré nie sú obnovené exportovacím procesom v rámci časového intervalu, sú expirované v zhromažďovacom procese. Ak šablóna nie je obnovená exportovacím procesom predtým než vyprší časový interval, zhromažďovací proces musí vyradiť šablónu a rovnako aj všetky súčasné a budúce záznamy dát, ktoré sú s danou šablónou spojené.

V každom čase by mal zhromažďovací proces udržiavať nasledujúci formát pre všetky súčasné záznamy šablón a záznamy voľby šablón [30]: <exportovací proces, zdrojové ID pozorovacej domény (observation domain), ID šablóny, definícia šablóny, posledné prijatie> .

Dátová sieť s IP prevádzkou pozostáva z IP tokov prechádzajúcich cez prvky siete. Zhromažďovací proces IPFIXu by teda mal byť schopný prijať informáciu toku (flow information) prechádzajúcu cez viaceré prvky siete v rámci dátovej siete. To si vyžaduje jednotnosť v metódach, ktoré reprezentujú informáciu toku (flow information) a spôsob komunikácie tokov od prvkov siete ku zbernému bodu (collecting point). IPFIX protokol poskytuje prístup k informácii IP toku (IP flow information).

4.3.6 Pozorovacia doména

Množina pozorovacích bodov, ktoré sú najväčšou zoskupenou množinou informácií tokov (flow information) na IPFIX zariadení, sa nazýva pozorovacia doména (observation domain). Pozorovacia doména poskytuje jedinečné ID kolektoru na identifikáciu exportovaných paketov, ktoré sú ním generované.

Pozorovacia doména môže byť prepojená s tým istým exportovacím procesom. Napr. pozorovacou doménou môže byť router line-card, ktorý pozostáva z viacerých rozhraní, kde každé rozhranie predstavuje pozorovací bod.[28]

4.4 IPFIX správa

IPFIX správa je správa pochádzajúca z exportovacieho procesu, ktorá nesie IPFIX záznamy daného exportovacieho procesu, a ktorej cieľom je zhromažďovací proces. IPFIX správa je zapuzdrená v rámci transportnej vrstvy.

Hlavička správy je prvá časť IPFIX správy, ktorá poskytuje základné informácie o správe ako napr. verzia IPFIXu, dĺžka správy, poradové číslo správy, ...

Set (sada) je termín pre množinu/súbor záznamov, ktoré majú podobnú štruktúru. V IPFIX správe môže za hlavičkou správy nasledovať jeden alebo viac setov(sád). Existujú 3 typy sád:

- Template Set
- Options Template Set
- Data Set

Obsah typu sád je zobrazený v nasledujúcej tabuľke:

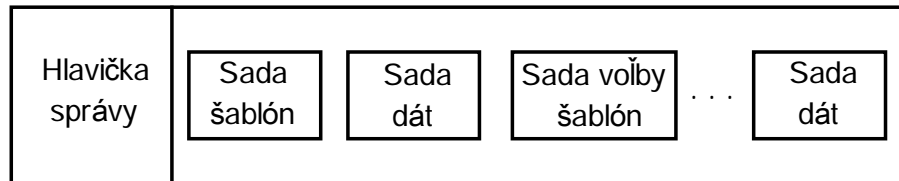
Set	Obsah	
	Šablóna	Záznam
Data Set	x	Záznamy dát
Template Set	Záznamy šablón	x
Options Template Set	Záznamy voľby šablóny	x

Tab. 4.1: Obsah typu sád.

Data Set je zložený zo záznamov dát, nezahŕňa však žiaden záznam šablóny. Záznam šablóny alebo záznam voľby šablóny (Options Template Record) definuje záznam dát. Template Set obsahuje iba záznamy šablón. Options Template Set pozostáva iba zo záznamov voľby šablóny (Options Template Record(s)) [30].

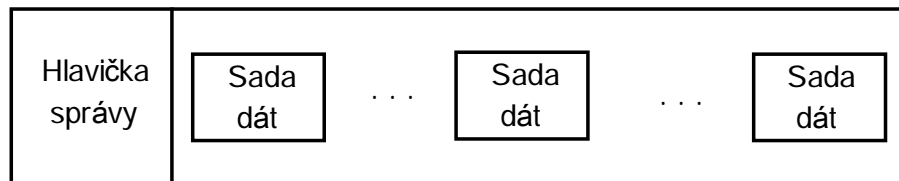
Príklady formátu IPFIX správ[30]:

1. IPFIX správa pozostáva zo sady šablón, dát a voľby šablón – Novo vytvorená šablóna je exportovaná tak skoro, ako je to možné. Takže, ak existuje IPFIX správa so sadou dát, ktoré sú pripravené na export, sada šablón a voľby šablón nasledujú za touto informáciou.



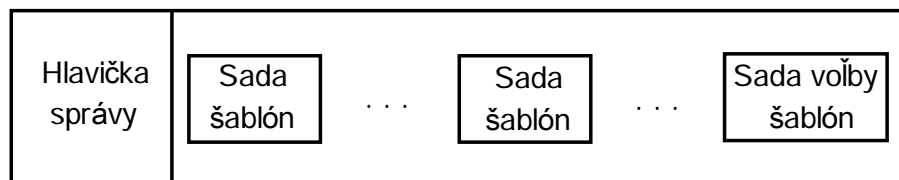
Obr. 4.3: Formát IPFIX správ, 1. príklad.

2. IPFIX správa pozostáva iba zo sady dát – Po tom ako príslušný záznam šablóny bol definovaný a poslaný zhromažďovaciemu procesu, väčšia časť IPFIX správy pozostáva zo sady dát.



Obr. 4.4: Formát IPFIX správ, 2. príklad.

3. IPFIX správa pozostáva len zo sady šablón a voľby šablón – Ak je ako transportný protokol použitý UDP protokol, sada šablón a voľby šablón musia byť posielané pravidelne, aby zabezpečili, že zhromažďovací proces obsahuje záznamy šablón a voľby šablón, keď odpovedajúce záznamy dát sú prijímané.



Obr. 4.5: Formát IPFIX správ, 3. príklad.

Na IPFIX zariadení môže byť funkčnosť protokolu rozdelená medzi pozorovaciu doménu (observation domain) a exportovací proces. IPFIX protokol zabezpečuje na IPFIX zariadení:

1. kódovanie riadiacej informácie (control information) do šablón
2. kódovanie tokov zadržaných na meracom bode do záznamov tokov
3. pakovanie záznamu toku a/alebo riadiacej informácie (control information) do exportovaných paketov založených na exportovacej taktike
4. použitie transportnej vrstvy na poslanie exportovaných paketov do kolektora

IPFIX protokol na kolektore je zodpovedný za[20]:

1. prijatie a uloženie riadiacej informácie (control information)
2. dešifrovanie a uloženie záznamov tokov použitím riadiacej informácie (control information)

IPFIX zariadenie je zariadenie obsahujúce aspoň jeden pozorovací bod, vymeriavací proces a exportovací proces. Obyčajne odpovedajúci pozorovací bod, vymeriavací proces a exportovací proces sú spoločne umiestnené na tomto zariadení, napr. na smerovači.

Funkcia IPFIX protokolu v zariadení IPFIX je na strane exportovacieho procesu. IPFIX protokol plní úlohu, ktorej cieľom je získať toky z procesu zaznamenávania tokov alebo ich získať priamo od meracieho procesu a prenáša ich ku zberaču.

IPFIX protokol spravuje:

- výber a posielanie kontrolných informácií a záznamov o tokoch
- kódovanie kontrolných informácií o tokoch
- expiráciu tokov
- správanie pri preťažení
- selektívny export záznamov o tokoch

IPFIX protokol vykonáva nasledujúce funkcie:

- kódovanie vybraných kontrolných informácií do šablón
- kódovanie tokov pozorovaných v pozorovacom bode do záznamov o tokoch
- použitie transportnej vrstvy na posielanie exportovaných paketov zberaču
- spracovanie preťaženia IPFIX zariadenia
- aplikácia selektívnych filtrov

4.5 NetFlow verzia 5

NetFlow je implementáciou špecifikácie IPFIX firmy Cisco. Existuje viacero verzií protokolu NetFlow (verzia 1, 5, 7, 8, 9). V rámci tejto diplomovej práce sa budem zaoberať verzou 5 ale najmä 9, pretože práve táto verzia NetFlowu je kompatibilná so štandardom IPFIX (viac kap.3.3).

Toky sú vo verzii 5 vždy identifikované tou istou množinou pevných atribútov. Zhromaždené údaje sú asynchrónne exportované a potom odstránené z pamäti tokov (flow cache). Údaje sú exportované cez NetFlow Export UDP datagramy, ktoré pozostávajú maximálne z 30 záznamov tokov (flow records). Tieto datagramy sú exportované aspoň raz za sekundu alebo ihneď, keď je k dispozícii UDP datagram ukončeného toku. Položky pamäte môžu expirovať kvôli jednej z nasledujúcich príčin:

- uplynul limit doby existencie toku (interval 1 až 60 minút, štandardne 30 minút),
- uplynul čas nečinnosti toku – pozorovacím miestom neprešiel žiaden paket patriaci do toku (interval 10 až 600 sekúnd),
- pamäť tokov je plná a potrebuje byť zaznamenaný nový tok, bol signalizovaný koniec toku (TCP FIN príznak).

Formát toku pre NetFlow verziu 5[7]:

Použitie	Počet paketov
	Počet bajtov
Čas dňa	Začiatok sysUpTime
	Koniec sysUpTime
Využitelnosť portu	Vstupný ifIndex
	Výstupný ifIndex
QoS	Typ služby
	TCP návestie (flag)
	Protokol
Z / Do	Zdrojová IP adresa
	Cieľová IP adresa
Aplikácia	Zdrojový TCP/UDP port
	Cieľový TCP/UDP port
Smerovanie	Adresa nasledujúceho skoku
	Zdrojové AS číslo
	Cieľové AS číslo
	Zdrojová prefix maska
	Cieľová prefix maska

Tab. 4.2: Formát toku NetFlow v5.

4.6 NetFlow verzia 9

NetFlow verzia 9 [15] je zovšeobecnená verzia Cisco NetFlow protokolu. Predchádzajúce verzie NetFlowu, najmä verzia 5, boli široko implementované a používané pre exportovanie a zbieranie informácií o IP tokoch.

Pretože všetky predchádzajúce verzie protokolu NetFlow (verzia 1, 5, 7, 8) neboli flexibilné a prispôsobiteľné, vznikla nová verzia protokolu. Verzia 9 slúži ako exportovací formát (Export Format). Je nezávislá na základnom protokole, je vhodná pre akýkoľvek spoľahlivý protokol, ako napr. TCP, SCTP. [7]

NetFlow verzia 9 je na rozdiel od verzie 5 pri definícii toku založená na šablónach. Šablóny predstavujú rozšíriteľný návrh pre exportovaný paket. Táto vlastnosť umožňuje budúce rozšírenia bez potreby zmeny základných vlastností formátu záznamu tokov.

Šablóna (Template) je usporiadaná n-tica (napr. <type, length>, TLV), použitá na určenie štruktúry a sémantiky určitej informácie, ktorá komunikuje s IPFIX

zariadeniami smerom na kolektor. Každá šablóna je jedinečne identifikovateľná nejakým spôsobom (napr. použitím Template ID).

Použitie šablón má niekoľko výhod:

- NetFlow je odolný voči zmenám nových alebo vyvíjajúcich sa protokolov, pretože je možné jednoducho dodať podporu pre tieto protokoly
- nové vlastnosti môžu byť k NetFlow protokolu pridané jednoducho bez znefunkčnenia existujúcich implementácií
- aplikácie pre zhromažďovanie alebo analýzu dát môžu byť jednoduchým spôsobom (zmenou šablóny) obohatené o tieto nové vlastnosti

Exportovaný paket sa v protokole NetFlow verzia 9 skladá z dátových položiek, položiek definujúcich šablóny a z položiek nastavujúcich parametre pre zhromažďovací proces. [12]

NetFlow spolu s exportérom posiela šablóny (Templates), voľby (Options) a dáta (Data) množiny tokov (FlowSets) kolektoru. Množina tokov je postupnosť záznamov dát rovnakého formátu. NetFlow je jediným prípadom protokolu pracujúcim pod UDP. Kvôli jednoduchej nesmerovej (unidirectional) povahe protokolu, mohol by relatívne priamo pridávať mapovanie k iným transportným protokolom ako SCTP (Stream Control Transmission Protocol – Protokol určený pre transport signalizačných správ cez IP sieť) alebo TCP (Transmission Control Protocol – Protocol na riadenie prenosu).[15]

5. Konceptia architektúry základného meracieho nástroja "BasicMeter"

5.1 Architektúra meracej platformy

Merací nástroj vychádza z návrhu architektúry štandardu IPFIX. Podľa tohto štandardu je architektúra meracej platformy zložená z troch častí. Niekedy sa zvykne do tejto architektúry zakresľovať aj SQL databáza, lenže tá nevyjadruje predstavu meracej platformy podľa štandardu IPFIX.

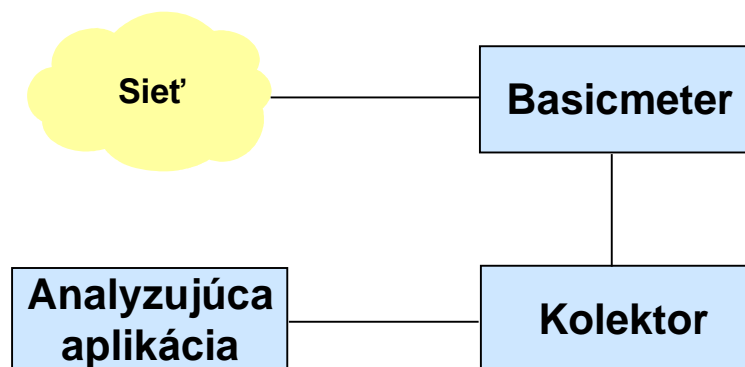
Architektúra meracej platformy je teda zložená z týchto častí:

Basicmetera je to aplikácia, ktorá slúži ako merací proces. Je určená na zachytávanie paketov. Zároveň vytvára aj dáta pre zhromažďovací proces.

Kolektora predstavuje zhromažďovací proces. Slúži na spracovanie exportovaných paketov z exporovacieho procesu.

Analyzujúcej aplikácie – má prístup k exportovaným dátam. Na požiadanie používateľa vykonáva grafickú alebo štatistickú analýzu.

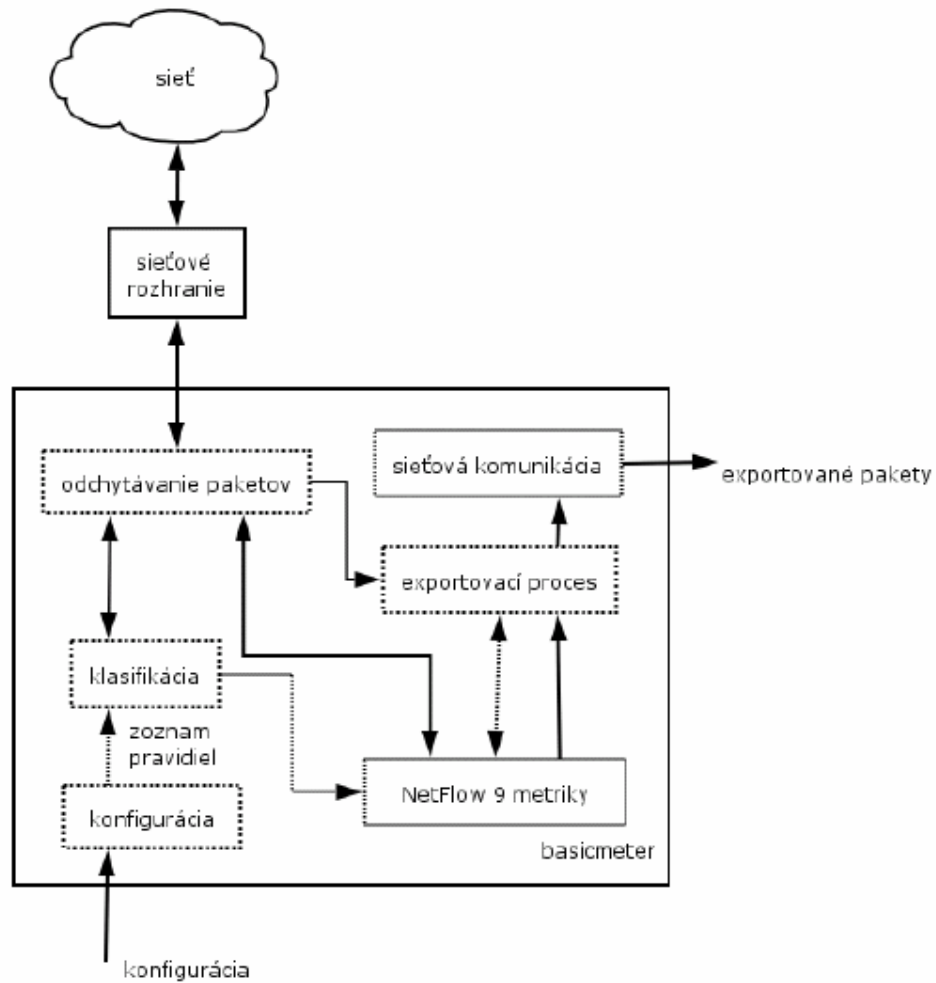
Architektúra meracej platformy podľa štandardu IPFIX je zobrazená na nasledujúcom obrázku.



Obr. 5.1: Architektúra meracej platformy.

5.2 Analýza architektúry BasicMetra

Jednoduchá koncepcia implementácie základného meracieho nástroja (BasicMeter) je znázornená na Obr.5.2.



Obr. 5.2: Architektúra meracieho nástroja.

Konfigurácia

Konfigurácia je časť, ktorá načítava a spracováva vstup konfiguračných parametrov aplikácie a to buď z príkazového riadku alebo z textového konfiguračného súboru.

Konfigurácia využíva dve knižnice:

- libconfuse - knižnica pre spracovanie textových konfiguračných súborov [11]
- libpopt

Táto časť zároveň slúži ako inicializačný prvok pre ostatné časti.

Klasifikácia

Úlohou klasifikátora je nájsť taký dátový tok, do ktorého patrí práve spracovávaný paket. Požadovanú informáciu o identifikátore toku odovzdáva do časti, kde sa daný tok vytváral, teda do časti NetFlow 9 metriky.

Výhodou klasifikácie je znižovanie množstva dát pre vyhodnotenie merania a znižovanie času, ktorý je potrebný na spracovanie ďalšími časťami meracieho bodu.

V klasifikátore je implementovaný algoritmus lineárneho vyhľadávania. Je založený na spojkovom zozname, ktorý slúži na uchovávanie pravidiel. Ukladanie pravidiel prebieha od najvyššej priority po najnižšiu. V algoritme je využité sekvenčné porovnávanie pravidiel. Vybrané pravidlo musí vyhovovať všetkým poliam v hlavičke IP paketu.

Výhoda algoritmu lineárneho vyhľadávania spočíva v jeho jednoduchosti, pamäťovej efektívnosti. Nevýhodou je jeho zlá škálovateľnosť a čas klasifikácie paketu rastie lineárne s počtom pravidiel.

Odchytávanie paketov

Časť odchytávania a sledovania paketov (capture) sa zaoberá sledovaním a odchytávaním paketov na základe parametrov spracovaných konfiguračnou časťou. V procese inicializácie konfiguračnej časti sú tieto parametre odovzdané odchytávacej časti. Odchytávané sú všetky pakety, ktoré patria aspoň do jedného toku.

Časť odchytávania a sledovania paketov využíva knižnicu libpcap, ktorá bola vybraná vďaka svojej podpore na rôznych unixových operačných systémoch a aj na platforme Win32 ale len v obmedzenej verzii.

Jadrá unixových operačných systémov obsahujú jednoduchý paketový filter, s ktorým spolupracuje aj zvolená knižnica libpcap, čím sa proces filtrovania paketov zjednodušuje. Knižnica teda podporuje odchytávanie paketov a ich prenos z priestoru jadra do priestoru používateľa.

Medzi metódy znižovania počtu paketov patrí filtrovanie, vzorkovanie, hašovanie.

Filtrovanie je deterministický výber paketov založený na obsahu paketu, jeho spracovaní alebo deterministickej funkcii vyskytujúcej sa vo fáze výberu.

Hašovanie je založené na hašovacej funkcii, ktorá mapuje časť paketu (hlavička/telo) do pevne stanovenej veľkosti premennej. Hašovacia funkcia je nevratná.

Vzorkovanie popisuje systematický alebo náhodný výber podmnožiny elementov (vzorka) z množiny všetkých elementov (rodičovská populácia). Cieľom vzorkovania je výber reprezentatívnej vzorky. Merací proces môže podporovať vzorkovanie. V konfigurácii meracieho procesu však musí byť jednoznačne definovaná podpora vzorkovania.

Konfigurácia vzorkovania zahŕňa metódu vzorkovania a všetky potrebné parametre. Ak sa konfigurácia vzorkovania zmení počas meracieho procesu, všetky zhromažďovacie procesy musia byť informované o zmenách, ktoré nastali v konfigurácii. Zmena vzorkovania v meracom procese znamená odobratie vzorkovacej funkcie, pridanie novej vzorkovacej funkcie, zmenu parametrov a zmenu vzorkovacej metódy.[26]

Pre odchyťvanie paketov sa používajú rôzne metódy vzorkovania:

1. systematické vzorkovanie – popisuje proces výberu štartovacích bodov a interval výberu podľa deterministickej vzorky. V rámci systematického vzorkovania rozlišujeme:
 - systematické vzorkovanie založené na počte
 - systematické vzorkovanie založené na čase
2. náhodné vzorkovanie – určuje začiatkové body vzorkovacích intervalov v závislosti od náhodného procesu. Medzi metódy náhodného vzorkovania patrí:
 - poissonove vzorkovanie
 - geometrické vzorkovanie

Metriky protokolu NetFlow 9

Metriky protokolu NetFlow 9 predstavujú časť, ktorá slúži na napĺňanie dátových tokov. Tieto dátové toky boli získané z pozorovaných informácií na základe spracovania položiek v jednotlivých šablónach. Toky sú vytvárané v exportovacej časti.

Exportovacia časť

Exportovacia časť (exportovací proces) je samostatná časť programu. Slúži na spracovanie dát získaných odchyťvacou časťou sledovaním procesu a zároveň odosiela vytvorené toky do časti sieťovej komunikácie. V exportovacej časti sú pre správnosť zadania načítané a verifikované (validované) šablóny.

Exportovacia časť bola navrhnutá tak, aby podporovala protokol NetFlow verzie 9 aj s možnosťami vytvárania šablón.

Prenos exportovaných paketov prebieha prostredníctvom UDP protokolu (User Datagram Protocol).

Sieťová komunikácia

Časť sieťovej komunikácie predstavuje abstrakčnú vrstvu medzi expotortovacou časťou a transportnými protokolmi. Používaným protokolom je UDP (User Datagram Protocol).

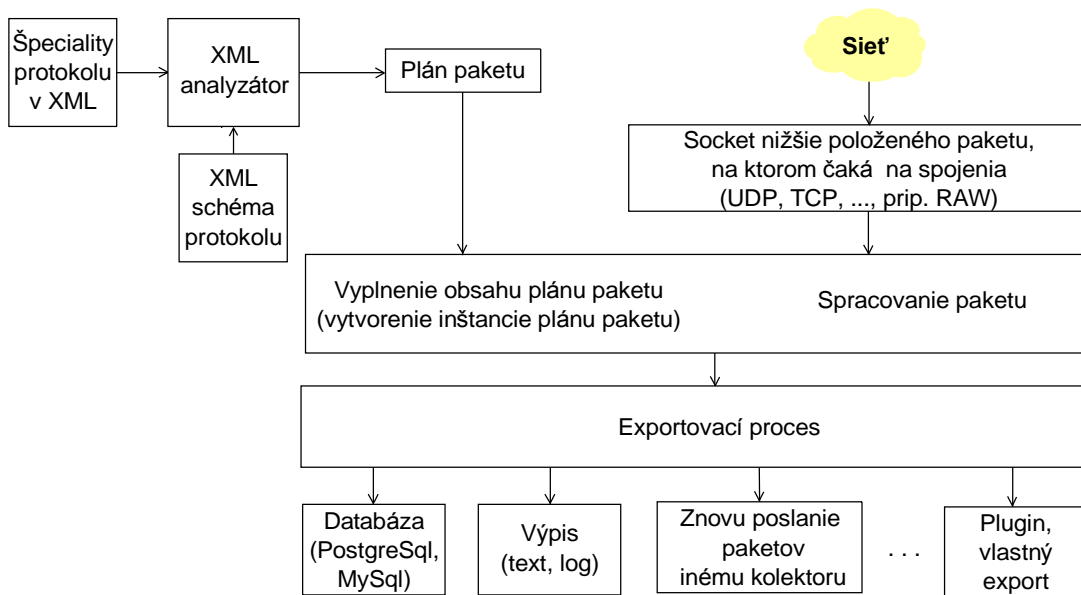
5.3 Analýza architektúry kolektora

Na označovanie a extrahovanie potrebných dát slúži generujúci program, ktorý na základe špecifikácie vytvorí plán dekompozície paketu. Nad vybranými dátami sa potom vykoná požadovaná operácia. Základom sú dobre definované akcie, ktoré by mali spĺňať úvahy o formálnom modelovaní protokolov.

Pri tejto metóde sa využíva schopnosť XML pre popis dát, ktoré by mohli prísť. XML dokument teda slúži na popis jednotlivých akcií a cieľov pre daný element. V XML dokumente sa dátam priradia akcie. Tento proces by sme mohli nazvať mapovanie (mapping).

Potrebné dáta sa získavajú z prichádzajúcich paketov. V skutočnosti XML súbor mapuje akcie priradené k príslušným častiam paketu. Zo zásady je vrstva návrhu oddelená od vrstvy spracovania paketu, takže funkcie ktoré by mali byť vytvorené nad poliami sú vytvorené nad štruktúrou dát. XML slúži ako konfiguračný alebo modelovací jazyk pre softvér, ktorý vytvára aplikácie a pracuje s daným protokolom.

Blokový diagram spracovania paketu je na nasledujúcom obrázku.



Obr. 5.3: Blokovaná schéma spracovania paketu v diagrame.

6. Návrh a implementácia „animačného modelu architektúry“ meracieho nástroja v prostredí Macromedia Flash MX

6.1 Úvod

Jedným z cieľov diplomovej práce je navrhnuť a vytvoriť animačný model architektúry meracej platformy na báze BasicMetra. V rámci predošlých diplomových prác [2], [31] boli rozobrané a špecifikované problémy súvisiace s projektom BasicMeter. Výstup tejto DP má slúžiť nielen pre potrebu lepšieho chápania modelu vyvíjanej meracej platformy na báze BasicMetra, ale aj pre potreby vzdelávania – názornej prezentácie princípov neintruzívnych meraní a celého meracieho nástroja, s jeho jednotlivými časťami ako aj s jednotlivými meraniami.

6.2 Výber SW nástrojov pre realizáciu animačného modelu

Animačný model architektúry bol navrhnutý a vytvorený v programe Macromedia Flash MX. Pre realizáciu bol flash vybraný kvôli jeho prezentačným možnostiam. Pomocou Flash MX je možné vytvárať aplikácie, s ktorými môže používateľ komunikovať prostredníctvom tlačidiel, ovládania myšou a pod. Flash je vhodným a vďačným nástrojom pre vývojárov, pretože implicitne pracuje s vektorovou grafikou, ako aj s bitovými mapami.

Vo flashi je možné definovať viacero výstupných formátov:

- Flash (.swf)
- HTML (.html)
- GIF obrázkov (.gif)
- JPEG obrázkov (.jpg)

- PNG obrázok (.png)
- Windows Projektor (.exe)
- Macintosh Projektor
- Quick Time (.mov)

Je potrebné rozlišovať program, v ktorom sa samotný flash tvorí teda Flash MX a prehrávač, ktorý je schopný daný flash zobrazit' a prehrať. Pre flash MX je prehrávačom Flash Player verzie 6. [9]

6.3 Návrh animačného modelu architektúry BasicMetra

Celá aplikácia je zložená z 3 samostatných flash súborov. Prvý súbor, hlavný, ktorý je spustiteľný (teda má príponu .exe), má funkciu prepínača jazykov. Program je pre univerzálnosť použitia animačného modelu vytvorený v dvoch verziách, v slovenskej a v anglickej. Obe verzie ako aj hlavná časť aplikácie sú vygenerovanými súbormi s príponou .swf.

Vytvorenú aplikáciu je možné spustiť súborom s príponou .exe (start.exe). Otvorí sa okno prehrávača Flash Player 6. Na disk je ale potrebné si uložiť nielen súbor s príponou .exe, ale aj súbory s príponou .swf, pretože na tieto súbory sa spustiteľný program počas svojho behu odkazuje.

Samostatný návrh animačného modelu je rozdelený do dvoch častí:

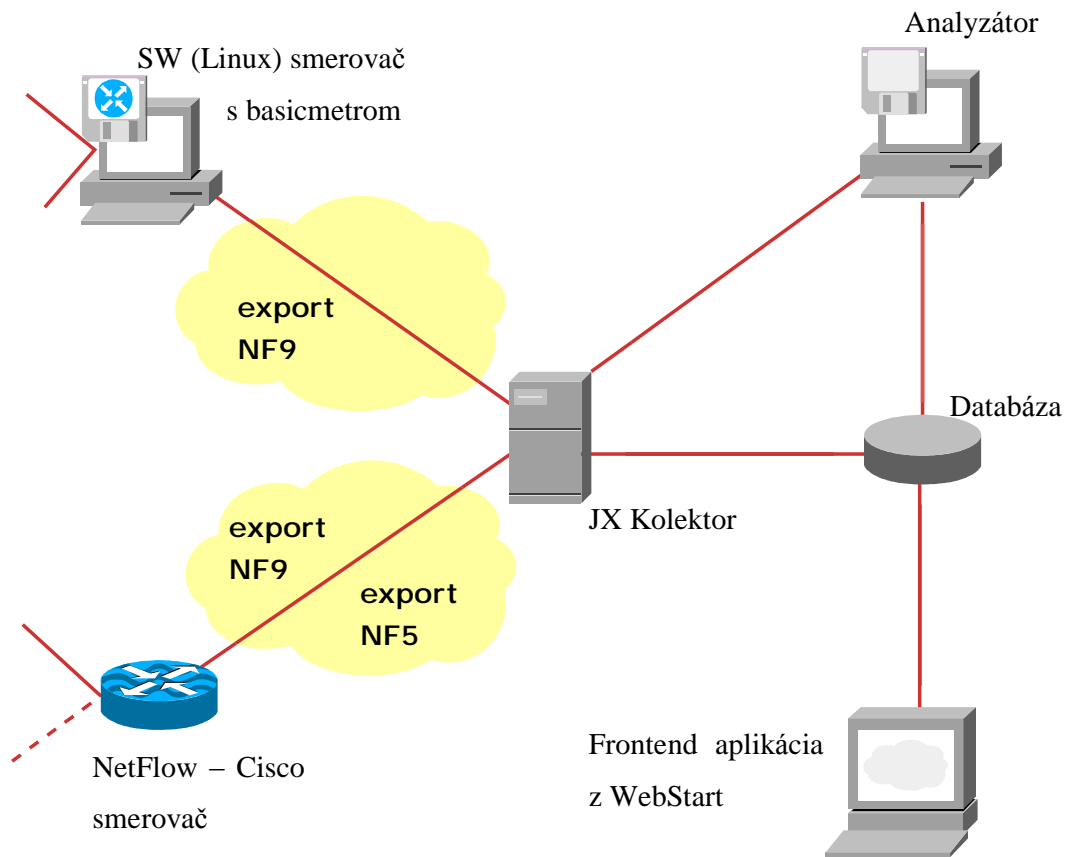
- meracia platforma
- meranie QoS parametrov

Obe časti poskytujú dostatok informácií pre pochopenie architektúry meracej platformy, ako aj princípov jednotlivých meraní parametrov QoS. Časti, ktoré sú náročné na pochopenie sú prezentované vo forme pohyblivých animácií, kde je krok po kroku vysvetlený princíp fungovania.

6.3.1 Meracia platforma

V názve meracia platforma je zahrnutá celá architektúra meracej platformy a architektúra meracieho nástroja (popísané v kapitole 5.1 a 5.2) a koncept experimentu.

Základom celej animácie vytvorenej vo flashi je schéma samotného meracieho nástroja



Obr. 6.1: Model meracieho nástroja.

6.3.1.1 Analýza jednotlivých častí schémy

Merací proces, t.j. export NF9

Merací proces pozostáva z množiny funkcií, ktorá okrem iného zahŕňa zachytávanie hlavičky paketu, označovanie časom (timestamping), vzorkovanie (sampling), klasifikáciu, a správu záznamov o tokoch. Tento postup je rovnaký pre každý paket, ktorý prejde cez daný SW smerovač.

Ak uplynie časový limit, toky sa vyexportujú. Exportujú sa len toky, ktoré sa považujú za expirované. Pri exporte tokov je nutné exportovať šablónu (Template), na základe ktorej potom zberací proces vie dekodovať dáta, ktoré mu prídu (inak ich zahodí).[25]

Zberací proces a kolektovanie do databázy

Kolektor zberá dáta od jedného a viacerých exportérov, predspracuje ich a archivuje v databáze (ide o SQL dotazy na databázu).

Zberací proces a priame spojenie

Pre pseudo merania v reálnom čase sa pripojenie na databázu obchádza a vytvára sa spojenie priamo s analyzátorom, kde sa posielajú len dáta potrebné na vyhodnocovanie daného merania (na základe dohody medzi kolektorom a analyzátorom).

Analyzátor a databáza

Analyzátor z databázy vyberá potrebné údaje SQL dotazmi pre zobrazenie výsledkov meraní.

6.3.1.2 Analýza funkcionality a jednotlivých komponentov architektúry

Informácie o BasicMetri

Aplikácia je ovládateľná z príkazového riadku. Jazyk C++ bol vybraný ako hlavný implementačný jazyk pre jeho rozsiahle použitie pod OS UNIX a platformovej nezávislosti. S ohľadom na maximálnu programovú prenositeľnosť nebudú použité žiadne neštandardné funkcie jadra OS.

Aplikácia je logicky rozdelená do dvoch častí:

- časť pre odchyťvanie a monitorovanie paketu
- exportovacia časť (exportovací proces)

Časť pre odchyťvanie a monitorovanie paketu použitá pre tento účel je dobre známa knižnica libpcap. Táto knižnica bolo vybraná pre jej širokú podporu na rôznych OS. Knižnica libpcap pracuje s jednoduchou filtráciou paketu, ktorá je zahrnutá takmer v každom jadre OS UNIX.

Exportovací proces je nezávislou časťou programu. Spracúva dáta odchytené odchyťvacou časťou. Táto časť bola navrhnutá tak, aby mohla podporovať NetFlow verziu 9 s voľbou vytvorenia šablóny. Prenos vyexportovaných paketov je riadený UDP protokolom.

Informácie o kolektore

Skratka JXColl je odvodená od Java XML Collector. Je to program, ktorý sa stará o zber tokov (informácie o prúdoch údajov, Netflow 5/9) z jedného alebo viacerých bodov exportu. Jeho hlavnou úlohou je pripraviť dáta na ďalšie spracovanie a prezentáciu koncovou užívateľskou aplikáciou (analyzátorom). Dokáže spracovávať dáta od viacerých exportérov vysielajúcich UDP pakety s tokmi.

Vykoná zopár jednoduchých akcií s obsahom tokov a pripraví dáta na uloženie do archívov alebo ich pošle priamo analyzátorovi pre okamžitý výstup používateľovi. Zatiaľ bol testovaný s voľne prístupnými SRBD (systém riadenia bázy dát) ako napr. PostgreSQL a MySQL avšak s istými problémami. Dizajn databázy a nízke rozdielne rozdistribúovanie dát prispeli k slabej výkonnosti, z ktorej vyplynuli dlhé čakania pri

požiadavkách na dáta. Z tohto dôvodu analyzátor potreboval viac času na vytvorenie výstupov v prijateľnom čase. Preto sa vytvorilo priame spojenie medzi analyzátorom a kolektorom. Optimalizuje dotazy a posiela iba dáta potrebné pre danú meraciu metódu.

JXColl je vyvíjaný ako Java konzolová aplikácia; XML značí možnosť rozpoznávania a využívania obsahu prichádzajúcich paketov na základe ich popisu pomocou XML.

Analyzátor

Hlavným cieľom je poskytnúť používateľovi rozhranie a výstup (grafy, štatistiky, atď.). Je vyvinutý v jazyku Java pre používateľské potreby ako aplikácia spracovaná použitím technológie Java Web Start.

Má dve základné charakteristiky:

- schopnosť spustiť aplikáciu z web stránky (hoci JRE s Web Start je potrebné)
- verzie programu sú vždy aktualizovateľné

Analyzátor môže sprístupniť databázy tokov a pripojiť sa priamo do zberacieho procesu. V súčasnosti podporuje PostgreSQL a MySQL databázy. Bezpečnosť je overená pri pripojení, aj databáza aj kolektor vyžadujú platný login a heslo. Predstavuje tok dát pre danú metódu v grafickej podobe buď ako statický výstup (analýza histórie trafiky), alebo ako pseudo reálne časové meranie.

Výstup môže byť filtrovaný na základe niekoľkých výberov (IP adresa, číslo portu a označkovanie časom – timestamp). V tomto bode sa využívajú metódy meraní. Súčasne plne využíva metódy pre sledovanie šírky pásma a počtu paketov. Práca na dĺžke paketov, použití portu, jednosmernom oneskorení, kolísaní oneskorenia a strate paketu je vo vývoji.

Môžu sa vyvolať akcie založené na udalostiach, ako napr. upovedomenie, log, ... V zásade všetky ďalšie časti takéhoto podsystému len pripravujú dáta pre inteligentnú aplikáciu, ktorá je pripravená spracovať ich a vyprodukovať odpovede alebo výstupy.

Databáza

Pod databázou rozumieme množinu dát, ktoré nejakým spôsobom spolu súvisia. Na uloženie množiny dát – exportovaných paketov sa predpokladá akýkoľvek druh databázy (súbor, databáza pracujúca so súborovým systémom, vyhradená partícia disku pre uloženie neformátovaných dát, atď.)

Hoci SQL databáza nie je súčasťou špecifikácie IPFIX ani protokolu NetFlow verzie 9, bola zvolená ako databázový sklad pre jej jednoduché použitie, dobré možnosti ďalšieho spracovania uložených dát a pre jej dobré možnosti získavania uložených dát.

Príklad šablóny FlowSet

Formát záznamu NetFlow verzie 9 sa skladá z hlavičky paketu. Za ňou nasleduje aspoň jeden alebo viacej FlowSetov šablón alebo dát. Šablóna FlowSetu poskytuje popis polí, ktoré budú prítomné v budúcich dáta FlowSetoch. Tieto dáta FlowSet sa môžu vyskytnúť neskôr v rámci toho istého vyexportovaného paketu alebo v nasledujúcich vyexportovaných paketoch. Šablóny FlowSetov sú jedným z kľúčových prvkov v novom formáte NetFlow verzie 9. Šablóny vysoko zvyšujú flexibilitu formátu záznamu NetFlow, pretože umožňujú, aby NetFlow kolektor alebo zobrazovacia aplikácia spracovali NetFlow dáta bez toho, aby poznali formát dát vopred. Šablóny sa používajú na popísanie typu a dĺžky individuálnych polí v rámci záznamu dát NetFlow, ktoré sa hodia k ID šablóny.[6]

Netflow Version 9 Header (32 bits)		Template FlowSet (16 bits)		PROT (0x0004)	
Version 9	Count = 1 (FlowSet)	FlowSet ID = 0		Length = 1	
System Uptime		Length = 80 bytes		_TOS (0x0005)	Length = 1
Unix Seconds		Template ID = 257		Length = 1	L4_PORT (0x0007)
Package Sequence		Field Count = 18		Length = 2	L4_DST_PORT (0x000B)
Source ID		LAST_SWITCHED (0x0015)		Length = 2	IPv4_NEXT_HOP (0x000F)
		Length = 4		Length = 4	Length = 4
		FIRST_SWITCHED (0x0016)		DST_MASK (0x000D)	Length = 1
		Length = 4		_MASK (0x0009)	Length = 1
		BYTES_32 (0x0001)		TCP_FLAGS (0x0006)	Length = 1
		Length = 4		Length = 1	DST_AS (0x0011)
		PKTS_32 (0x0002)		Length = 2	Length = 2
		Length = 4		Length = 2	_AS (0x0010)
		INPUT_SNMP(0x000A)		Length = 1	Length = 2
		Length = 2			
		OUTPUT_SNMP(0x000E)			
		Length = 2			
		IPv4_ADDR (0x0008)			
		Length = 4			
		IPv4_DST_ADDR (0x000C)			
		Length = 4			

Obr. 6.2: Príklad šablóny FlowSet.

Význam jednotlivých polí je vysvetlený v nasledujúcich tabuľkách:

Názov poľa	Význam
Version	Verzia NetFlow záznamov vyexportovaná v danom pakete, táto hodnota je 0x0009.
Count	Počet záznamov FlowSet (šablón a dát) obsiahnutých v rámci daného paketu.
System Uptime	Čas uvádzaný v milisekundách od začiatku prvého bootovania zariadenia.
UNIX Seconds	Sekundy od 0000 riadené univerzálnym časom (UTC) 1970
Package Sequence	Inkrementačné sekvenčné počítadlo všetkých exportovaných paketov poslaných daným exportovacím zariadením. Táto hodnota je narastajúca a môže byť použitá na identifikovanie, či chýbajú nejaké exportované pakety.
Source ID	Toto pole má 32 bitovú hodnotu, ktorá garantuje jedinečnosť pre všetky toky exportované z príslušného zariadenia.

Tab. 6.1: Popis formátu hlavičky paketu protokolu NetFlow v9.

Názov poľa	Význam
FlowSet ID	Používa sa na rozlíšenie záznamov šablón a dát. Záznam šablóny má vždy FlowSet ID v rozsahu 0-255. Záznam dát má vždy nenulové FlowSet ID väčšie ako 255.
Length	Vzťahuje sa na celkovú dĺžku daného FlowSetu. Keďže samostatný FlowSet môže obsahovať viackrát to isté ID šablóny, dĺžka by mala byť použitá na určenie pozície ďalšieho záznamu FlowSetu, ktorý by mohol byť buď šablóna alebo dáta FlowSet. Dĺžka je vyjadrená vo formáte typ/dĺžka/hodnota (TLV).
Template ID	Keď smerovač generuje odlišné šablóny FlowSetov, aby prispôbil typ dát NetFlowu, ktoré majú byť exportované, každej šablóne je pridané jedinečné ID. Táto jedinečnosť sa vzťahuje na smerovač, ktorý generoval ID šablóny. Šablóny, ktoré definujú formáty záznamu dát sa číslujú od 256. 0-255 je rezervované pre FlowSet ID.
Field Count	Udáva počet polí v zázname šablóny. Pretože šablóna FlowSetu môže obsahovať viacero záznamov šablón, toto pole umožňuje analyzátoru určiť koniec súčasného záznamu šablóny a začať ďalší záznam.
Field Type	Táto číselná hodnota reprezentuje typ poľa. Možné hodnoty tohto poľa sú určené obchodníkmi. Hodnoty dodané Cisco sú zhodné so všetkými platformami, ktoré podporujú NetFlow verziu 9. Súčasne definované typy poľa sú popísané v tab. 6.3
Field Length	Toto číslo udáva dĺžku poľa Field Type v bajtoch.

Tab. 6.2: Popis šablóny FlowSetu protokolu NetFlow v9.

Field Type	Hodnota	Dĺžka v bajtoch	Popis
LAST_SWITCHED	21	4	Doba prevádzky systému, pri ktorej bol posledný paket toku presunutý
FIRST_SWITCHED	22	4	Doba prevádzky systému, pri ktorej bol prvý paket toku presunutý
BYTES_32	1	N (implicitne 4)	Vstupné počítadlo s dĺžkou N x 8 bitov pre počet bajtov spojených s IP tokom
PKTS_32	2	N (implicitne 4)	Vstupné počítadlo s dĺžkou N x 8 bitov pre počet paketov spojených s IP tokom
INPUT_SNMP	10	N (implicitne 4)	Index vstupného rozhrania, implicitná hodnota pre N je 2, ale môže byť použitá aj vyššia hodnota
OUTPUT_SNMP	14	N (implicitne 4)	Index výstupného rozhrania, implicitná hodnota pre N je 2, ale môže byť použitá aj vyššia hodnota
IPv4_ADDR	8	4	IPv4 zdrojová adresa
IPv4_DST_ADDR	12	4	IPv4 cieľová adresa
PROT	4	1	Bajt pre IP protokol
_TOS	5	1	Bajt pre typ služby (Type of Service) je nastavený, ak sa zapisuje vstupné rozhranie
L4_PORT	7	2	TCP/UDP číslo zdrojového portu, napríklad FTP, Telnet alebo podobné
L4_DST_PORT	11	2	TCP/UDP číslo cieľového portu, napríklad FTP, Telnet alebo podobné
IPv4_NEXT_HOP	15	4	IPv4 adresa nasledujúceho skoku smerovača
DST_MASK	13	1	Počet susedných bitov v cieľovej adrese masky podsiete, t.j. submaska v slash (/) notácii
_MASK	9	1	Počet susedných bitov v zdrojovej adrese masky podsiete, t.j. submaska v slash (/) notácii
TCP_FLAGS	6	1	Narastanie všetkých TCP flagov viditeľných pre daný tok
PST_AS	17	N (implicitne 2)	Číslo cieľového BGP autonómneho systému, kde N môže byť 2 alebo 4
_AS	16	N (implicitne 2)	Číslo zdrojového BGP autonómneho systému, kde N môže byť 2 alebo 4

Tab. 6.3: Popis poľa Field Type protokolu NetFlow v9.

Príklad dáta FlowSet

Šablóny a dáta FlowSetu môžu byť zmiešané v rámci jedného vyexportovaného paketu. Keď sa interpretuje formát dát FlowSetu protokolu Netflow verzie 9, nemôžu byť polia rozobrané bez odpovedajúceho ID šablóny. Ak je prijatý dáta FlowSet, ktorý nemá vhodné ID šablóny, záznam by mal byť zahodený. ID šablóny v zázname šablóny mapuje ID FlowSet do odpovedajúceho dáta FlowSetu. Usporiadanie údajov v zázname údajov sa mapuje do formátov polí určených v zázname šablóny. Záznamy dát nie sú nevyhnutne predchádzané ich odpovedajúcou šablónou v rámci vyexportovaného paketu.[6]

Netflow Version 9 Header (32 bits)		Data FlowSet (32Bits)	
Version 9	Count = 1 (FlowSet)	FlowSet ID = 257	Length = 80 bytes
System Uptime		147.232.22.132	
Unix Seconds		62.168.11.131	
Package Sequence		0.0.0.0	
Source ID		9	
		2056	
		147.232.22.132	
		62.168.11.131	
		0.0.0.0	
		8	
		924	
		147.232.22.132	
		62.168.11.131	
		0.0.0.0	
		5	
		1057	

Obr. 6.3: Príklad FlowSetu dát.

Význam jednotlivých polí je vysvetlený v nasledujúcej tabuľke:

Názov poľa	Význam
FlowSet ID = Template ID	FlowSet ID nadradzuje každú skupinu záznamov v rámci NetFlow v9 FlowSetu dát. FlowSet ID sa mapuje do ID šablóny. Kolektor a zobrazovacie aplikácie by mali používať FlowSet ID, aby mapovali príslušný typ a dĺžku do akýchkoľvek hodnôt poľa, ktoré nasledujú.
Length	Toto pole udáva dĺžku FlowSetu dát. Dĺžka je vyjadrená vo formáte TLV. To znamená, že hodnota zahŕňa bajty použité pre FlowSet ID a dĺžku samotných bajtov ako aj dĺžku zloženú z akýchkoľvek zahrnutých záznamov dát.
Record N - Field N	Zvyšok FlowSetu dát je zbierkou hodnôt poľa. Typ a dĺžka polí bola preddefinovaná v zázname šablóny odkazovanej pomocou FlowSet ID resp. pomocou ID šablóny.
Padding	Výplň by mala byť vložená, aby zarovnala koniec FlowSetu na 32 bitovú hranicu. Je dôležité si uvedomiť, že dĺžka poľa bude zahrňovať aj tieto bity.

Tab. 6.4: Popis FlowSetu dát protokolu NetFlow v9.

6.3.2 Meranie QoS parametrov

Prezentácia postupov merania QoS parametrov je druhou časťou animačného modelu. Pomocou flash animácie je názorne zobrazený priebeh jednotlivých meraní, ktorý vysvetľuje princíp činnosti, merania a vyhodnocovanie jednotlivých typov meraní QoS parametrov.

Táto diplomová práca je zameraná na tri druhy meraní QoS parametrov:

- jednosmerné oneskorenie (one-way delay)
- kolísanie oneskorenia (jitter)
- spätočné oneskorenie (RTT – Round Trip Time)

Jednosmerné oneskorenie

Jednosmerné oneskorenie (one-way delay) je čas potrebný k odoslaniu paketu a jeho prijatiu v cieľi [26]. Jednosmerné oneskorenie patrí medzi tie časové charakteristiky, na meranie ktorých sa používajú dva meracie body.

Skladá sa z dvoch častí:

- času potrebného na prenesenie paketu cez fyzické médium, čo je funkcia prenosovej rýchlosti linky
- času, ktorý predstavuje oneskorenie spôsobené radením do front, spracovaním v sieťových zariadeniach a preťažením liniek

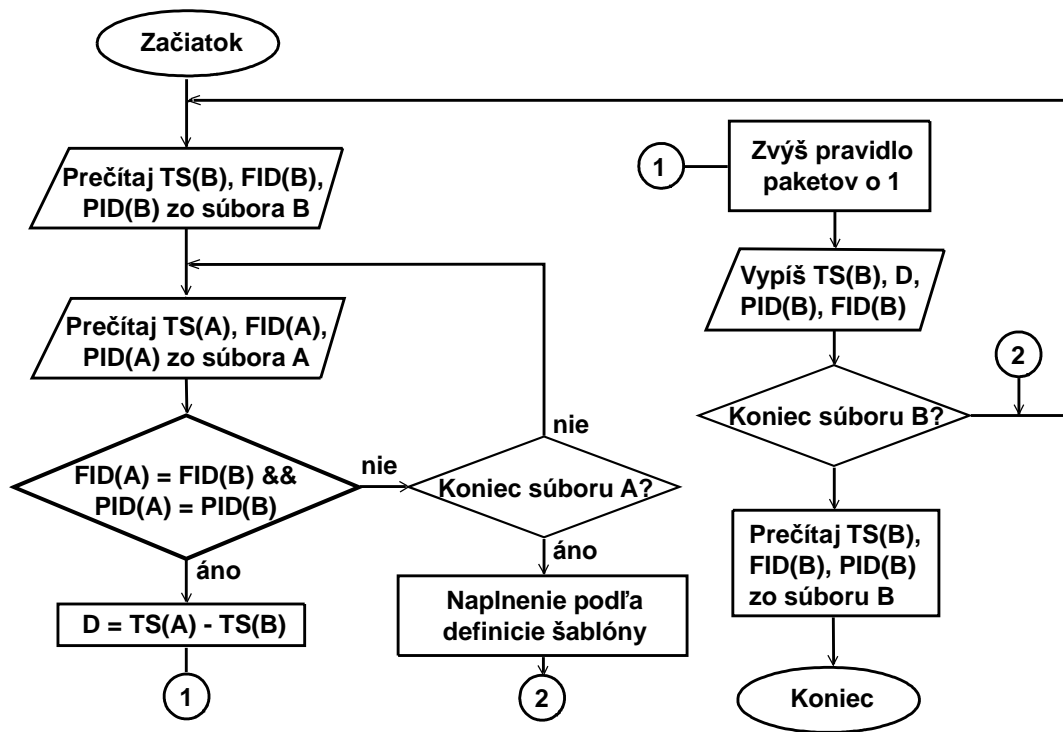
Výpočet jedinečného identifikátora ID sa vykonáva na základe MD5 algoritmu [29]. MD5 je kryptografická hašovacia funkcia určená na generovanie 128 bitového odtlačku správy ľubovoľnej dĺžky. Používa sa na autentifikáciu správ.

Princíp merania jednosmerného oneskorenia spočíva v prechode paketu prvým meracím bodom, vygenerovania jednoznačného identifikátora paketu, vygenerovania časovej známky udávajúcej čas prechodu paketu a odoslanie získaných údajov na zberný bod.

Samotné oneskorenie paketu je definované ako rozdiel príchodu paketu do meracích bodov: $\Delta T = T_2 - T_1$

Keďže na meranie jednosmerného oneskorenia sa používajú dva meracie body, je potrebné zabezpečiť presnú synchronizáciu času na jednotlivých meracích bodoch a identifikovať paket. Táto metóda sa opiera o pasívne merania. Z toho dôvodu nie je možné modifikovať paket tak, aby obsahoval údaje o vygenerovanom identifikátore resp. o časovej známke, ktorá k nemu prislúcha. Tieto údaje by sa mali uchovávať vo vopred definovanom zbernom bode. V praxi nie je nutné, aby tento bod bol fyzicky odlišný od meracieho bodu, môže to byť s ním totožný proces.

Algoritmus výpočtu jednosmerného oneskorenia je na nasledujúcom obrázku.



Obr. 6.4: Algoritmus výpočtu jednosmerného oneskorenia.

Pre algoritmus na výpočet jednosmerného oneskorenia sú potrebné dva súbory BasicMetra. Obidva súbory obsahujú dáta v rovnakom formáte a sú výsledkom merania v dvoch bodoch.

Po spustení sa vykonáva vonkajší cyklus čítania riadku zo súboru B, až kým sa neprečíta celý súbor. Riadok obsahuje časovú značku (TS), identifikátor toku (FID), identifikátor paketu (PID). Vnútorňý cyklus pozostáva z čítania riadku zo súboru A.

Potom sa porovná FID a PID z oboch súborov. Ak sú rovnaké, tak sa vypočíta oneskorenie D ako rozdiel časových značiek ($D = TS(A) - TS(B)$), zvýši sa počítadlo paketov a na výstup sa vypíše TS(B), PID, FID.

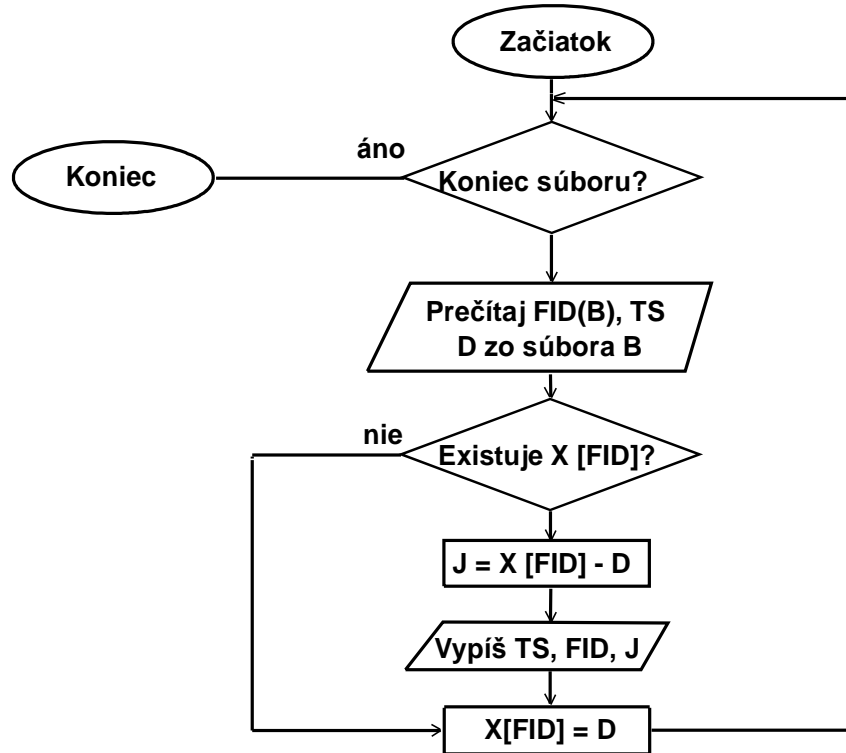
Ak nie sú rovnaké, tak sa číta ďalší riadok zo súboru A a opakuje sa porovnanie. Ak sa narazí na koniec súboru A, zvýši sa počítadlo stratených paketov a pokračuje sa čítaním ďalšieho riadku zo súboru B.

Po dosiahnutí konca súboru B sa vypíšu počítadla paketov a stratených paketov a program sa ukončí.

Kolísanie oneskorenia

Kolísanie oneskorenia (delay variation, jitter) je definované pre dva pakety ako rozdiel medzi jednosmerným oneskorením jedného paketu a jednosmerným oneskorením druhého paketu.[8]

Algoritmus výpočtu kolísania oneskorenia je na Obr. 6.5



Obr. 6.5: Algoritmus výpočtu kolísania oneskorenia.

Ako vstup algoritmu pre výpočet kolísania oneskorenia sa používa výstup z programu pre výpočet oneskorenia.

Po spustení sa začne vykonávať cyklus čítania riadku zo súboru. Po prečítaní riadku obsahujúceho časovú známku, identifikátor toku (FID) a oneskorenia (D), sa kontroluje existencia prvku poľa X s indexom FID (X[FID]).

Ak existuje, tak sa vypočíta kolísanie oneskorenia J ako rozdiel $X[\text{FID}] - D$ a vypíše sa na výstup spolu s TS a FID. Potom sa uchová hodnota D do prvku X[FID].

Cyklus pokračuje kontrolou konca súboru a čítaním ďalšieho riadku. Ak neexistuje X[FID], tak sa do X[FID] uchová hodnota D a pokračuje sa v cykle.

Spiatočné oneskorenie

Spiatočné oneskorenie (round trip-time, RTT) patrí medzi časové charakteristiky, ktoré je možné merať aj jedným meracím bodom.

Spiatočné oneskorenie je čas potrebný k odoslaniu paketu zo zdroja, jeho prijatiu v cieľi, okamžitému odoslaniu naspäť k zdroju a jeho prijatiu v zdroji.[1]

Čas uzavretej slučky je čas, ktorý uplynie medzi vyslaním paketu s požiadavkou v jednom mieste a prijatím príslušného paketu s odpoveďou (napr. TCP-SYN/SYN-ACK). Tento spôsob sa ale nepoužíva na odhad jednosmerného oneskorenia, pretože nie je možné zabezpečiť rovnakú cestu pre požiadavku aj odpoveď. Cesty môžu mať rôzne charakteristiky a slučka môže byť asymetrická.

7. Zhodnotenie dosiahnutých výsledkov riešenia diplomovej práce

Cieľom diplomovej práce bolo vytvoriť animačný model, ktorý by podrobne zobrazoval celú koncepciu architektúry meracieho nástroja a meraní QoS parametrov. Projekt bol realizovaný v prostredí Macromedia Flash MX.

Animačný model bolo možné realizovať aj iným spôsobom ako použitím flash aplikácie, ale práve flash aplikácie poskytujú zaujímavý spôsob ako tvoriť pútavé animácie takmer jednoduchým spôsobom.

Výhodou riešenia je jeho použiteľnosť na prezentačné účely. V súčasnosti je totiž veľmi dôležitým znakom upútať a zaujať, čo predpokladám sa mi vytvorením tohto animačného modelu aj podarilo. Animačný model je praktickým výstupom celej teoretickej časti, princípov, metód ako aj meraní parametrov QoS, na ktorých je založený merací nástroj. Z tohto hľadiska bude teda tento projekt možné použiť aj na vzdelávacie účely.

V prípade ďalšieho rozšírenia koncepcie meracieho nástroja BasicMetra a realizovaní ďalších meraní QoS parametrov je možné nové poznatky doplniť do súčasnej podoby animačného modelu. Vhodné by bolo tiež doplniť animácie o vykresľovanie grafov.

V budúcnosti by mohlo byť zaujímavé prepojiť animáciu s „reálnym funkčným BasicMetrom“ tak, aby bolo možné prezentovať výstupy meraní v rámci flash animácie.

8. Zoznam používaných skratiek

ACK	Acknowledgement	Potvrdenie
AS	Autonomous System	Autonómny systém
CGI	Common Gateway Interface	Všeobecné rozhranie brány
CPU	Central Processing Unit	Základná jednotka / procesor
DSCP	DiffServ CodePoint	DiffServ riadiaci bod
FID	Flow Identifier	Identifikačné číslo toku
FIN	Finish	Ukončenie spojenia
FTP	File Transfer Protocol	Protokol prenosu súborov
GIF	Graphics Interchange Format	Grafický formát s využitím kompresie LZW (Lempel-Ziv-Welch)
HTML	Hypertext Markup Language	Hypertextový značkovací jazyk
ICMP	Internet Control Message Protocol	Protokol riadiacich správ internetu
ID	Identifier	Identifikačné číslo
IETF	Internet Engineering Task Force	Technická štandardizačná skupina pre internet
IP	Internet Protocol	Medzisieťový protokol
IPFIX	Internet Protocol Flow Information eXport	Export informácií o IP tokoch
JPEG	Joint Photographic Experts Group	Norma na ukladanie jednotlivého statického rastrového obrazu
JRE	Java Runtime Enviroment	Vývojové prostredie Java
LAN	Local-Area Network	Lokálna sieť
MPLS	MultiProtocol Label Switching	Protokol pre prepínanie IP paketov na základe pridelených značiek
MRTG	The Multi Router Traffic Grapher	Zapisovač viacsmerovej prevádzky
NNTP	Network News Transport Protocol	Protokol prenosu sieťových správ
PID	Packet Identifier	Identifikačné číslo paketu

PNG	Portable Network Graphics	Prenos obrazu po sieti
POP3	Post Office Protocol version 3	Protokol pre prístup užívateľov k e-mail správam uloženým na mail serveri
PSTN	Public Switched Telephone Network	Verejná prepínaná telefónna sieť
QoS	Quality of Service	Kvalita služby
RRA	Round Robin Archive	Archív Round Robin
RRD	Round Robin Database	Databáza Round Robin
SCTP	Stream Control Transmission Protocol	Protokol určený pre transport PSTN signalizačných správ cez IP sieť
SMS	Short Message Service	Mechanizmus doručovania krátkych textových správ v mobilných sieťach
SMTP	Simple Mail Transfer Protocol	Jednoduchý protokol prenosu pošty
SNMP	Simple Network Management Protocol	Jednoduchý protokol manažmentu siete
SQL	Structured Query Language	Štruktúrovaný dotazovací jazyk
SRBD		Systém riadenia bázy dát
TCP	Transmission Control Protocol	Protokol na riadenie prenosu
TCP/IP	Transmission Control Protocol / Internet Protocol	Protokol na riadenie prenosu / medzisieťový protokol
TLV	Treshold Limit Value	Hraničná limitná hodnota
ToS	Type of Service	Typ služieb
TS	TimeStamp	Časová značka
UDP	User Datagram Protocol	Používateľský datagramový protokol
UTC	Coordinated Universal Time	Koordinovaný univerzálny čas
W3C	World Wide Web Consortium	Medzinárodné združenie vyvíjajúce štandardy pre web
XML	Extensible Markup Language	Otvorený W3C štandard pre popis dátových štruktúr

9. Zoznam použitej literatúry

- [1] ALMES, G. – KALIDINDI, S. – ZEKAUSKAS, M.: A Round-trip Delay Metric for IPPM. [online]. September 1999. RFC 2681. Dostupné na internete: <<http://www.faqs.org/rfcs/rfc2681.html>>.

- [2] ANDRÉ, Marián: Meranie a vyhodnocovanie prevádzkových parametrov v počítačových sieťach. Košice: Technická univerzita. Fakulta Elektrotechniky a informatiky. Katedra počítačov a informatiky. 2003. 73 strán. Vedúci diplomovej práce: Ing. František Jakab.

- [3] BARBER, S.: Monitoring Your Network with Freely Available Statistics Reporting Tools. [online]. 7. február 1998. Dostupné na internete: <<http://www.academ.com/nanog/feb1998/nettools/tsld001.htm>>.

- [4] BERRY, I.: Cacti the complete rrdtool-based graphing solution. [online]. 2004. Dostupné na internete: <<http://www.raxnet.net/products/cacti/>>.

- [5] Carnegie Mellon University. Identifying tools that aid in detecting signs of intrusion. [online]. Január 2001. Dostupné na internete: <<http://www.cert.org/security-improvement/implementations/i042.07.html>>.

- [6] Cisco Systems. Cisco IOS NetFlow Version 9 Flow-Record Format. [online]. Dostupné na internete: <http://www.cisco.com/en/US/tech/tk648/tk362/technologies_white_paper09186a00800a3db9.shtml>.

- [7] Cisco Systems. NetFlow Overvie. [online]. Február 2003. Dostupné na internete: <http://bm.cnl.tuke.sk/data/netflow/nflov_pg-new.pdf>.

- [8] DEMICHELIS, P.: IP Packet Delay Variation Metric for IP Performance Metrics. [online]. November 2002. RFC 3393. Dostupné na internete: <<http://www.faqs.org/rfcs/rfc3393.html>>.
- [9] FRANKLIN, D: Macromedia Flash MX. Kompletní průvodce. 1. vyd. Brno: Computer Press. 2003. 846 strán. ISBN 80-7226-831-7.
- [10] GALSTAD, E.: Nagios. [online]. Máj 2003. Dostupné na internete: <www.nagios.org>.
- [11] HEDENFALK, M. [online]. Dostupné na internete: <<http://www.nongnu.org/confuse/>>.
- [12] JAKAB, F.: Tvorba sieťových prostredí pre televzdelávanie (Metódy optimalizácie hodnotenia a riadenia v počítačových sieťach – meranie a vyhodnocovanie prevádzkových parametrov v počítačových sieťach). Dizertačná práca. Košice. 2004.
- [13] Jelsoft Enterprises Ltd. [online]. Marec 2003. Dostupné na internete: <<http://forums.burst.net/archive/index.php/t-260.html>>.
- [14] KOHLER, P. – CLAISE, B.: IPFIX fine-tunes traffic analysis. [online]. 11. august 2003. Dostupné na internete: <<http://www.nwfusion.com/news/tech/2003/0811techupdate.html>>.
- [15] LEINEN, S.: Evaluation of Candidate Protocols for IP Flow Information Export (IPFIX). [online]. Október 2004. RFC 3955. Dostupné na internete: <<http://mailman.rfc-editor.org/pipermail/rfc-dist/2004-October/000680.html>>.
- [16] MONITOR TOOL.com. [online]. Dostupné na internete: <www.monitortools.com>.

- [17] NEJMAN, J.: NetFlow Monitor – Overview. [online]. 2003. Dostupné na internete: <http://netflow.cesnet.cz/n_netflow.php>.
- [18] Netways. Nagios Monitoring. [online]. August 2003. Dostupné na internete <<http://www.netways.de/Nagios.1097.0.html>>.
- [19] Network Monitoring Tools. [online]. Dostupné na internete: <http://staff.science.uva.nl/~jblom/datatag/wp3_1/tools/monitor_tools.html>.
- [20] NORSETH, K.: Architecture Model for IP Flow Information Export. Draft-ietf-ipfix-architecture-02.txt. [online]. December 2002. <<http://norseth.org/ietf/ipfix/draft-ietf-ipfix-architecture-02.txt>>.
- [21] OETIKER, T.: About RRD Tool. [online]. Október 2003. Dostupné na internete: <<http://people.ee.ethz.ch/~oetiker/webtools/rrdtool/index.html>>.
- [22] OETIKER, T.: About SmokePing. [online]. Marec 2003. Dostupné na internete: <<http://people.ee.ethz.ch/~oetiker/webtools/smokeping/>>.
- [23] OETIKER, T.: MRTG. [online]. Júl 2002. Dostupné na internete: <<http://people.ee.ethz.ch/~oetiker/webtools/mrtg>>.
- [24] OSDN Open Source Development Network. Usage Statistics. [online]. 2004. Dostupné na internete: <http://sourceforge.net/project/stats/?group_id=34886>.
- [25] PANKAJ, G. – MCKEOWN, N.: Packet Classification on Multiple Fields. Proc. Sigcomm. Computer Communication Review. vol. 29, no. 4, pp 147-60. September 1999. Harvard University.
- [26] PAXSON, V. – ALMES, G. – MAHDAVI, J. – MATHIS, M.: Framework for IP Performance Metrics. [online]. Máj 1998. RFC 2330. Dostupné na internete: <<http://www.faqs.org/rfcs/rfc2330.html>>.

- [27] PUŽMANOVÁ, Rita. Moderní komunikační síte. 1. vyd. Praha: Computer Press, 1998. 446 strán. ISBN 80-7226-098-7.
- [28] QUITTEK, J. – ZSEBY, T. – CLAISE, B. – ZANDER, S.: Requirements for IP Flow Information Export (IPFIX). [online]. Október 2004. RFC 3917. Dostupné na internete: <<http://www.faqs.org/rfcs/rfc3917.html>>.
- [29] RIVEST, R.: The MD5 Message-Digest Algorithm. [online]. Apríl 1992. RFC 1321. Dostupné na internete: <<http://www.faqs.org/rfcs/rfc1321.html>>.
- [30] SADASIVAN, G. et al.: Architecture for IP Flow Information Export. Draft-ietf-ipfix-architecture-07. [online]. Marec 2005. Dostupné na internete: <<http://www.ietf.org/internet-drafts/draft-ietf-ipfix-architecture-07.txt>>.
- [31] SUČÍK, Juraj: Príspevok k problematike merania a vyhodnocovania parametrov kvality služieb (QoS) v počítačových sieťach. Košice: Technická univerzita. Fakulta Elektrotechniky a informatiky. Katedra počítačov a informatiky. 2003. 66 strán. Vedúci diplomovej práce: Ing. František Jakab.
- [32] SWEENY, R.: Monitoring Yours Enterprise PACS With Nagios, Cacti and Smokeping. [online]. Marec 2004. Dostupné na internete: <<http://people.ee.ethz.ch/~oetiker/webtools/smokeping/pub/contrib/EnterprisePACSMonitoringwithNagiosSmokepingandCacti.pdf>>.
- [33] WALTNER, Ch.: Cisco's NetFlow Selected as Basis for IETF Standard. IP data flow export technology cornerstone to network management. [online]. 2005. Dostupné na internete: <http://newsroom.cisco.com/dlls/innovators/software_standards/idw_052003.html>.
- [34] ZAJAC, B.: Realtime monitoring systems for your computer and networking equipment. [online]. Júl 1999. Dostupné na internete: <<http://www-cgi.cnn.com/TECH/computing/9907/19/monitoring.idg/>>.

10. Zoznam príloh

1. CD médium

- diplomová práca v elektronickej podobe
- prílohy v elektronickej podobe
- flash aplikácia – Meranie a vyhodnocovanie parametrov QoS

2. Používateľská príručka

3. Systémová príručka

11. Zoznam obrázkov a tabuliek

Zoznam obrázkov

Obr. 3.1: MRTG graf.....	8
Obr. 3.2: Graf SmokePing	12
Obr. 3.3: Graf Cacti.....	15
Obr. 3.4: Štatistické zobrazenie.	15
Obr. 4.1: Architektúra pasívnych meraní.	20
Obr. 4.2: Architektúra IPFIX.....	22
Obr. 4.3: Formát IPFIX správ, 1. príklad.	27
Obr. 4.4: Formát IPFIX správ, 2.príklad.	27
Obr. 4.5: Formát IPFIX správ, 3.príklad.	27
Obr. 5.1: Architektúra meracej platformy.	32
Obr. 5.2: Architektúra meracieho nástroja.	33
Obr. 5.3: Bloková schéma spracovania paketu v diagrame.....	38
Obr. 6.1: Model meracieho nástroja.....	41
Obr. 6.2: Príklad šablóny FlowSet.	46
Obr. 6.3: Príklad FlowSetu dát.	49
Obr. 6.4: Algoritmus výpočtu jednosmerného oneskorenia.	52
Obr. 6.5: Algoritmus výpočtu kolísania oneskorenia.....	53

Zoznam tabuliek

Tab. 4.1: Obsah typu sád.	26
Tab. 4.2: Formát toku NetFlow v5.....	30
Tab. 6.1: Popis formátu hlavičky paketu protokolu NetFlow v9.....	46
Tab. 6.2: Popis šablóny FlowSetu protokolu NetFlow v9.	47
Tab. 6.3: Popis poľa Field Type protokolu NetFlow v9.	48
Tab. 6.4: Popis FlowSetu dát protokolu NetFlow v9.....	50