

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

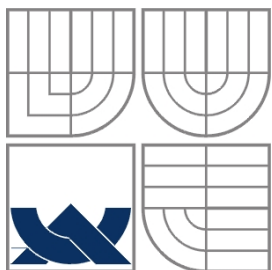
IDENTIFIKACE POČÍTAČE NA ZÁKLADĚ ČASOVÝCH
ZNAČEK PAKETŮ

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

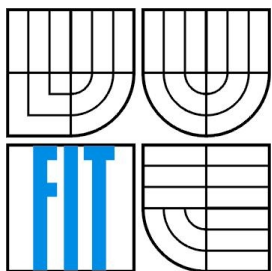
AUTOR PRÁCE
AUTHOR

BC. MARTIN KRBA

BRNO 2012



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
ÚSTAV POČÍTAČOVÝCH SYSTÉMŮ

FACULTY OF INFORMATION TECHNOLOGY
DEPARTMENT OF COMPUTER SYSTEMS

IDENTIFIKACE POČÍTAČE NA ZÁKLADĚ ČASOVÝCH ZNAČEK PAKETŮ

COMPUTER IDENTIFICATION BASED ON PACKET'S TIMESTAMPS

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

BC. MARTIN KRBA

VEDOUCÍ PRÁCE
SUPERVISOR

ING. JAN KAŠTIL

BRNO 2012

Abstrakt

Primárním způsobem identifikace zařízení v počítačové síti je využití jejich MAC adresy a IP adresy. Cílem této práce je vytvoření aplikace, která dokáže jednoznačně identifikovat počítač v síti Internet, a to i v případě, kdy dojde ke změně jeho IP adresy nebo MAC adresy. Základem je využití drobných časových odchylek v hardvérovém zařízení, tzv. clock skew. Ty se vyskytují v každých hodinách, které jsou založeny na krystalovém oscilátoru. Jejich využití je zvláště výhodné, protože není potřebná žádná úprava sledovaných zařízení a při určité implementaci ani jejich vědomá spolupráce. Zjištění těchto hodnot spočívá v zachycení dostatečného množství paketů nesoucích časové známky, tzv. timestamps. Uplatnění jednoznačné identifikace zařízení je velice široké, příkladem může být vyšetřování počítačové kriminality, sledování zařízení využívající různé přístupové body, zjištění počtu zařízení za routerem s překladem síťových adres (NAT).

Abstract

Basic way how to identify a device in computer network is by MAC address and IP address. Main goal of this work is to create an application capable of clear identification of devices in computer network regardless change of their MAC address or IP address. This is done by exploiting tiny deviations in hardware clock known as clock skew. They appear in every clock based on quartz oscillator. Using clock skew is beneficial, because there is no need of any changes in fingerprinted device nor their cooperation. Accessing these values is done by capturing packets with timestamps included. Application of this method is very wide, for example computer forensics, tracking the device using different access points or counting devices behind router with NAT.

Klíčová slova

časová odchylka, časová známka, identifikace, TCP Timestamp Option, metoda nejmenších čtverců, lineární regrese

Keywords

clock skew, timestamp, identification, TCP Timestamp Option, least squares, linear regression

Citace

Bc. Martin Krba: Identifikace počítače na základě časových značek paketů, diplomová práce, Brno, FIT VUT v Brně, 2012

Identifikace počítače na základě časových značek paketů

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracoval samostatně pod vedením Ing. Jana Kaštila. Uvedl jsem všechny literární prameny a publikace, ze kterých jsem čerpal.

.....
Bc. Martin Krba
21.5.2012

Poděkování

Chcel by som sa poďakovať pánovi Ing. Janovi Kaštilovi za vedenie pri tejto diplomovej práci, usmernenie a množstvo cenných rád.

© Bc. Martin Krba, 2012

Tato práce vznikla jako školní dílo na Vysokém učení technickém v Brně, Fakultě informačních technologií. Práce je chráněna autorským zákonem a její užití bez udělení oprávnění autorem je nezákonné, s výjimkou zákonem definovaných případů.

Obsah

Obsah.....	1
1 Úvod.....	3
2 Meranie času v moderných počítačových sieťach.....	5
2.1 Real-time clock.....	6
2.2 Časová odchýlka.....	6
2.3 Systémy merania času.....	7
2.4 Časová známka.....	8
2.5 Synchronizácia času.....	9
3 TCP/IP.....	12
3.1 ICMP.....	12
3.1.1 ICMP Timestamp request a reply.....	13
3.2 TCP.....	14
3.2.1 TCP Timestamp Option.....	15
3.2.1.1 RTTM.....	16
3.2.1.2 PAWS.....	17
3.2.1.3 Timestamp clock.....	17
4 Identifikácia zariadení v rámci počítačových sietí.....	19
4.1 Techniky identifikácie z hľadiska spolupráce označovaného zariadenia.....	19
4.2 Súčasné metódy identifikácie.....	20
4.3 Vzdialené označenie fyzických zariadení.....	21
4.3.1 Princíp metódy.....	21
4.3.2 Možnosti využitia.....	22
4.3.3 Ochrana pred identifikáciou.....	23
5 Lineárna regresia.....	24
5.1.1 Metóda najmenších štvorcov.....	25
5.1.2 Ortogonálna regresia.....	27
5.1.3 Rekurzívna metóda najmenších štvorcov.....	29
5.1.4 Metóda aktualizácie odhadu časovej odchýlky pri príchode paketu váženým priemerom.....	32
6 Implementácia.....	34
6.1 Zachytávanie paketov.....	34
6.2 Analyzovanie hlavičiek a získanie časových známok.....	34
6.3 Výpočty časovej odchýlky.....	35
6.4 Aktualizácia hodnoty časovej odchýlky.....	35
7 Testovanie a dosiahnuté výsledky.....	36

7.1 Porovnanie regresných metód.....	36
7.2 Rozlíšenie zariadení.....	37
7.3 Vplyv vynechávania paketov.....	38
7.4 Priebežná aktualizácia časovej odchýlky.....	43
8 Záver.....	45
8.1 Pokračovanie práce.....	45
Literatúra.....	46
Zoznam príloh.....	48

1 Úvod

Primárnym spôsobom identifikácie zariadení v počítačovej sieti je využitie ich MAC adresy a IP adresy. MAC adresa je priradená sieťovej karte priamo pri jej výrobe a je jedinečným identifikátorom sieťového zariadenia. IP adresa jednoznačne identifikuje zariadenie v počítačovej sieti, ktoré komunikuje prostredníctvom protokolu IP. Ich zmena môže nastať napríklad pri vypršaní platnosti IP adresy a následnom priradení inej IP adresy DHCP serverom, zmene miesta pripojenia, využitím inej sieťovej karty alebo manuálnym prepísaním MAC adresy.

Cieľom tejto práce je vytvorenie aplikácie, ktorá dokáže jednoznačne identifikovať počítač v sieti Internet, a to aj v prípade, keď dôjde k zmene jeho IP adresy alebo MAC adresy. Základom identifikácie je využitie drobných časových odchýlok v hardvéri zariadení, tzv. clock skew. Tie sa vyskytujú v každých hodinách, ktoré sú založené na kryštálovom oscilátore. Ich využitie je obzvlášť výhodné, pretože nie je potrebná žiadna úprava sledovaných zariadení a pri určitej implementácii ani ich vedomá spolupráca. Zistenie týchto hodnôt spočíva v zachytení dostatočného množstva paketov nesúcich časové známky, tzv. timestamps. Tie sú zahrnuté len v paketoch niektorých protokolov a to len za určitých okolností.

Uplatnenie jednoznačnej identifikácie zariadenia je veľmi široké, príkladom môže byť vyšetrovanie počítačovej kriminality, sledovanie zariadenia, ktoré využíva rôzne prístupové body, zistenie počtu zariadení za routerom s prekladom sieťových adries (NAT), a to bez ohľadu na použitie statickej alebo dynamickej IP adresy a skúmanie bloku IP adries pre určenie, či sú využívané virtuálnymi strojmi v rámci honeynetu. Táto diplomová práca je založená na publikácii z [12], ktorá obsahuje špecifikáciu tohto postupu.

Ďalším cieľom, ktorý si táto práca stanovuje dosiahnuť, je vykonanie testov na implementovanej aplikácii, a to za účelom preskúmania jej možností z pohľadu budúceho nasadenia do skutočnej prevádzky, napríklad v routeri alebo inom sieťovom uzle. Vzhľadom na zvyšujúcu sa rýchlosť prenosov a objemu prenesených dát v súčasných sieťach vzrastá počet paketov, ktoré musí takéto zariadenie s obmedzenými zdrojmi spracovať. Návrh testov sa preto zameriava na zníženie počtu potrebných výpočtov.

V prípade implementácie tejto metódy v rámci bezpečnostného systému by bolo vhodné mať vždy dostupnú hodnotu časovej odchýlky získavanej z aktuálne zachytávaného toku dát. Pre tento prípad bol implementovaný a následne testovaný postup priebežnej aktualizácie časovej odchýlky pri prijatí každého nového paketu.

Druhá kapitola je zameraná na princípy merania času v moderných počítačových sieťach. Je tu uvedený spôsob fungovania vnútorných hodín elektronických zariadení a niektoré ich nedostatky, najmä časové odchýlky. Taktiež sú tu popísané časové známky, najpoužívanejšie reprezentácie času

a dostupné spôsoby synchronizácie zariadení. Tretia kapitola popisuje rodinu protokolov TCP/IP a možné využitie niektorých protokolov z tejto rodiny pre potreby označenia sieťových zariadení. Využívajú sa pritom časové známky, ktoré sú za určitých podmienok obsiahnuté v prenášaných paketoch. Obsahom štvrtej kapitoly je samotný popis metódy identifikácie zariadení na základe zachytených časových známok, jej možného využitia, ale aj ochrany pred touto metódou. Piata kapitola poskytuje priestor popisu matematických metód, ktoré budú pre účely identifikácie použité. Jedná sa o lineárnu regresiu, ktorá sa využíva na získanie regresnej priamky štatistických dát. V nasledovnej šiestej kapitole sa nachádza postup implementácie výslednej aplikácie. Návrh testov a prehľadné vyhodnotenie dosiahnutých výsledkov je obsiahnuté v siedmej kapitole. Záverečná ôsma kapitola sumarizuje dosiahnuté výsledky ako celok, prínos tejto práce a takisto naznačuje možnosti jej pokračovania.

2 Meranie času v moderných počítačových sieťach

V tejto kapitole budú popísané základné spôsoby merania času a pre tento účel využívané zariadenia, reprezentácia času najpoužívanejšími systémami a rôzne synchronizačné techniky, ktoré slúžia na udržanie rovnakého času na rôznych zariadeniach. Cieľom je poukázať na slabiny lokálnych hodín nachádzajúcich sa v počítačoch a ďalších zariadeniach, ktoré sú využité pri metóde vzdialeného označenia takýchto zariadení.

Na úvod sú uvedené základné pojmy tak, ako sú definované v RFC 1305 [18]. Čas udalosti je abstrakcia určujúca poradie ich výskytu v rámci nejakej sústavy. Oscilátor je generátor schopný produkovať presnú frekvenciu s danou toleranciou. Hodiny sú oscilátor spolu s počítadlom, ktoré zaznamenáva počet cyklov od jeho inicializácie na danú hodnotu v danom čase. Hodnota počítadla v akomkoľvek čase sa nazýva epocha času. Epochy sú konečné a závisia na presnosti počítadla. Synchronizácia frekvencie znamená nastaviť oscilátory tak, aby bežali s rovnakou frekvenciou. Synchronizácia času je nastavenie hodín na rovnakú hodnotu v rámci danej epochy s ohľadom na UTC. Synchronizácia hodín je kombináciou synchronizácie času a frekvencie. Rozlíšenie hodín (tik) je najmenšia jednotka, o ktorú môže byť čas zvýšený. Offset hodín je odchýlka od skutočného času. Presný čas má v danom momente nulový offset. Časová odchýlka (clock skew) v danom čase je rozdiel frekvencie hodín a štandardnej frekvencie.

Takmer každé hodiny pozostávajú zo zdroja striedavého periodického signálu, zosilňovacieho obvodu a prídavných obvodov. Ako zdroj striedavého periodického signálu slúži rezonančný obvod. Tým môže byť kryštálový piezoelektrický oscilátor, jednoduchý LC obvod alebo dokonca RC obvod. Zosilňovací obvod invertuje signál oscilátora a jeho časť vracia späť do oscilátora, aby sa zachovala oscilácia. Prídavné obvody, ktorými môžu byť programovateľné i pevné násobiče a deliče kmitočtu, fázové závesy a multiplexory, produkujú výstupný signál s požadovanými parametrami.

Okrem klasických hodín sa využívajú aj presnejšie prístroje, takzvané atómové hodiny. Ich základom je oscilácia molekúl alebo atómov vhodnej látky, ktorej vibrácie sú rýchle a stabilné v čase. Najčastejšie sú to atómy vodíka, rubídia alebo cézia. V súčasnosti sa používa najmä cézium, ktoré je tiež základom definície sekundy v sústave SI. Jedna sekunda je podľa [28] doba trvania 9 192 631 770 periód žiarenia, zodpovedajúca prechodu medzi dvoma hyperjemnými hladinami základného stavu atómu ^{133}Cs . Ich presnosť je taká vysoká, že môže dôjsť k odchýlke atómových hodín o maximálne jednu sekundu za stopäťdesiat rokov. Vysoká cena však nedovoľuje ich použitie v bežných prístrojoch, preto sa v počítačoch používajú takmer výhradne kryštálové oscilátory.

Za dokonalých okolností, kedy by boli hodiny nastavené na úplne presný čas a zároveň by frekvencia ostala nemenná, by hodiny vždy ukazovali presný čas. Avšak dosiahnutie tohto stavu nie

je v praxi možné. Výskyt náhodných a systematických odchýlok je prirodzený pre každý oscilátor. Taktiež aj efekt vonkajších vplyvov zohráva svoju úlohu a ovplyvňuje kvalitu hodín. Tá sa hodnotí štyrmi základnými mierami. Sú to presnosť frekvencie, stabilita frekvencie, presnosť času a stabilita času.

2.1 Real-time clock

Real-time clock (RTC) sú počítačové hodiny, ktoré udržiavajú aktuálny čas. Jadro tohto modulu je tvorené kaskádou počítadiel, pričom prvé (počítadlo sekúnd) počíta hodinový signál s kmitočtom 1 Hz, ktorý je vytváraný sústavou preddeličiek hodinového signálu produkovaného generátorom hodín. Ich výhodou je nízka spotreba energie, uvoľnenie hlavného systému pre náročné výpočty a niekedy aj vyššia presnosť v porovnaní s inými metódami. Súčasťou RTC býva alternatívny zdroj energie umožňujúci nepretržité počítanie času v situáciách, kedy je primárny zdroj energie nedostupný. Presnosť počítania času je závislá predovšetkým na použitom kryštálovom oscilátore generátora hodín. Jeho obvyklá frekvencia je 32 768 Hz (2^{15} cyklov za sekundu), ktorá je vhodná pre jednoduché obvody binárneho počítadla.

2.2 Časová odchýlka

Časová odchýlka, z anglického Clock skew, je neželaný efekt, ktorý postihuje každé hodiny založené na kryštálovom oscilátore. Obvykle sa udáva v bezrozmernej jednotke PPM (Parts Per Million), ktorá udáva počet výskytov na jeden milión pozorovaní. Relatívna časová odchýlka je rozdiel frekvencií dvoch porovnávaných hodín. Spôsobuje, že hodnoty na týchto hodinách sa od seba vzdľaľujú (drift apart). Najčastejšie sa jedná o rozdiely v milisekundách, avšak v niektorých prípadoch môže relatívna časová odchýlka nadobudnúť hodnotu aj niekoľkých minút.

Príčinou vzniku odchýlky frekvencie kmitania kryštálového oscilátora je predovšetkým jeho nedokonalosť, starnutie a taktiež náchylnosť na vonkajšie vplyvy. Tými sú kolísanie dodávanej elektrickej energie, magnetické a elektrické polia, výkyvy teploty, vlhkosti, tlaku a iné faktory okolitého prostredia. Zmena frekvencie má za následok zvyšovanie offsetu hodín. Aj keď sú prejavy tejto odchýlky relatívne malé, v niektorých situáciách môžu spôsobiť isté problémy. Napríklad pri meraní vlastností počítačovej siete, kde sú vyžadované časové známky, môžu imitovať preťaženosť siete zdanlivo dlhším časom potrebným na doručenie paketu. Odstránenie alebo zmiernenie efektov spôsobených časovými odchýlkami bolo jedným z dôvodov zavedenia synchronizačných techník, ako napríklad využívanie protokolu NTP.

Časové odchýlky boli predmetom mnohých štúdií, publikovaných napríklad v [29], [23] a [19], ktoré boli zamerané napríklad na možnosti výpočtu týchto odchýlok a ich následného odstránenia z výsledkov meraní oneskorenia na sieti. Na rozdiel od nich sa táto práca inšpirovaná [12] snaží o ich využitie. Pri skúmaní časových odchýlok sa zistilo, že majú niekoľko špecifických vlastností. Vďaka lineárne sa zväčšujúcemu odklonu od skutočného času bola určená ich stálosť v čase. Taktiež rozloženie časových odchýlok v rôznom hardvéri je dostatočné pre potreby implementovanej metódy. Keďže sú závislé len od použitého hardvéru hodín, vykazujú nezávislosť na operačnom systéme, spôsobe pripojenia k sieti a umiestnení v nej. Vďaka použitiu relatívnej časovej odchýlky je táto metóda nezávislá na označujúcom zariadení. Pre výpočet odchýlok je potrebné relatívne malé množstvo zachytených paketov. Tieto špecifické vlastnosti časových odchýlok umožňujú ich využitie pre rozlíšenie jednotlivých zariadení prítomných v počítačovej sieti.

Časová odchýlka sa nemusí nutne vzťahovať len priamo na časové údaje v zmysle časových známk. Je ju možné pozorovať aj z periodicky sa opakujúcich udalostí. Príkladom môže byť kontrola prítomnosti novej pošty na serveri pomocou poštového klienta každých n minút, či iné aktivity plánované v pravidelných intervaloch. Za pomoci Fourierovej transformácie je možné časovú odchýlku určiť z časov výskytov týchto udalostí.

2.3 Systémy merania času

V roku 1925 zaviedla IAU (International Astronomical Union) pojem Svetový čas (Universal Time (UT)). Je to pásmový čas okolo nultého poludníka, ktorý prechádza greenwichskou hviezdárňou vo Veľkej Británii. Tento čas sa inak nazýva aj UT0, alebo aj Greenwich Mean Time (GMT). Postupne sa z neho vyvinuli ďalšie druhy času, medzi ktorými sú väčšinou len minimálne odlišnosti, avšak v aplikáciách vyžadujúcich veľkú presnosť určenia času sú tieto rozdiely významné [8].

Zohľadnením nepravidelných pohybov zemských pólov, známych aj ako Chandlerovo kolísanie zemskej osi (Chandler wobble), vzniká UT1. Toto kolísanie je spôsobené rotáciou neguľovitého telesa, v tomto prípade Zeme, pričom jeho perióda je 433 dní. Podrobný popis tohto javu je obsahom [9]. Úpravou UT1 o ročné variácie v zemskej rotácii získame UT2, ktorý sa však už takmer nepoužíva.

Iným prístupom k meraniu času je použitie atómových hodín, ktoré nezávisia na pohyboch Zeme. Využitím približne dvesto atómových hodín umiestnených v zhruba sedemdesiatich laboratóriách po celom svete sa získava Medzinárodný atómový čas (International Atomic Time (TAI)). Z aktuálneho času jednotlivých atómových hodín je určená presná hodnota ako vážený priemer, pretože medzi nimi môžu byť rozdiely aj niekoľko nanosekúnd. Tento výpočet sa vykonáva

v Bureau International des Poids et Mesures (Medzinárodný úrad pre miery a váhy), kde sú údaje zbierané pomocou GPS.

Nástupcom GMT a zároveň najpoužívanejším časovým štandardom na svete je od roku 1972 systém UTC (Coordinated Universal Time), ktorý je založený na TAI s využitím priestupných sekúnd. Tie sú pridávané ako kompenzácia spomaľovania zemskej rotácie a sú podľa potreby vkladané 30.06., prípadne 31.12.. O ich pridaní rozhoduje International Earth Rotation and Reference Systems Service (IERS). Do dnešného dňa (30.12.2011) bolo pridaných celkom 34 priestupných sekúnd [31]. Najbližšie dôjde k pridaní priestupnej sekundy 30.06.2012. Taktiež vo výpočtových technológiách našiel UTC svoje uplatnenie.

2.4 Časová známka

Časová známka, z anglického Timestamp, je sekvencia znakov, ktorá väčšinou znamená presný dátum a čas, kedy sa odohrala určitá udalosť. Príkladom môže byť čas vytvorenia alebo poslednej modifikácie súboru, dátum a čas zosnímania fotografie, doručenie emailu, vykonanie finančnej transakcie alebo okamih odoslania či príjmu paketu. V niektorých prípadoch nie je časová známka skutočným časom výskytu udalosti, ale okamih jej zaznamenania počítačom. Rozdiel by však mal byť čo najmenší. Niekedy je dokonca časová známka použitá na číslovanie udalostí a vtedy nemusí nutne obsahovať časové údaje. Jej obsah musí spĺňať jedinú podmienku, ktorou je možnosť jednoznačného určenia poradia udalostí reláciou „odohráva sa pred“ (happens before).

Obsahom časových známok je teda takmer vždy dátum a čas, pričom sa využívajú dva hlavné typy formátu. Prvým je formát určený pre ľahkú čitateľnosť ľuďmi. Príkladom môže byť 21:03:47 02.03.2012 UTC alebo Sat Jan 7 13:31:22 2012. Pri tejto reprezentácii môže v určitých prípadoch dôjsť k dezinterpretácii hodnôt, pretože existujú regionálne odlišnosti v zápise dátumov. Napríklad zápis 01/02/03 môže byť interpretovaný niekoľkými spôsobmi ako 1. február 2003, 3. február 2001 alebo 2. marec 2001. Štandardizácia reprezentácie času a dátumu je obsahom ISO 8601, ktorá rieši tieto konfliktné situácie jednotným formátom zápisu. Druhý formát je usporiadaný na jednoduché spracovanie počítačmi, pre ktoré je optimálna práca s dátami vo forme čísel. Potreba reprezentovať čas ako číslo bola vyriešená počítaním sekúnd od počiatku epochy, ktorý sa však v rôznych systémoch a aplikáciách líši. Najpoužívanejšími počiatkami epoch sú:

- 00:00:00 UTC dňa 01.01.1900 využívaný napríklad protokolom NTP
- 00:00:00 UTC dňa 01.01.1970 známy ako Unixový čas
- 00:00:00 UTC dňa 06.01.1980 počiatok epochy GPS Time

Rovnako sa rôzni aj využitý dátový typ, ktorým je čas reprezentovaný. Najčastejšie je to 32 bitová celočíselná hodnota, ktorá udáva len počet sekúnd alebo 64 bitové číslo s desatinnou časťou, ktorá

vyjadruje menšie časové jednotky. V závislosti na rozlíšení hodín sú to milisekundy, mikrosekundy, nanosekundy či dokonca pikosekundy. Pri použití tohto systému je však nutné pripomenúť, že epocha nie je nekonečný interval a jej dĺžka závisí na použitom dátovom type. Po jej uplynutí dôjde k pretečeniu počítadla a reštartu počítania, čo spôsobí, že jedno číslo bude reprezentovať dva rôzne dátumy. Toto je neželaný efekt, preto sa zavádza číslovanie epoch.

2.5 Synchronizácia času

Aj napriek tomu, že použité kryštálové oscilátory sú relatívne presné, postupom času sa prejaví časová odchýlka a systém nedisponuje správnym časom. Táto skutočnosť by sama osebe neprekážala, avšak potreba zhodného času po sieti komunikujúcich zariadení je v mnohých aplikáciách nevyhnutná. Preto bolo vyvinutých niekoľko postupov, ktoré zabezpečujú synchronizovanie zariadení.

Network Time Protocol (NTP) je jedným z najstarších, ešte stále používaných, sieťových protokolov (aktuálna verzia NTPv4). Vďaka svojej presnosti (rozlíšenie približne 200 pikosekúnd) je najpoužívanejším protokolom pre určenie času na Internete. Nasledovný popis protokolu NTP pochádza z RFC 5905 [15] a [16]. NTP tvorí hierarchický vrstvený systém, ktorého jednotlivé vrstvy sa nazývajú stratum a majú priradené číslo predstavujúce úroveň. Najvyššia vrstva má číslo 0 a patria tu zariadenia ako atómové hodiny, GPS alebo rádiové hodiny. Vo vrstve stratum 1 sa nachádzajú počítače priamo napojené na zariadenia zo stratum 0. Ďalšie vrstvy sú vždy synchronizované s využitím serverov z predchádzajúcej úrovne. Pritom platí, že presnosť sa postupne so zvyšujúcim sa číslom vrstvy znižuje. NTP servery nastavujú časové známky, ktoré sú uložené v 64 bitoch, kde prvých 32 bitov určuje počet sekúnd od 00:00:00 01.01.1900 a zvyšných 32 bitov obsahuje desatinnú časť a umožňuje teoretické rozlíšenie až 2^{-32} . K pretečeniu škály dochádza každých 2^{32} sekúnd, teda 136 rokov. Budúca verzia NTP by už mala využívať časové známky uložené v 128 bitoch. Pri synchronizácii sa najskôr musí vypočítať RTT

$$RTT = (TS_{Reference} - TS_{Originate}) - (TS_{Transmit} - TS_{Receive}) \quad (1)$$

a následne offset

$$offset = \frac{(TS_{Receive} - TS_{Originate}) + (TS_{Transmit} - TS_{Reference})}{2} \quad (2)$$

kde $TS_{Reference}$ je čas, kedy bol systémový čas naposledy synchronizovaný alebo nastavený. $TS_{Originate}$ je čas odoslania žiadosti klientom, $TS_{Receive}$ je čas prijatia žiadosti serverom a $TS_{Transmit}$ je čas odoslania odpovede serverom. Zohľadnením týchto hodnôt je možné určiť presný čas, avšak len v prípade, kedy je oneskorenie komunikácie symetrické na trasách klient-server a server-klient. V opačnom prípade môže dôjsť k systematickej odchýlke rovnej polovici rozdielu týchto oneskorení. Na dosiahnutie

najlepšieho výkonu pri synchronizácii je nutné, aby bolo jadro operačného systému schopné riadiť čas fázovým závesom, namiesto dosadzovania času do systémových hodín procesom NTPD (NTP daemon).

Zjednodušená forma NTP má názov SNTP (Simple NTP) [17]. Neuvažuje oneskorenie paketov na sieti, ani si nepamätá stav predchádzajúcej komunikácie. SNTP démon využíva jedinú časovú známku, ktorú získa jednou žiadosťou od NTP serveru. Je používaný v prípadoch, keď nie je vyžadovaná veľká presnosť. Klient založený na SNTP nemôže slúžiť ako časový server.

Aj keď je systém Global Positioning System (GPS) zameraný prevažne na určovanie polohy objektov na Zemi a nad Zemou, používa sa aj na šírenie presného času, časových intervalov a frekvencie. Kým navigačné prijímače využívajú GPST (GPS Time) na pomoc pri výpočte polohy, prijímače času a frekvencie používajú satelitné prenosy GPS na kontrolu časovacích signálov a oscilácie. Presnosť frekvencií sa pohybuje v rádoch 10^{-11} a časová presnosť je približne 14 ns. Okrem presnosti je veľkou výhodou systému GPS jeho neustála dostupnosť z akéhokoľvek miesta na Zemi. Tá je dosiahnutá vďaka rozmiestneniu satelitov, pričom z každého miesta na Zemi sú vždy viditeľné minimálne štyri satelity. Družice systému GPS obsahujú atómové hodiny s rubídiovým alebo céziovým oscilátorom. Aj napriek vysokej stabilite ich oscilácie je kvôli relativistickým vplyvom potrebné synchronizovať čas medzi družicami. GPST je riadený zo zariadenia Master Control, ktoré ho udržuje synchronizované s UTC s maximálnou odchýlkou jednej mikrosekundy.

Automated Computer Time Service (ACTS) je od roku 1988 využívaný na synchronizáciu systémového času zariadení, ktoré využívajú analógový modem. Okrem modemu je potrebný už len jednoduchý softvér. Čas je týmto protokolom možné synchronizovať s presnosťou niekoľkých milisekúnd od UTC.

Zámerom zavedenia protokolu Daytime boli testovacie a meracie účely v počítačových sieťach. RFC 867 [24] definuje službu využívajúcu TCP alebo UDP na porte 13.

Služba Time môže podľa RFC 868 [25] bežať na TCP alebo UDP porte 37. Presnosť je kvôli rozlíšeniu jednej sekundy nízka, avšak pre niektoré aplikácie dostatočná. Výhodou tohto protokolu je jeho jednoduchosť.

Precision Time Protocol (PTP) je ďalším protokolom určeným pre synchronizáciu času v počítačovej sieti. Bol vytvorený pre oblasti, kde je potrebná vyššia presnosť, akú ponúka NTP a kde je finančne neprípustné využitie GPS prijímačov v každom uzle siete. Požaduje len minimálne nároky na výkon počítača a šírku pásma, vďaka čomu môže byť použitý v jednoduchých a lacných zariadeniach. Taktiež nevyžaduje veľkú réžiu. V rámci LAN dosahuje vysokú presnosť 1 ns, vďaka čomu je vhodný pre meracie a kontrolné systémy v rámci jednej podsiete. PTPv2 je nová verzia pôvodného protokolu [30], ktorá zavádza niekoľko vylepšení. Presnosť sa zlepšila na hodnotu 2^{-16} ns

a zvýšila sa aj rýchlosť synchronizácie (možná implementácia v jednej fáze). Využíva rovnakú epochu ako Unix time a je založený na TAI. Šírenie správ prebieha pomocou multicastu.

Väčšina počítačov, ktoré sú profesionálne spravované, si aktualizuje svoj systémový čas na presný čas pomocou NTP, prípadne pomocou SNTP v prípadoch, kedy nie je požadovaná najvyššia možná presnosť. Avšak bežné počítače používané laikmi ostávajú zvyčajne nastavené na východzie nastavenia. V mnohých prípadoch nie je synchronizácia nastavená vôbec, v lepšom prípade sa synchronizácia vykonáva občas, napríklad pri štarte systému alebo raz za určitú dobu (týždeň). Systémový čas takýchto zariadení môže byť teda zdrojom časovej odchýlky.

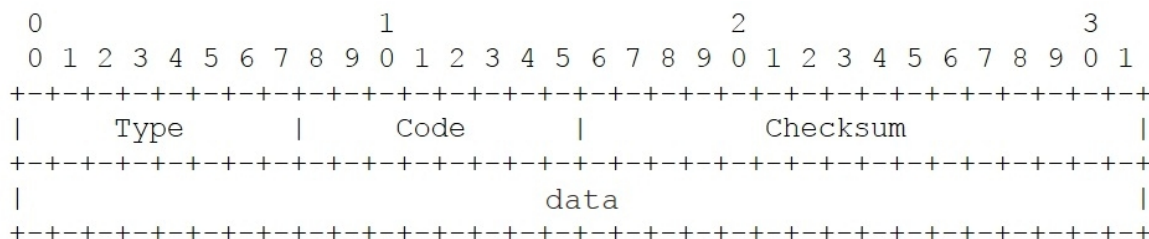
3 TCP/IP

Transmission Control Protocol/Internet Protocol (TCP/IP) je rodina protokolov, ktorá je jednou z aplikácií sedemvrstvovej štruktúry modelu ISO/OSI. Jej názov je tvorený podľa vedúcich protokolov transportnej a sieťovej vrstvy. Internet je dátovou sieťou, ktorá využíva práve túto rodinu protokolov. Patria tu napríklad protokoly IP, TCP, UDP, ICMP, IGMP alebo ARP.

Pre potreby tejto práce sú zaujímavé tie protokoly, ktoré v sebe uchovávajú buď priamo informácie o aktuálnom systémovej čase označovaného zariadenia, alebo údaje, ktoré sú od tejto informácie odvodené, teda časové známky. Protokoly, ktorých pakety obsahujú časovú známku a teda môžu byť využité pri implementovanej metóde, budú priblížené v nasledovných podkapitolách.

3.1 ICMP

Internet Control Management Protocol (ICMP) je súčasťou IP protokolu. Slúži na signalizáciu výnimočných udalostí v sieťach postavených na IP protokole. Protokol ICMP balí svoje dátové pakety do IP protokolu. IP hlavička je v týchto paketoch nasledovaná ICMP hlavičkou. ICMP správy sú zvyčajne generované ako reakcia na chyby v IP datagramoch, prípadne na smerovacie alebo diagnostické účely. Príkladom využitia tohto protokolu sú napríklad programy ping alebo traceroute (tracert). Táto podkapitola vychádza z RFC 792 [26].



Obrázok 3.1: ICMP hlavička (Zdroj: RFC 792 [26])

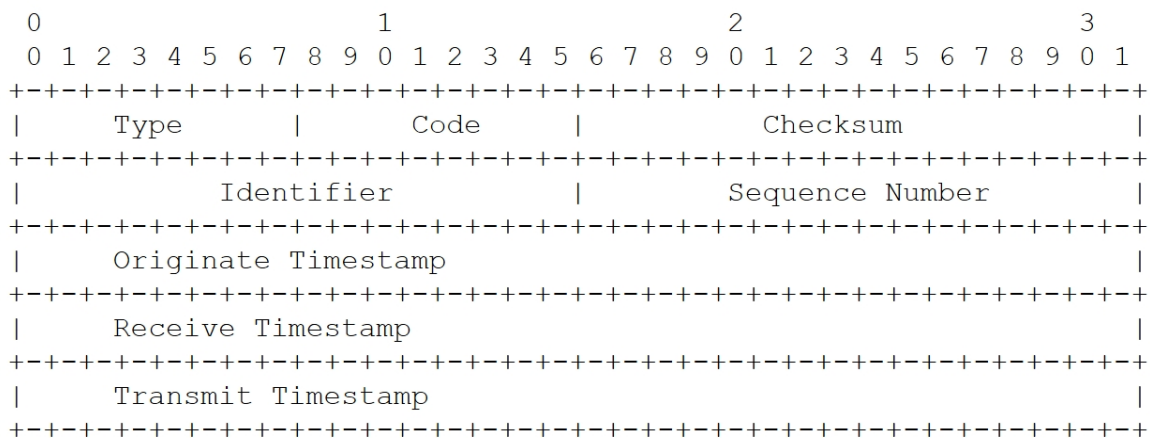
Typy najpoužívanejších ICMP správ (v zátvorke je uvedené číslo používané v položke Type):

- Echo Reply (0) – odpoveď na požiadavku
- Destination Unreachable (3) – informácia o nedostupnosti cieľa
- Source Quench (4) – žiadosť o zníženie počtu zasielaných správ
- Redirect (5) - presmerovanie
- Echo (8) – požiadavka na odpoveď
- Time Exceeded (11) – vypršal časový limit
- Parameter Problem (12) – nesprávna IP hlavička
- Timestamp Request (13) – žiadosť o časovú známku
- Timestamp Reply (14) – odpoveď na žiadosť o časovú známku

- Information Request (15) – žiadosť o informáciu
- Information Reply (16) - odpoveď na žiadosť o informáciu

3.1.1 ICMP Timestamp request a reply

Protokol ICMP umožňuje časovú synchronizáciu pomocou mechanizmu využívajúceho dvojicu správ Požiadavka na časovú synchronizáciu (Timestamp Request) a Odpoveď na požiadavku na časovú synchronizáciu (Timestamp Reply). Prvú odošle zdrojový počítač s vyplnenou položkou čas odoslania žiadosti (Originate Timestamp), ktorá je časom poslednej manipulácie so správou pred jej odoslaním. Cieľový počítač vyplní v odpovedi dve položky. Sú to čas okamihu prijatia žiadosti (Receive Timestamp) a čas odoslania odpovede (Transmit Timestamp), teda čas poslednej manipulácie so správou pred jej odoslaním. Zdrojový počítač si pri prijatí odpovede zaznamená čas a z časových známkov určí Round Trip Time (RTT).



Obrázok 3.2: ICMP Timestamp request and reply (Zdroj: RFC 792 [26])

Hodnoty jednotlivých polí sú nasledovné:

- Type má hodnotu 13 pre timestamp request a 14 pre timestamp reply
- Code je 0
- Checksum (kontrolný súčet) je 16-bitový jednotkový doplnok jednotkového doplnku súčtu ICMP správy, začínajúc od ICMP Type
- Identifier a Sequence Number slúžia ako pomôcka na priradenie žiadosti a zodpovedajúcej odpovede

Časové známky sú 32-bitové čísla reprezentujúce čas vo formáte UTC. V prípade, že nie je dostupný čas v milisekundách, alebo nemôže byť uvedený vo formáte UTC, je ako časová známka vložený akýkoľvek dostupný čas, pričom je táto neštandardná skutočnosť indikovaná najvyšším bitom. Získané informácie sa dajú využiť aj na určenie časovej odchýlky hodín zariadení. Podmienkou je, aby bol identifikátor schopný odosielať ICMP Timestamp Request a identifikované

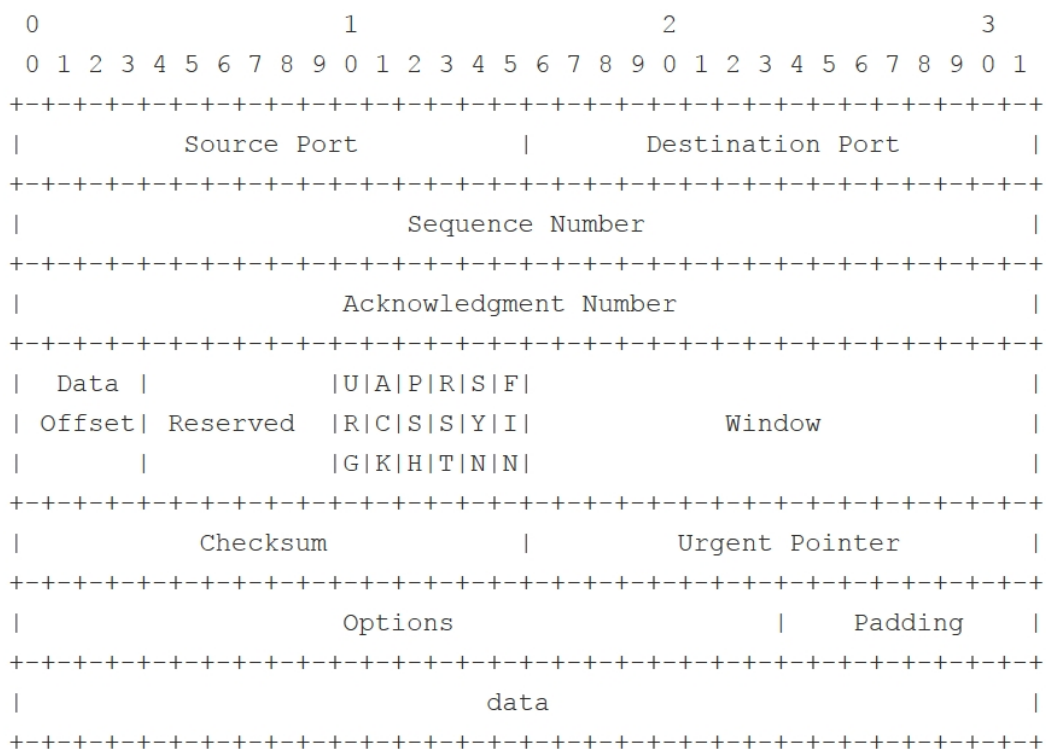
zariadenie odosielať ICMP Timestamp Reply. Limitom využitia ICMP je aj fakt, že mnohé firewally blokujú tento protokol. Taktiež sa zariadenie nesmie nachádzať za NATom.

3.2 TCP

Transmission Control Protocol (TCP) patrí medzi základné protokoly celého Internetu. Konkrétne sa využíva v transportnej vrstve a jeho špecifikácia je obsahom dokumentu RFC 793 [4], ktorý slúžil ako zdroj informácií využitých v tejto podkapitole. Bol navrhnutý tak, aby spoľahlivo pracoval po takmer každom prenosovom médiu a to bez ohľadu na prenosovú rýchlosť, oneskorenie, poškodenie dát, či duplikáciu a doručenie v zlom poradí.

Protokol TCP je spojovanou službou, ktorá ustanoví plne duplexné spojenie medzi dvoma koncovými bodmi (aplikáciami), pričom každý je definovaný špeciálnym číslom - portom. Nadobúda hodnoty 0 - 65535, pričom tento rozsah je rozdelený na tri skupiny. Známe porty využívajú čísla portov 0 - 1023, registrované porty zaberajú interval 1024 - 49151 a pre súkromné porty sú vyhradené zvyšné čísla, teda 49152 - 65535.

TCP garantuje spoľahlivé doručovanie a doručovanie v správnom poradí, na čo využíva princíp tzv. „sliding window“, ktoré zabezpečuje časovače a opätovné zasielanie paketov.



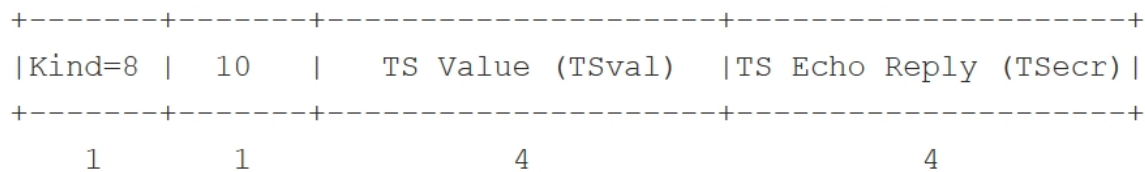
Obrázok 3.3: TCP hlavička (Zdroj: [4])

Pri návrhu protokolu TCP si jeho tvorcovia uvedomili, že v budúcnosti by mohli byť potrebné jeho úpravy a vylepšenia. Výsledkom tejto úvahy je položka Options, ktorá umožňuje pružné rozšírenie TCP hlavičky o voliteľné polia. Vďaka nim je možné vylepšovanie protokolu TCP o nové funkcie, a to bez obáv o narušenie spätnej kompatibility. Ich účelom je pomoc pri prekonávaní problémov v moderných sieťach, ktoré sa vyskytujú pri prenosoch veľkých objemov dát vysokou rýchlosťou. Jeden paket môže vo svojej TCP hlavičke obsahovať aj niekoľko rozšírení. Každé rozšírenie obsahuje jednobajtovú položku Kind, ktorá určuje typ rozšírenia. Ďalšie dve položky, Length a Data, sú voliteľné. Length poskytuje informáciu o celkovej dĺžke daného rozšírenia v bajtoch, Data obsahuje samotné hodnoty. Za posledným rozšírením sa nachádza oblasť Padding, ktorá zarovná TCP hlavičku na násobok 32 bitov. Typy najpoužívanejších TCP Options (v zátvorke je uvedené číslo, ktorým sú jednotlivé rozšírenia identifikované):

- End of Option list (0) – koniec zoznamu rozšírení
- No-Operation (1) – žiadna operácia
- Maximum Segment Size (2) – určuje najväčšie množstvo dát, ktoré je komunikujúce zariadenie schopné prijať v jedinom TCP segmente
- WSOPT – Window Scale (3) – zvýšenie veľkosti prijímacieho okna nad maximum 65 535 bajtov
- SACK Permitted (4) – povolenie selektívneho potvrdzovania
- SACK (5) – selektívne potvrdenie
- TSOPT – Timestamp Option (8) – zahrnutie časových známk

3.2.1 TCP Timestamp Option

Využitie TCP na vzdialené označenie fyzických zariadení v sieti Internet je možné vďaka rozšíreniu tohto protokolu, ktoré bolo zavedené za účelom zvýšenia rýchlosti a môžeme ho nájsť v RFC 1323 [11], z ktorého vychádza táto podkapitola. Konkrétne sa jedná o umožnenie vkladania časových známk (Timestamp Option) do paketov, ktoré sú využité pre RTTM (Round Trip Time Measurement) a PAWS (Protect Against Wrapped Sequences). Táto možnosť nahrádza dovtedy používanú dvojicu Echo a Echo Reply. Zahrnutie TCP Timestamp Option v TCP toku nie je automatické, musia byť pre to splnené určité podmienky. V oboch koncoch TCP spojenia musí TCP zásobník implementovať toto rozšírenie podľa RFC 1323 [11]. Táto podmienka je splnená vo všetkých rozšírených operačných systémoch. Druhou podmienkou je, aby iniciátor TCP spojenia zahrnul toto rozšírenie do počiatočného SYN paketu. Obe komunikujúce strany následne vkladajú TSOpt do všetkých ďalších paketov.



Obrázok 3.4: TCP Timestamps Option (Zdroj: [11])

TCP Timestamp Option v položke Data obsahuje dva údaje, ktorými sú dve časové známky s veľkosťou 32 bitov. TSval (Timestamp Value) je aktuálny čas odosielateľa v okamihu posielania správy. TSecr (Timestamp Echo Reply) je platná len v prípade, ak je v TCP hlavičke nastavený ACK bit. Obsahuje a tým zároveň potvrdzuje časovú známku obdržanú v TSval. Ak nie je táto položka platná, musí byť rovná nule.

TCP je symetrický protokol, ktorý umožňuje posielanie dát kedykoľvek v ktoromkoľvek smere. Môže sa teda vyskytnúť potvrdzovanie časových známok (timestamp echoing) v oboch smeroch. Pre jednoduchosť a zachovanie symetrie sú časové známky posielané aj potvrdzované v oboch smeroch. Vďaka kombinovaniu časových známok a ich potvrdeniu v jedinom TSOpt sa zvyšuje účinnosť.

3.2.1.1 RTTM

Presný a aktuálny odhad RTT je potrebný pre prispôsobenie sa zmenám v zaťažení siete a tým predchádzanie nestabilite v preťažených sieťach, známej ako „congestion collapse“. Presné meranie RTT je však náročné a jeho implementácie pred zavedením TCP Timestamp Option boli často chybné. Mnohé sa spoliehali na vzorku jediného paketu pripadajúceho na celé prenosové okno. Tento prístup bol však dostatočný len pre malé okná. Ak sa pozrieme na odhad RTT ako na problém spracovania signálu, kde má dátový signál určitú frekvenciu a je vzorkovaný s nižšou frekvenciou, nižšia vzorkovacia frekvencia porušuje Nyquistovo kritérium a môže byť zdrojom aliasingu. Problémom môže byť takisto strata paketov, kedy by do výpočtov RTT boli zahrnuté aj opätovne odoslané segmenty.

Riešením týchto problémov je využitie TCP Timestamp Option v metóde zvanej Round-Trip Time Measurement (RTTM). Použitie RTTM je obzvlášť potrebné pri veľkých oknách, kde hrozia nestability spôsobené aliasingom. RTTM sa počíta pomocou rozdielu časových známok v odoslanom pakete a k nemu prislúchajúcej správe ACK. Zvýšenie presnosti je umožnené vďaka priemerovaniu hodnôt RTT.

```

                <A, TSval=1, TSecr=120> ----->

<----- <ACK(A), TSval=127, TSecr=1>

                <B, TSval=5, TSecr=127> ----->

<----- <ACK(B), TSval=131, TSecr=5>

```

Obrázok 3.5: RTTM - Posielanie a potvrdzovanie časových známok (Zdroj: [11])

3.2.1.2 PAWS

PAWS je mechanizmus slúžiaci na identifikáciu a odmietnutie starých duplicitných paketov, ktoré by mohli narušiť nadviazané TCP spojenie. Pracuje v rámci jedného TCP spojenia, pričom využíva stav uložený v CCB (Connection Control Block). Predpokladá, že sú časové známky prítomné v každom pakete, pričom hodnoty v nich sú neklesajúce a monotónne. Základnou myšlienkou je, že paket môže byť zahodený, ak jeho časová známka nespĺňa túto podmienku. Keďže časové známky sú posielané v oboch smeroch, PAWS poskytuje ochranu pred duplicitnými dátovými segmentmi a rovnako aj ACK segmentami. Inicializácia PAWS nastáva pri prijatí {SYN} a {SYN, ACK} segmentov obsahujúcich časové známky. Porovnávanie časových známok prebieha v modulárnom 32-bitovom priestore nasledovne. Ak s a t sú časové známky, tak platí, že

$$s < t, \text{ ak } 0 < (t - s) < 2^{31} \quad (3)$$

Mechanizmus je však náchylný na útok DoS, kedy útočník podvrhne paket s časovou známkou výrazne vyššou, ako majú pakety korektnej komunikácie. Tým sa prepíše časová známka posledného prijatého paketu a to spôsobí, že všetky prichádzajúce pakety budú zahodené.

3.2.1.3 Timestamp clock

V oboch mechanizmoch sú časové známky neznamienkovými celočíselnými hodnotami v modulárnom 32 bitovom priestore. Časové známky obsahujú virtuálny čas nazývaný timestamp clock, ktorý musí byť aspoň približne úmerný reálnemu času, aby bolo možné merať skutočný RTT. Je dôležité chápať, že algoritmus PAWS nevyžaduje synchronizáciu odosielateľa a príjemcu. Odosielateľova časová známka je použitá na označenie paketov a odosielateľ používa potvrdenú časovú známku na určenie RTT. Príjemca berie časové známky ako monotónne rastúce sériové číslo, bez akejkoľvek potreby spájať ho so svojim časom. Z jeho pohľadu sa časová známka správa ako logické rozšírenie rádovo najvyšších bitov sekvenčného čísla. Kládie však isté nároky na frekvenciu

timestamp clock. Tá musí byť v rozmedzí 1 ms až 1 s za tik hodín. Tento interval spĺňa aj podmienky mechanizmu RTTM, ktorý nevyžaduje väčšie rozlíšenie ako granularitu časovača pre opätovné zaslanie, teda desiatky až stovky milisekúnd.

Metóda implementácie timestamp clock tak, aby spĺňal uvedené podmienky, závisí na systémovej softvéri a hardvéri. Najčastejšie používané frekvencie pre timestamp clock sú 2Hz, 10Hz, 100Hz, 250Hz a 1000Hz. Väčšina systémov resetuje svoj timestamp clock pri štarte. Ak dôjde k pretečeniu timestamp clock (k nastaveniu najvyššieho bitu), PAWS mechanizmus spôsobí zamrznutie spojenia zahodením všetkých prichádzajúcich segmentov. Tento stav pretrvá až do opätovnej zmeny hodnoty najvyššieho bitu. Použitím frekvencie timestamp clock z uvedeného rozsahu nastane takáto situácia každých 24,8 dní až 24800 dní. TCP spojenie, ktorého nečinnosť trvá dlhšie ako 24 dní, je veľmi neobvyklé, avšak kladenie akýchkoľvek obmedzení na životnosť spojenia je nežiaduca. Preto PAWS mechanizmus pri nečinnosti spojenia trvajúcej 24 dní zneplatní poslednú časovú známku. Prijatie nového segmentu s časovou známkou, ktorá by neprešla testom tak už neznamená jeho zahodenie.

4 Identifikácia zariadení v rámci počítačových sietí

Pri komunikácii po sieti Internet je z mnohých dôvodov potrebná možnosť jednoznačnej identifikácie komunikujúceho zariadenia. Primárnym spôsobom identifikácie zariadení v počítačovej sieti je využitie ich fyzickej a logickej adresy.

Fyzická MAC adresa je priradená sieťovej karte priamo pri jej výrobe a je unikátnym identifikátorom sieťového zariadenia. Hovorí sa jej tiež fyzická adresa a pozostáva zo 48 bitov. IP adresa jednoznačne identifikuje zariadenie v počítačovej sieti, ktoré používa protokol IP. Verzia IPv4 má 32 bitov, nová verzia IPv6, zavedená z dôvodu nedostatku pôvodných adries, má 128 bitov. Práve MAC adresa ako jedinečný údaj by mala slúžiť pre nepopierateľnú identifikáciu hardvéru, avšak možnosť zmeniť MAC adresu zariadenia softvérovo do značnej miery znižuje dôveru v nevyvrátiteľnú identifikáciu pomocou tejto techniky.

Taktiež logická IP adresa sa ukazuje ako veľmi nedostatočný identifikátor, najmä vzhľadom na skutočnosť, že často býva pridelovaná dynamicky, a tak napríklad zmenou bodu pripojenia toho istého zariadenia alebo vypršaním platnosti IP adresy už nie sme schopní povedať, či sa jedná o to isté zariadenie. Dokonca pri použití NATu pristupuje niekoľko rôznych zariadení z lokálnej siete k Internetu pod jedinou verejnou IP adresou. Týmto spôsobom sa šetria verejné adresy, avšak komplikuje sa tým možnosť identifikovať zariadenia umiestnené za NATom.

Idea využiť údaje obsiahnuté v súboroch cookies sa taktiež ukazuje ako nedostatočná, pretože tieto súbory nemusia byť vždy k dispozícii, či už vďaka ich nedostupnosti identifikujúcemu zariadeniu, alebo použitiu šifrovania pri ich prenosoch.

Okrem týchto možností existujú postupy a nástroje, ktoré pri identifikácii využívajú iné údaje. Líšia sa v spôsobe získavania týchto údajov a v požiadavkách na spoluprácu zariadenia.

4.1 Techniky identifikácie z hľadiska spolupráce označovaného zariadenia

Na základe spolupráce zariadenia pri jeho identifikácii môžeme značkovacie techniky rozdeliť do troch skupín. Prvou je skupina pasívnych techník, pri ktorých musí byť útočník schopný zachytiť komunikáciu zo zariadenia, ktoré chce označiť. Opakom sú aktívne techniky vyžadujúce schopnosť útočníka zahájiť spojenie. Tretiu skupinu tvoria semipasívne techniky, kde je spojenie zahájené označovaným zariadením a až následne je vyžadovaná schopnosť útočníka cez toto spojenie komunikovať a interagovať.

Každý z uvedených prístupov má svoje klady i zápory. Napríklad pasívne techniky sú nedetekovateľné sledovaným zariadením, pasívne a semipasívne môžu byť aplikované aj v prípade, kedy sa sledované zariadenie nachádza za NATom alebo firewallom, semipasívne a aktívne techniky môžu byť zase aplikované naprieč dlhším obdobím.

4.2 Súčasné metódy identifikácie

V súčasnosti existuje niekoľko kvalitných nástrojov pre jednoznačnú identifikáciu operačného systému zariadenia alebo typu zariadenia. Identifikácia operačného systému funguje na princípe využitia určitých parametrov protokolu TCP, ktorými sú:

- Initial packet size (16 bitov)
- Initial TTL (8 bitov)
- Window size (16 bitov)
- Max segment size (16 bitov)
- Window scaling value (8 bitov)
- "don't fragment" flag (1 bit)
- "sackOK" flag (1 bit)
- "nop" flag (1 bit)

Každý operačný systém i jeho verzia nastavuje tieto parametre na odlišné východiskové hodnoty. Zachytením týchto hodnôt je možné vytvoriť „odtlačok“ skúmaného systému, pomocou ktorého sa dajú pomerne spoľahlivo určiť operačné systémy. Popis tohto postupu bol prevzatý z [32].

Príkladom takéhoto nástroja je p0f [27], ktorý sa radí medzi metódy využívajúce pasívne techniky. Zachytáva sieťovú komunikáciu a určí operačný systém zariadenia analyzovaním určitých položiek v zachytených paketoch (TTL, TOS). Je schopný taktiež detekovať spôsob pripojenia zariadenia k sieti, určité paketové filtre, firewally a rozloženia NAT.

Ďalším programom je napríklad program Nmap [21], ktorý je možné použiť pre rozsiahle siete i jediné zariadenie. Patrí do skupiny aktívnych techník, pretože využíva špeciálne upravené pakety a následne analyzuje prijaté odpovede. Okrem svojej hlavnej funkcie, skenovania portov a identifikácie služieb bežiacich na týchto portoch vrátane ich verzií, dokáže určiť aj operačný systém skenovaného zariadenia, typ používaného firewallu alebo paketového filtra a taktiež typ zariadenia. Pri zisťovaní operačného systému sa využívajú práve časové známky umiestnené v TCP Timestamp Option, ktoré boli popísané v predchádzajúcej kapitole (3.2.1).

Zástupcami takýchto programov sú aj xprobe2 [1], Ettercap [6] alebo RING [22].

Špecifická skupina nástrojov je zameraná na identifikáciu hardvéru v sieti, ktoré využívajú mierne odlišné techniky. V nasledovnej podkapitole vysvetlím princíp metódy vzdialeného označenia fyzických zariadení, ktorej implementácia a následné testovanie sú predmetom tejto práce.

4.3 Vzdialené označenie fyzických zariadení

Implementovaná metóda nevyžaduje žiadnu spoluprácu od sledovaného zariadenia, patrí teda do skupiny pasívnych techník. Táto skutočnosť je umožnená faktom, že väčšina TCP zásobníkov implementuje TCP Timestamp Option podľa RFC 1323 [11] a teda každá komunikujúca strana zahŕňa časovú známku do každého odchádzajúceho paketu. Z týchto informácií je identifikátor schopný určiť časovú odchýlku identifikovaného zariadenia. Z uvedených skutočností je jasné, že identifikátorom môže byť akékoľvek zariadenie, ktoré je schopné zachytávať TCP pakety obsahujúce časové známky od sledovaného zariadenia. Vďaka tomu je táto metóda použiteľná v rôznych častiach siete, či už v jej uzloch (router), v koncovom zariadení alebo dokonca v systémoch, s ktorými označované zariadenie často komunikuje, napríklad vyhľadávač.

4.3.1 Princíp metódy

Samotné určenie časovej odchýlky prebieha nasledovne. Prvým krokom je zachytenie komunikácie obsahujúcej časové známky, pričom všetky dáta musia patriť do jedného TCP toku. V uvedených vzťahoch bude použité nasledovné označenie. Časová známka obsiahnutá v TSopt i-tého paketu je T_i , čas prijatia i-tého paketu je t_i , Hz označuje frekvenciu timestamp clock a pozorovaný offset i-tého paketu je y_i .

$$x_i = t_i - t_1 \quad (4)$$

$$v_i = T_i - T_1 \quad (5)$$

$$w_i = \frac{v_i}{Hz} \quad (6)$$

$$y_i = w_i - x_i \quad (7)$$

$$O_T = \{(x_i, y_i): i \in \{1, \dots, |T|\}\} \quad (8)$$

Množina offsetov O_T je množinou bodov v dvojrozmernej sústave a môžeme ňou preložiť regresnú priamku, ktorej smernica udáva približnú hodnotu časovej odchýlky.

Frekvencia timestamp clocku nie je pred výpočtom známa, je ju však možné určiť zo zachytených dát. Postup spočíva v nájdení priamky, ktorou je možné preložiť body z množiny I

$$I = \{(x_i, v_i): i \in \{1, \dots, |T|\}\} \quad (9)$$

a následne jej sklon zaokrúhliť na najbližšie celé číslo.

Kľúčovým bodom výpočtu je teda nájdenie regresnej priamky. Pre túto úlohu existuje množstvo rôznych postupov. Popis metód, ktoré som využil pri implementácii je obsahom nasledovnej kapitoly (5).

4.3.2 Možnosti využitia

Uplatnenie jednoznačnej identifikácie zariadenia je veľmi široké. Príkladom môže byť vyšetrowanie počítačovej kriminality, kedy došlo k porušeniu zákona hackerským útokom alebo zdieľaniu nelegálneho obsahu. Metóda v takom prípade slúži na potvrdenie, či sa dané zariadenie podieľalo na danej činnosti. Poskytnuté výsledky nie sú stopercentné, avšak v kombinácii s inými zdrojmi informácii môžu napomôcť pri sledovaní zariadenia, ktoré využíva rôzne prístupové body a tým aj jeho aktívne blokovanie inou metódou. Príkladom je nasledovná situácia. Z nejakého zariadenia dôjde k nelegálnej činnosti, pričom sa podarí získať časové známky z daného toku. Je určená časová odchýlka tohto zariadenia, ktorá je vložená do špeciálnej databázy. Na serveri obsahujúcom dôverné informácie je nasadený IDS (Intrusion Detection System). Všetky zariadenia, s ktorými tento server komunikuje, majú priebežne vypočítavanú svoju časovú odchýlku. Tá je porovnávaná s databázou a v prípade zhody alebo výraznej podobnosti odchýlok je vykonaná reakcia systému, ktorou môže byť zvýšenie citlivosti systému IDS na aktivity, ktoré sú vykonávané z podozrivého zariadenia. Inou možnosťou je dočasné zakázanie prístupu, kým sa neoverí, že sa skutočne nejedná o zariadenie z databázy, napríklad biometrickou autentizáciou užívateľa.

Taktiež je možné zistenie počtu zariadení za routerom s prekladom sieťových adries (NAT). Prvým krokom je rozdelenie zachytených dát podľa rozdielnych sekvencií časových známk. Tento úkon je pomerne jednoduchý, komplikáciou môže byť situácia, kedy majú odlišné zariadenia približne rovnakú hodnotu časových známk v určitom momente. Následne sa pre každú takúto množinu hodnôt aplikuje metóda výpočtu časovej odchýlky. Počet unikátnych časových odchýlok následne zodpovedá počtu zariadení, ktoré boli aktívne počas zachytávania komunikácie.

Situácia, kedy je využívaný DHCP server značne znemožňuje mapovanie siete, keďže pozorovateľ nemôže jednoznačne identifikovať jednotlivé uzly pomocou IP adresy a musí sa vysporiadať s kombináciou údajov z rôznych klientov. Aj tu teda nachádza svoje potenciálne uplatnenie metóda vzdialeného označenia fyzického zariadenia. Dokonca by jej použitím bolo možné s určitou pravdepodobnosťou sledovať priradovanie IP adries fyzickým zariadeniam.

Skúmanie bloku IP adries pre určenie, či sú využívané virtuálnymi strojmi v rámci honeynetu alebo odhalenie zdroja anonymných dát za pomoci dostupných neanonymných údajov z rovnakej linky sú ďalšími možnosťami využitia tejto metódy.

4.3.3 Ochrana pred identifikáciou

Keďže metóda je závislá na časových známkach obsiahnutých v prenášaných dátach, je niekoľko možností, ktorými sa dá voči identifikácii touto metódou brániť. Jednoduchým spôsobom sa javí zakázanie vkladania týchto informácií do odchádzajúcich paketov, čo je možné dosiahnuť úpravou registrov. Zníži sa tým však výkon protokolu TCP, keďže nebude možné použiť mechanizmus RTTM. Ďalším spôsobom je využiť udržiavanie systémového času pomocou NTP a vďaka tomu zredukovať časovú odchýlku timestamp clocku. Synchronizácia však musí byť častá, aby sa zamedzilo prejavaniu časovej odchýlky. Vykonanie presného odhadu frekvencie oscilátora, od ktorého sa odvíja systémový čas pri štarte systému sa javí ako ďalšia možnosť. Zariadenie by taktiež mohlo znáhodňovať alebo maskovať časové známky vynásobením náhodnou konštantou, ktoré vkladá do odchádzajúcej komunikácie.

5 Lineárna regresia

Táto kapitola obsahuje popis matematických metód, ktoré boli využité pri implementácii metódy vzdialeného označenia fyzického zariadenia. Popísaná metóda vyžaduje pre správny výpočet časovej odchýlky matematickú metódu, ktorá je schopná z množiny bodov určiť ich regresnú priamku. Regresná analýza je označenie pre skupinu štatistických metód, pomocou ktorých odhadujeme hodnotu istej náhodnej veličiny na základe znalosti iných veličín. Regresia v názve tejto skupiny metód znamená, že aj keď sú pozorované dáta premenné, majú tendenciu pohybovať sa okolo svojho stredy. Lineárnosť zase odkazuje na fakt, že využité rovnice vyjadrujúce závislosť závislej premennej sú lineárneho charakteru.

Závislá premenná Y závisí na nezávislých premenných X_1, X_2, \dots, X_n . V prípade, že vzťahy medzi premennými nie sú lineárne, je možné tieto vzťahy linearizovať použitím vhodných transformácií. Model tejto lineárnej regresie má tvar

$$Y = B_0 + B_1 \cdot X_1 + B_2 \cdot X_2 + \dots + B_n \cdot X_n + e, \quad (10)$$

pričom regresné koeficienty $B_0, B_1, B_2, \dots, B_n$ sú neznámymi parametrami tejto lineárnej rovnice a náhodná premenná e je chyba so strednou hodnotou rovnou nule. Teoretický model nikdy nedokáže presne popísať reálny systém. Je preto nutné uspokojiť sa s približným modelom, ktorého výsledky približne zodpovedajú skutočnosti. Tento model získame odhadom hodnôt regresných koeficientov. V prípade, kedy máme dva modely dosahujúce rovnaké výsledky, volíme ten jednoduchší. Niekedy je dokonca vhodné vynechať niektoré nezávislé premenné.

Pre potreby implementovanej metódy vo vzťahoch vystupuje jedna závislá premenná, ktorou je časová známka v paketoch, a jedna nezávislá premenná, ktorou je čas prijatia paketu. Priamka vyjadrujúca závislosť má rovnicu

$$y = k \cdot x + q, \quad (11)$$

kde k je smernica priamky určujúca tangens uhla zovretého priamkou a osou X a q je konštanta vyjadrujúca priesečník tejto priamky s osou Y . Smernica regresnej priamky zodpovedá hľadanej časovej odchýlke. Odchýlka e je v tomto prípade spôsobená oneskorením na sieti, ktoré je približne konštantné, avšak mierne kolíše v závislosti na zaťažení siete.

Úloha určiť k a q nie je triviálna, pretože dvojice $[x, y]$ takmer nikdy neležia na priamke. Vhodné odhady parametrov sa získavajú pomocou rôznych metód, pričom ich voľba závisí na niekoľkých faktoroch. Jedným z nich je výskyt chýb v meraní závislých a nezávislých premenných. Ďalším faktorom je výskyt takzvaných odľahlých dát, ktorých odchýlka je značná oproti ostatným dátam. Podľa toho, či je metóda robustná voči porušeniu jej predpokladov, sa metódy delia na dve skupiny. Prvá, skupina robustných metód, obsahuje tie metódy, ktoré sa dokážu vysporiadať

s limitmi tradičných metód a dosahujú relatívne presné výsledky. Patrí tu napríklad Ortogonálna regresná metóda, M-estimation, Theil-Sen estimator, Random Sample Consensus, Bayesovská lineárna regresia, či Metóda najmenších orezaných štvorcov. Druhá skupina obsahuje Metódu najmenších štvorcov, Metódu vážených mediánov, Lineárne programovanie a ďalšie. Metódy, ktoré som využil pri implementácii, budú popísané v nasledujúcich podkapitolách.

5.1.1 Metóda najmenších štvorcov

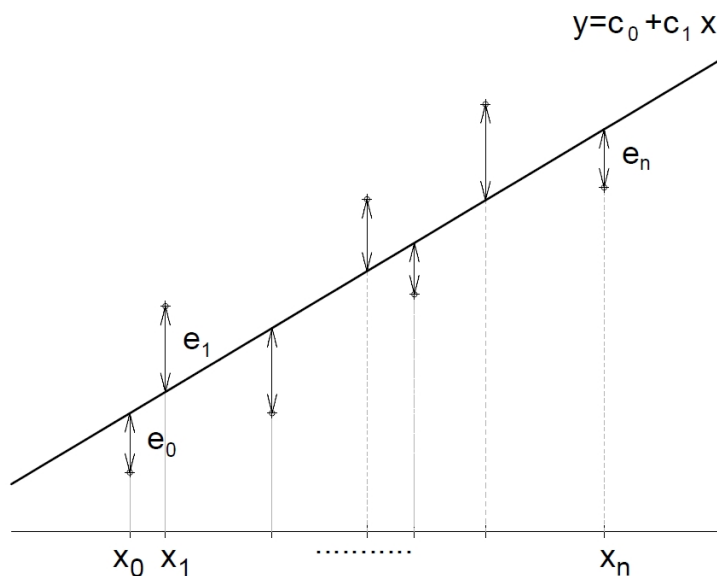
Metóda najmenších štvorcov je matematicko-štatistická metóda, používaná najmä pri spracovaní nepresných dát. V základnej podobe je určená pre riešenie nekompatibilných sústav lineárnych rovníc, vďaka čomu je vlastne ekvivalentná lineárnej regresii. Jej použitie je vhodné v situáciách, kde nezávislé premenné nie sú zaťažené chybami. Popis tejto numerickej metódy pochádza z [7].

Máme množinu bodov x_i , $i = 0, \dots, n$ a funkčné hodnoty y_i v nich. Aproximácia priamkou spočíva v hľadaní rovnice priamky

$$y = c_0 + c_1 \cdot x, \quad (12)$$

ktorá bude prechádzať bodmi $[x_i, y_i]$, $i = 1, \dots, n$. Chybu aproximácie v i -tom bode označíme ako e_i , pričom sa jedná o zvislú vzdialenosť bodu od regresnej priamky. Chyby aproximácie sú znázornené na nasledovnom obrázku (Obrázok 5.1) a vypočítajú sa ako

$$e_i = y_i - y(x_i) = y_i - c_0 - c_1 \cdot x_i. \quad (13)$$

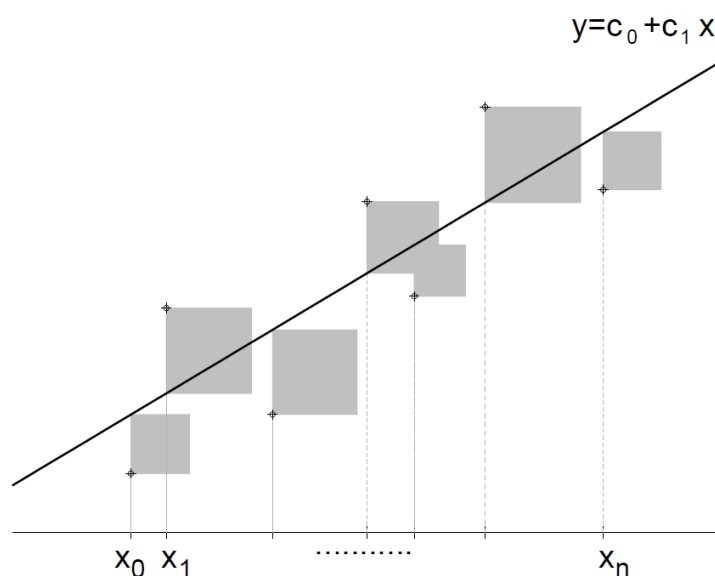


Obrázok 5.1: Odchýlky e_i (Zdroj: [7])

Keďže body $[x_i, y_i]$ sú dané, chyba závisí len na hodnotách koeficientov priamky c_0 a c_1 . Ukazuje sa, že vhodným kritériom pre určenie najlepšej priamky je, aby súčet druhých mocnín, teda štvorcov, chýb v jednotlivých bodoch bol čo najmenší, teda minimálny. Značíme ho ako ρ^2 . Situácia je znázornená na nasledovnom obrázku (Obrázok 5.2).

Chceme teda minimalizovať funkciu

$$\rho^2(c_0, c_1) = \sum_{i=0}^n (y_i - c_1 \cdot x_i)^2 . \quad (14)$$



Obrázok 5.2: Hľadaná priamka, pre ktorú je súčet obsahov štvorcov minimálny (Zdroj: [7])

Veličinu ρ^2 nazývame kvadratická odchýlka. Z diferenciálneho počtu funkcií viacerých premenných je známe, že nutnou podmienkou pre dosiahnutie minima je splnenie rovníc

$$\frac{\partial(\rho^2)}{\partial c_0} = 0 \quad a \quad \frac{\partial(\rho^2)}{\partial c_1} = 0 . \quad (15), (16)$$

Uskutočnením parciálnych derivácií dostaneme

$$\frac{\partial(\rho^2)}{\partial c_0} = -2 \left(\sum_{i=0}^n y_i - c_0(n+1) - c_1 \sum_{i=0}^n x_i \right) , \quad (17)$$

$$\frac{\partial(\rho^2)}{\partial c_1} = -2 \left(\sum_{i=0}^n x_i y_i - c_0 \sum_{i=0}^n x_i - c_1 \sum_{i=0}^n x_i^2 \right) . \quad (18)$$

Ak teraz položíme vypočítané parciálne derivácie rovné 0, dostaneme normálne rovnice s neznámymi c_0 a c_1 .

$$c_0(n+1) + c_1 \sum_{i=0}^n x_i = \sum_{i=0}^n y_i , \quad (19)$$

$$c_0 \sum_{i=0}^n x_i + c_1 \sum_{i=0}^n x_i^2 = \sum_{i=0}^n x_i y_i . \quad (20)$$

Pokiaľ medzi uzlami x_i nájdeme aspoň dva rôzne, teda neplatí, že $x_0 = x_1 = \dots = x_n$, alebo ak sú vektory $[1, 1, \dots, 1]$ a $[x_0, x_1, \dots, x_n]$ lineárne nezávislé, má táto sústava jediné riešenie.

Alternatívnym prístupom je využitie matic. Opäť máme množinu bodov x_i , $i = 0, \dots, n$ a funkčné hodnoty y_i v nich. Aproximácia priamkou spočíva v hľadaní rovnice priamky

$$y = c_0 + c_1 \cdot x, \quad (21)$$

ktorá bude prechádzať bodmi $[x_i, y_i]$, $i = 1, \dots, n$. Pre túto priamku, resp. jej koeficienty c_0 a c_1 by malo platiť

$$\begin{aligned} y_0 &\cong c_0 + c_1 \cdot x_0 \\ y_1 &\cong c_0 + c_1 \cdot x_1 \\ &\vdots \\ y_n &\cong c_0 + c_1 \cdot x_n \end{aligned} \quad (22)$$

Maticový zápis vyzerá nasledovne

$$y \cong Zc, \text{ kde } y = \begin{pmatrix} y_0 \\ y_1 \\ \vdots \\ y_n \end{pmatrix}, Z = \begin{pmatrix} 1 & x_0 \\ 1 & x_1 \\ \vdots & \vdots \\ 1 & x_n \end{pmatrix} \text{ a } c = \begin{pmatrix} c_0 \\ c_1 \end{pmatrix}. \quad (23)$$

Riešiť takúto úlohu má zmysel až pri minimálne troch bodoch (pri dvoch bodoch je riešenie triviálne, pri jednom bode nemá jednoznačné riešenie). Ak nahradíme \cong znakom $=$, dostávame sústavu rovníc, ktorá obsahuje viac rovníc ako neznámych, teda preurčenú sústavu rovníc. Skutočné riešenie by mala len v prípade, keby všetky zadané body ležali na jednej priamke. Inak nemá riešenie, preto hľadáme vektor c taký, pre ktorý je sústava splnená najlepšie. Je to taký vektor, pre ktorý je súčet druhých mocnín rozdielu ľavých a pravých strán minimálny, teda

$$\sum_{i=0}^n (y_i - c_0 - c_1 \cdot x_i)^2. \quad (24)$$

Sústavu teraz môžeme zapísať pomocou matice Z ako

$$Z^T Zc = Z^T y, \quad (25)$$

z čoho si vyjadríme neznámy vektor c ako

$$c = \frac{Z^T y}{Z^T Z}. \quad (26)$$

5.1.2 Ortogonálna regresia

Ďalšou metódou, ktorú je možné použiť pre lineárnu regresiu je Ortogonálna regresná metóda, nazývaná tiež Metóda totálnych najmenších štvorcov. V situáciách, kde sa vyskytujú výrazné odchýlky pri pozorovaní nezávislých premenných rovnako ako závislých premenných je použitie obvyčajnej metódy najmenších štvorcov nie práve najvhodnejšie. Ortogonálna regresná metóda je použiteľná v oboch situáciách. Na rozdiel od metódy najmenších štvorcov nevyužíva pri výpočte

vertikálnu vzdialenosť bodov od regresnej priamky, namiesto nej používa ich kolmú (ortogonálnu) vzdialenosť.

Regresná priamka má už spomínaný tvar

$$y = k \cdot x + q, \quad (27)$$

priamka na ňu kolmá má smernicu $-\frac{1}{k}$ a jej rovnica je

$$y' = -\frac{x}{k} + q'. \quad (28)$$

Ak táto kolmá priamka prechádza bodom $[x_0, y_0]$, jej rovnica vyzerá

$$y' = -\frac{x}{k} + \frac{x_0}{k} + y_0. \quad (29)$$

Kolmá priamka pretína regresnú priamku v bode $[x_i, y_i]$, ktorého súradnice sa vypočítajú ako

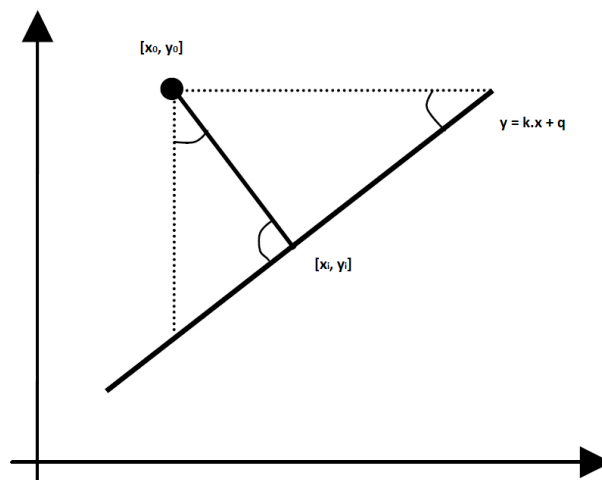
$$x_i = \frac{x_0 + k \cdot y_0 - k \cdot q}{k^2 + 1} \quad (30)$$

$$y_i = k \cdot x_i + q$$

Bod $[x_i, y_i]$ je najbližším bodom na regresnej priamke vzdialeným od bodu $[x_0, y_0]$ o vzdialenosť e

$$e = \sqrt{(x_0 - x_i)^2 + (y_0 - y_i)^2}. \quad (31)$$

Popísanú situáciu ilustruje nasledovný obrázok (Obrázok 5.3).



Obrázok 5.3: Ortogonálna regresná metóda

Model chyby merania definujeme podľa [2] nasledovne. Pozorované náhodné premenné

$$(X_i, Y_i), \quad i = 1, \dots, n \quad (32)$$

s príslušnými skutočnými hodnotami

$$(x_i, y_i), \quad i = 1, \dots, n, \quad (33)$$

kde $x_i = \mathbb{R}^m$ a $z_i = \mathbb{R}$. Premennú X_i v nasledovnom texte pre zjednodušenie považujeme za jednorozmernú. Náhodnú chybu spojenú s x_i označíme ako δ_i , náhodnú chybu spojenú s y_i ako ϵ_i . Potom

$$X_i = x_i - \delta_i \quad (34)$$

$$Y_i = y_i - \epsilon_i \quad (35)$$

Nakoniec predpokladajme, že y_i je dané ako funkcia x_i a množiny parametrov $\beta \in \mathbb{R}^p$

$$y_i = f(x_i; \beta) , \quad (36)$$

čo je možné zapísať alternatívne ako

$$Y_i = f(X_i + \delta_i; \beta) - \epsilon_i . \quad (37)$$

Akýkoľvek postup pre odhad parametrov β musí brať do úvahy, že obe náhodné premenné podliehajú náhodným chybám. Využitím kolmej vzdialenosti definujeme vzdialenosť r_i od bodu (X_i, Y_i) ku krivke $f(\tilde{x}; \tilde{\beta})$, kde „ $\tilde{\cdot}$ “ značí neodlišujúcu sa hodnotu premennej.

$$r_i^2 = \min_{\tilde{\epsilon}_i, \tilde{\delta}_i} \{ \tilde{\epsilon}_i^2 + \tilde{\delta}_i^2 \} \quad (38)$$

Najpravdepodobnejší odhad parametrov $\hat{\beta}$ je taký, ktorý minimalizuje sumu druhých mocnín r_i .

$$\hat{\beta} = \min_{\tilde{\beta}, \tilde{\delta}, \tilde{\epsilon}} \sum_{i=1}^n \{ \tilde{\epsilon}_i^2 + \tilde{\delta}_i^2 \} \quad (39)$$

Odstránením $\tilde{\epsilon}$ a pridaním váh k jednotlivým pozorovaniám získame neviazaný problém minimalizácie, ktorý nazveme Problém váženej ortogónálnej regresie.

$$\min_{\tilde{\beta}, \tilde{\delta}} \sum_{i=1}^n w_i^2 \{ [f(X_i + \tilde{\delta}_i; \tilde{\beta}) - Y_i]^2 + d_i^2 \tilde{\delta}_i^2 \} , \quad (40)$$

kde $w_i \geq 0$ a $d_i > 0$. Takto definovaný problém môže byť použitý v štatistických aplikáciách, pri aproximácii krivkou a v iných aplikáciách, kde chyby a s nimi spojené váhy nemajú žiadny vedľajší štatistický význam.

5.1.3 Rekurzívna metóda najmenších štvorcov

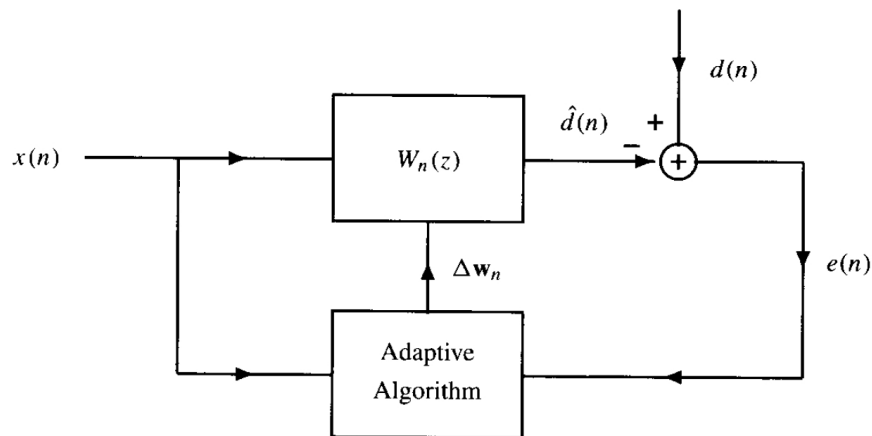
Digitálne filtre sú neodmysliteľnou súčasťou procesu spracovania signálov. Filtre sú použité pre dosiahnutie požadovaných spektrálnych charakteristík signálu, napríklad pre odstránenie šumu a rušivých signálov. Problémom pri návrhu filtrov je skutočnosť, že mnohé signály sú nestacionárne. Z toho vyplýva potreba vytvárať adaptívne filtre, ktoré upravujú svoje parametre podľa nejakého algoritmu. Adaptívne filtre sa dokážu prispôsobiť neznámemu prostrediu a sledovať signál alebo charakteristiku systému s meniacim sa časom. Odhad signálu sa získa ako

$$\hat{d}(n) = \sum_{k=0}^p w_n(k) x(n - k) , \quad (41)$$

kde $w_n(k)$ je hodnota k -tého koeficientu filtra v čase n . V mnohých ohľadoch je návrh adaptívnych filtrov náročnejší oproti návrhu filtrov pre stacionárne signály. Pre každú hodnotu n je totiž potrebné nájsť množinu optimálnych koeficientov filtra $w_n(k)$, pre $k = 0, 1, \dots, p$. Problém môže byť podstatne zjednodušený, ak poľavíme z požiadavky, že w_n minimalizuje chybu v každom čase n a zväžíme namiesto toho aktualizáciu koeficientov spôsobom

$$w_{n+1} = w_n + \Delta w_n, \quad (42)$$

kde Δw_n je korekcia aplikovaná na koeficienty w_n v čase n pre vytvorenie nových koeficientov w_{n+1} v čase $n+1$. Medzi takéto filtre, presnejšie rekurzívne adaptívne filtre, patrí Rekurzívna metóda najmenších štvorcov. Jej nasledovný popis pochádza z [10].



Obrázok 5.4: Bloková schéma adaptívneho filtra (Zdroj: [10])

Aby sa znížila závislosť na dávnych vstupných hodnotách, zavádza sa váhový faktor λ pre každú vzorku v nákladovej funkcii

$$\varepsilon(n) = \sum_{i=0}^n \lambda^{n-i} |e(i)|^2, \quad (43)$$

kde s použitím aktuálnych koeficientov filtra

$$w_n = [w_n(0), w_n(1), \dots, w_n(p)]^T \quad (44)$$

je chyba signálu $e(i)$, teda rozdiel medzi požadovaným signálom a odfiltrovaným výstupom v čase i vypočítaná pre všetky časy $I \leq i \leq n$ ako

$$e(i) = d(i) - y(i) = d(i) - w_n^T \cdot x(i). \quad (45)$$

Keď $\lambda = 1$, majú všetky chyby rovnakú váhu a hovoríme o Rekurzívnej metóde najmenších štvorcov s rastúcim oknom. Ak platí $0 < \lambda < 1$, tak sa vplyv dávnych chýb exponenciálne znižuje. Čím je váhový faktor λ menší, tým majú predchádzajúce hodnoty menší vplyv a filter sa stáva náchylnejší na nové hodnoty, čoho dôsledkom je kolísanie hodnôt koeficientov. Získavame tak Metódu exponenciálne vážených najmenších štvorcov, kde ε je tzv. zabúdaci faktor (forgetting factor).

Minimalizácia nákladovej funkcie parciálnymi deriváciami s ohľadom na všetky koeficienty w_n

$$\frac{\partial \varepsilon(n)}{\partial w_n^*(k)} = \sum_{i=0}^n y \lambda^{n-i} e(i) \frac{\partial e^*(i)}{\partial w_n^*(k)} = - \sum_{n=0}^n \lambda^{n-i} e(i) x^*(i-k) = 0, \quad (46)$$

pre $k = 0, 1, \dots, p$. Dosadením za $e(i)$ a úpravou rovnice získame

$$\sum_{l=0}^p = w_n(l) \left[\sum_{i=0}^n \lambda^{n-i} x(i-l) x^*(i-k) \right] = \sum_{i=0}^n \lambda^{n-i} d(i) x^*(i-k). \quad (47)$$

Maticový zápis rovnice má po úprave tvar

$$w_n = \frac{r_{dx}(n)}{R_x(n)}, \quad (48)$$

kde $R_x(n)$ je exponenciálne vážená deterministická autokorelačná matica pre x_n

$$R_x(n) = \sum_{i=0}^n \lambda^{n-i} x^*(i) x^T(i) \quad (49)$$

a $r_{dx}(n)$ je deterministická krížová korelácia medzi $d(n)$ a $x(n)$

$$r_{dx}(n) = \sum_{i=0}^n \lambda^{n-i} d(i) x^*(i) \quad (50)$$

Chceme derivovať rekurzívne riešenie tvaru

$$w_n = w_{n-1} + \Delta w_{n-1}, \quad (51)$$

kde Δw_{n-1} je korekcia aplikovaná na riešenie v čase $n-1$.

Z uvedených vzťahov môžeme odvodiť

$$r_{dx}(n) = \lambda r_{dx}(n-1) + d(n) x^*(n) \quad (52)$$

$$R_x(n) = \lambda R_x(n-1) + x^*(n) x^T(n) \quad (53)$$

Pre ďalšie výpočty je potrebná inverzia $R_x^{-1}(n)$, ktorú pre zjednodušenie budeme odteraz značiť ako $P(n)$.

$$R_x^{-1}(n) = \lambda^{-1} R_x^{-1}(n-1) - \frac{\lambda^{-2} R_x^{-1}(n-1) x^*(n) x^T(n) R_x^{-1}(n-1)}{1 + \lambda^{-1} x^T(n) R_x^{-1}(n-1) x^*(n)} \quad (54)$$

Zavedieme prírastkový faktor

$$g(n) = \frac{\lambda^{-1} P(n-1) x^*(n)}{1 + \lambda^{-1} x^T(n) P(n-1) x^*(n)}, \quad (55)$$

ktorý môžeme upraviť na tvar

$$g(n) = P(n) x^*(n) \Rightarrow R_x(n) g(n) = x^*(n). \quad (56)$$

Aby sme dokončili rekurziu, musíme derivovať rovnicu časového prírastku pre vektor koeficientov

$$w_n = \lambda P(n) r_{dx}(n-1) + d(n) P(n) x^*(n). \quad (57)$$

Po úprave dostávame

$$w_n = w_{n-1} + \alpha(n) g(n), \quad (58)$$

kde $\alpha(n)$ je rozdiel medzi $d(n)$ a odhadom $\hat{d}(n)$, ktorý je vytvorený aplikovaním predchádzajúcej množiny koeficientov na nové dáta

$$\alpha(n) = d(n) - w_{n-1}^T x(n) . \quad (59)$$

Ak je hodnota $\alpha(n)$ malá, súčasná množina koeficientov daného filtra je blízko ich optimálnych hodnôt a sú na ne uplatňované len drobné korekcie. V opačnom prípade je potrebná výraznejšia úprava koeficientov pre zlepšenie odhadu $\hat{d}(n)$.

Ostáva ešte popísať inicializáciu výpočtu. Keďže algoritmus zahŕňa rekurzívnu aktualizáciu vektora w_n a inverznej autokorelačnej matice $P(n)$, sú potrebné počiatočné podmienky

$$R_x(0) = \delta I \quad a \quad w_0 = 0 . \quad (60)$$

Na rozdiel od metódy LMS (Least Mean Squares), ktorá taktiež patrí medzi rozšírené metódy v oblasti adaptívnych filtrov, vyžaduje popísaná metóda väčšie množstvo výpočtov, ktoré však vedie k vysokej miere konvergencie. Súhrn Rekurzívnej metódy exponenciálne vážených najmenších štvorcov je obsahom Prílohy A.

5.1.4 Metóda aktualizácie odhadu časovej odchýlky pri príchode paketu váženým priemerom

Tento postup aktualizácie odhadu hodnoty časovej odchýlky pri príchode paketu (v nasledovnom kontexte je slovom paket myslený TCP paket obsahujúci časovú známku v TCP Timestamp Option) za použitia váženého priemeru bol navrhnutý ako alternatíva k rekurzívnej metóde najmenších štvorcov. Tento prístup je jednoduchý a pozostáva z malého množstva operácií. Je pravdepodobné, že metóda v nasledovných testoch nedosiahne také dobré výsledky, ako rekurzívna metóda najmenších štvorcov. Ide skôr o pokus nájsť výpočtovo nenáročnú metódu, ktorá by mohla byť uplatnená v zariadení s obmedzenými zdrojmi pre spracovanie paketu, napríklad v routeri.

Princíp navrhutej metódy je nasledovný. Namiesto ukladania všetkých údajov o danom TCP toku je v pamäti uchovávaných len niekoľko informácií. Sú to časová odchýlka s , priesečník regresnej priamky s osou Y $intersect_Y$, počet paketov v danom toku N , časová známka z prvého paketu t_{s0} a čas príchodu prvého paketu t_{r0} . Pri zachytení i tého paketu získame dva údaje, ktorými sú čas príchodu paketu t_{ri} a časová známka v ňom obsiahnutá t_{si} . Tieto údaje sú následne upravené

$$t_s = t_{si} - t_{s0} \quad a \quad t_r = t_{ri} - t_{r0} . \quad (61)$$

Máme teda bod $[t_s, t_r]$ a môžeme určiť smernicu priamky q , ktorá prechádza týmto bodom a priesečníkom regresnej priamky s osou Y

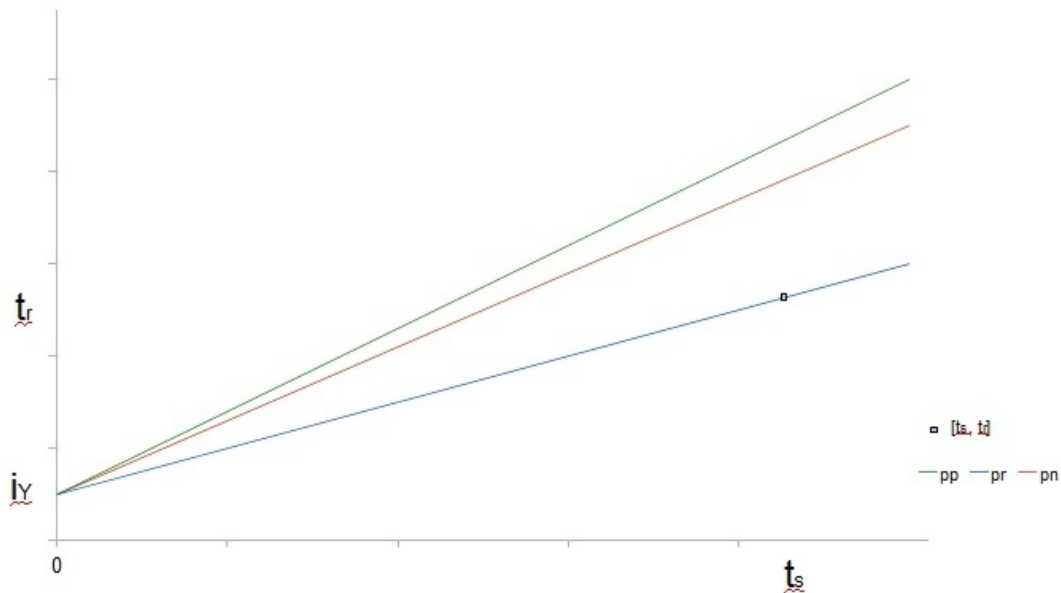
$$q = \frac{t_s - intersect_Y}{t_r} . \quad (62)$$

Získanú smernicu využijeme na úpravu odhadu časovej odchýlky pomocou váženého priemeru. Váha je určená počtom paketov, ktoré boli v TCP toku zaznamenané. Výsledná časová odchýlka sa vypočíta nasledovne

$$skew = \frac{s \cdot w_s + q \cdot w_q}{w_s + w_q}, \quad (63)$$

kde $w_s = N$ a $w_q = 1$.

Tento princíp je načrtnutý na nasledovnom obrázku (Obrázok 5.5). Označenie priamok je pp pre pôvodnú priamku, pr pre priamku určenú bodom $[t_s, t_r]$ a pn pre novú priamku, ktorej smernica je novým odhadom časovej odchýlky. Všetky tri priamky sa pretínajú v bode $intersect_y$. V reálnej situácii sú rozdiely smerníc týchto priamok oveľa menšie, tu sú použité veľké rozdiely pre názornosť obrázka. Skutočná úprava pôvodnej smernice je vďaka použitiu váženého priemeru minimálna, pretože pri veľkom počte paketov sa váha jediného paketu výrazne neprejaví.



Obrázok 5.5: Metóda aktualizácie odhadu časovej odchýlky pri príchode paketu váženým priemerom

6 Implementácia

Popísaná metóda Vzdialeného označenia fyzického zariadenia bola implementovaná ako skript v jazyku Python. Výsledná aplikácia funguje nasledovne. Po jej spustení sa vyhledá dostupné sieťové rozhranie, na ktorom je možné zachytávanie paketov. Pre každý prijatý paket je spustená obslužná procedúra, ktorá analyzuje jeho hlavičku. Ak sa jedná o paket protokolu TCP, ktorý obsahuje rozšírenie TCP Timestamp Option, spracuje sa v ňom obsiahnutá časová známka a čas prijatia tohto paketu. Tieto dve novo získané hodnoty sú priradené do množiny offsetov, ktorá prislúcha danému TCP toku. Tok je identifikovaný pomocou kombinácie MAC adresy, IP adresy a portu. Port je zohľadnený, aby bolo možné rozoznať aj jednotlivé zariadenia umiestnené za NATom. Po zachytení dostatočného množstva paketov je uskutočnený výpočet časovej odchýlky. Jednotlivé fázy a ich realizácia sú popísané v nasledovných podkapitolách.

6.1 Zachytávanie paketov

Na načúvanie na sieťovom rozhraní a zachytávanie prichádzajúcich paketov bol použitý modul pcap. Prvým krokom je nájdenie sieťového rozhrania, na ktorom je možné odchyťvanie paketov. Nasleduje získanie deskriptora pre zachytávanie paketov. Po jeho získaní už môže začať nekonečný cyklus, v ktorom sa spracúvajú zachytené pakety volaním obslužnej funkcie pre každý paket.

6.2 Analyzovanie hlavičiek a získanie časových známok

V obslužnej funkcii nastáva rozloženie hlavičky paketu, ktoré je realizované využitím funkcií modulu dpkt. Najskôr sa získa ethernetová hlavička a určí sa typ vnorenej hlavičky. Ak je to hlavička protokolu IP, pokračuje sa v spracovaní. Nasleduje kontrola, či hlavička vnorená v IP hlavičke je protokolu TCP. Po úspešnom splnení podmienky je vykonaná analýza rozšírení TCP Options. Ak paket obsahuje TCP Timestamp Option, zaznamená sa časová známka spolu s časom prijatia paketu k príslušnému toku pre neskorší výpočet časovej odchýlky alebo sa priamo aktualizuje jej hodnota. Príslušnosť k toku je daná údajmi z hlavičiek, konkrétne ethernetovou MAC adresou, IP adresou a TCP portom.

6.3 Výpočty časovej odchýlky

Vyhodnotenie zaznamenaných dát spočíva vo výpočte smernice regresnej priamky, ktorá udáva frekvenciu timestamp clock potrebnú pre úpravu údajov pred ďalším výpočtom. Po ich úprave nasleduje opäť výpočet smernice regresnej priamky, ktorá už tentoraz udáva časovú odchýlku zariadenia. Pre oba výpočty je použitá vždy rovnaká funkcia. Výpočet bol uskutočnený pomocou niekoľkých modulov.

Prvým modulom je numpy, ktorý obsahuje implementáciu Metódy najmenších štvorcov. Druhý modul, ktorý bol využitý pre lineárnu regresiu, je scipy. Ten obsahuje odlišnú implementáciu Metódy najmenších štvorcov. Taktiež sa v tomto module nachádza implementácia Metódy ortogónálnej regresie. K zahrnutiu viacerých metód bolo prístupné pre otestovanie nimi dosahovaných výsledkov. Výsledky ich porovnania sú obsahom nasledovnej kapitoly (7.1).

6.4 Aktualizácia hodnoty časovej odchýlky

Modul rlspy obsahuje implementáciu rekurzívnej metódy najmenších štvorcov, ktorá bola použitá pre aktualizáciu hodnoty časovej odchýlky. Ako prvý krok sa vytvorí model s počiatočnými podmienkami. Príchod paketu obsahujúceho časovú známku spustí aktualizáciu odhadu hodnoty časovej odchýlky.

Druhý spôsob výpočtu bol navrhnutý ako pokus o alternatívu rekurzívnej metódy. Postup aktualizácie hodnoty časovej odchýlky váženým priemerom pri príchode každého nového paketu bol implementovaný nasledovne. O každom toku sú udržiavané informácie o jeho veľkosti (počte paketov), časovej odchýlke, bode priesečníka regresnej priamky s osou Y, prvej časovej známke t_{s0} a čase prvého prijatia paketu t_{r0} . Po prijatí paketu sa určí jeho príslušný tok a upraví sa hodnoty t_r a t_r spôsobom, ktorý bol uvedený pri popise tejto metódy. Následne sa vypočíta smernica priamky, ktorá prechádza bodom určeným časovou známkou v pakete t_s a časom prijatia paketu t_r a bodom priesečníka $intersect_Y$ regresnej priamky s osou Y. Zachytený paket, ktorý prislúcha k danému toku, ovplyvní hodnotu pôvodnej časovej odchýlky s , ktorej prislúcha váha w_s . Vplyv tohto paketu je daný jeho váhou w_q .

7 Testovanie a dosiahnuté výsledky

Vykonanie testov na implementovanej aplikácii pozostávalo z niekoľkých fáz. Najskôr bolo potrebné porovnať výsledky jednotlivých regresných metód a na ich základe vybrať tú najvhodnejšiu pre použitie v ostatných testoch. Druhou fázou bolo overenie správnosti implementácie, teda či je aplikácia schopná rozlíšiť zariadenia podľa časových známok. Návrh tretej sady testov sa zamerával na odľahčenie metódy možnosťou vynechania niektorých paketov z výpočtov. Ďalšia množina testov mala za úlohu overiť, či je možné aktualizovať hodnotu časovej odchýlky priebežne pri zachytení jedného paketu. Dosiahnuté výsledky sú prezentované a diskutované v nasledovných podkapitolách.

7.1 Porovnanie regresných metód

Skript umožňuje využiť pre výpočet regresnej priamky tri rôzne metódy, preto je prvým testom porovnanie nimi dosahovaných výsledkov.

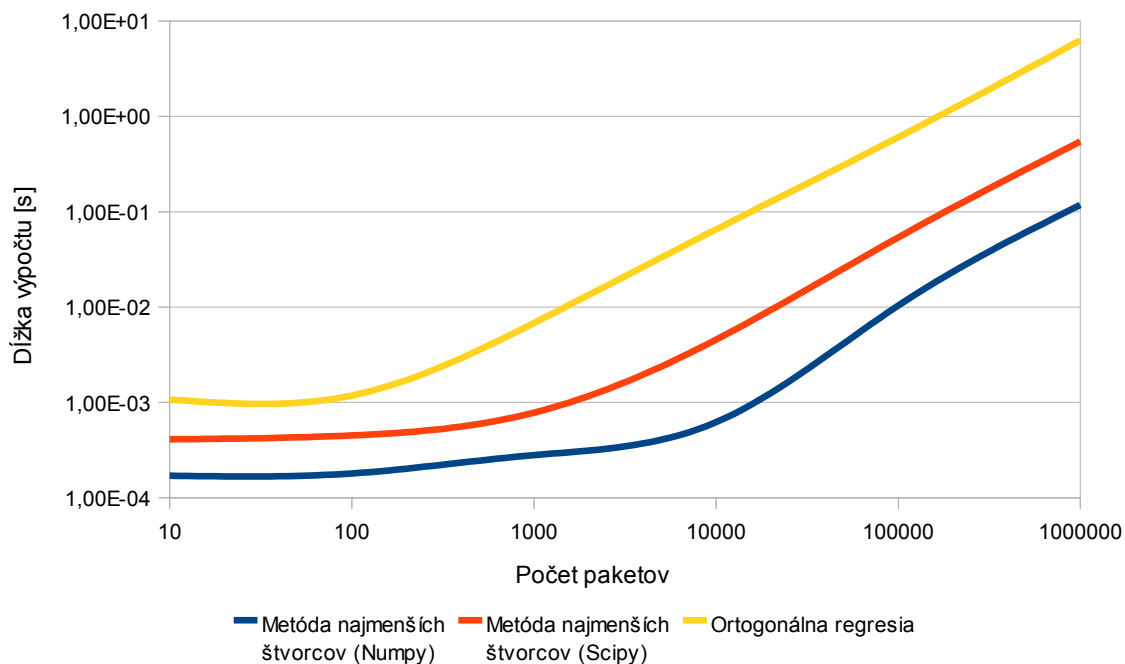
Čo sa týka získaných hodnôt časovej odchýlky, metódy sa od seba navzájom líšia len o veľmi malé hodnoty, ktoré závisia na počte paketov. Odlišnosti výsledkov sú uvedené v nasledovnej tabuľke (Tabuľka 1), kde sú metódy označené nasledovným spôsobom. Metóda najmenších štvorcov z modulu Numpy má označenie 1., Metóda najmenších štvorcov z modulu Scipy má označenie 2. a Ortogonálna regresná metóda má označenie 3..

Počet paketov	Rozdiely medzi metódami v časových odchýlkach [ppm]		
	1. - 2.	2. - 3.	1. - 3.
10	3,93E+00	2,68E-03	3,94E+00
100	2,75E-01	1,89E-04	2,75E-01
1 000	2,33E-02	1,71E-03	2,50E-02
10 000	1,53E-03	-4,16E-04	1,12E-03
100 000	3,05E-06	6,66E-08	3,12E-06
1 000 000	1,75E-08	-2,34E-08	-5,90E-09

Tabuľka 1: Rozdiely vo vypočítaných hodnotách časovej odchýlky medzi jednotlivými metódami

Rozdiely vo výsledkoch sa zmenšujú so zvyšujúcim sa počtom paketov. Metóda 1. sa od ostatných dvoch mierne líši. Rozdiely, ktoré by spôsobili problémy s presnosťou metódy sa však objavujú len pri veľmi malom počte paketov. Metódy 2. a 3. dosahujú porovnateľné výsledky pre všetky merané prípady. Pri počte 10 000 paketov už sú rozdiely medzi všetkými metódami zanedbateľné. Takéto drobné rozdiely nehrajú pri výbere algoritmu pre ďalšie testovanie žiadnu rolu, preto nasleduje test zameraný na rýchlosť výpočtu.

V nasledovnom grafe (Graf 7.1) je zobrazená závislosť doby trvania výpočtu časovej odchýlky jednotlivými metódami na veľkosti zaznamenaného toku (počte paketov).



Graf 7.1: Závislosť dĺžky výpočtu jednotlivých metód na počte paketov

Z grafu je zrejmé, že z testovaných metód je najrýchlejšia implementácia Metódy najmenších štvorcov z modulu Numpy, a preto bude v ďalších testoch použitá práve táto metóda.

7.2 Rozlíšenie zariadení

Táto podkapitola je zameraná na overenie samotnej podstaty implementovanej metódy, teda či je pomocou nej možné identifikovať zariadenie v rámci počítačovej siete. Sada testov pozostávala z opakovaného zachytenia TCP tokov z toho istého zariadenia, pričom jednotlivé toky boli zachytené s odstupom niekoľkých hodín, až niekoľkých dní. Toky s vysokým počtom paketov boli prenosom súborov s veľkosťou niekoľkých gigabajtov cez FTP. V nasledovnej tabuľke (Tabuľka 2) je prehľad získaných informácií o zachytených tokoch kategorizovaný podľa zariadení, ktorým bol daný tok priradený. Každé meranie obsahuje informácie o počte paketov, z ktorého pozostával daný tok a vypočítanej frekvencie timestamp clock a časovej odchýlky.

Všetky merania poskytli presný odhad frekvencie timestamp clock. Z tabuľky je zrejmé, že pomocou implementovanej metódy je možné s určitou mierou pravdepodobnosti identifikovať zariadenie pomocou vypočítanej hodnoty časovej odchýlky. Najmenší rozdiel časových odchýliek jedného zariadenia je 0,85 ppm (zariadenie 6), najväčší 4,06 ppm (zariadenie 2).

Zariadenie	Meranie	Počet paketov	Frekvencia [Hz]	Časová odchýlka [ppm]
1	1.	882 515	100	38,6669
	2.	701 166		36,6956
	3.	790 173		35,7823
	4.	818 599		36,5021
2	1.	1 525 624	1000	44,2241
	2.	2 137 141		44,7453
	3.	815 415		43,8820
	4.	1 627 954		47,9395
3	1.	869 702	1000	13,2529
	2.	883 969		14,6440
	3.	932 454		14,1776
4	1.	1 202 174	250	30,4378
	2.	1 089 902		31,0771
	3.	1 238 003		31,6368
5	1.	874 704	1000	7,6355
	2.	858 317		6,2031
	3.	887 236		4,3915
6	1.	784 422	1000	17,7428
	2.	691 065		18,5923
	3.	659 954		18,4665
7	1.	2 655 923	250	21,8796
	2.	2 782 219		22,9392
	3.	2 366 215		23,2215

Tabuľka 2: Výsledky meraní časovej odchýlky pre rôzne zariadenia

Rozdiely v hodnotách umožňujú odlišiť jednotlivé zariadenia. Samozrejme však musíme vziať do úvahy fakt, že sa v reálnej situácii môžu vyskytnúť zariadenia, ktorých časová odchýlka je veľmi podobná. Najmenší rozdiel časových odchýlok dvoch rôznych zariadení je 3,29 ppm. Táto hodnota je menšia ako najväčší rozdiel časových odchýlok jedného zariadenia. Tento fakt je dôvodom, prečo nemôžu byť získané výsledky brané ako jednoznačná identifikácia, ale len ako miera pravdepodobnosti, že sa jedná o to isté zariadenie. Aj táto miera pravdepodobnosti je však pomerne dostatočná pre využitie metódy na účely, ktoré boli uvedené v podkapitole 4.3.2.

7.3 Vplyv vynechávania paketov

Táto skupina testov bola zaradená za účelom preskúmania možností metódy z pohľadu budúceho nasadenia do skutočnej prevádzky, napríklad v routeri alebo inom sieťovom uzle. Vzhľadom na zvyšujúcu sa rýchlosť prenosov a objemu prenesených dát v súčasných sieťach vzrastá počet paketov, ktoré musí takéto zariadenie s obmedzenými zdrojmi spracovať. Skúmal sa teda vplyv

vynechávania paketov na výsledky metódy. Metodicky je možné vynechávať pakety rôznymi spôsobmi. Z nich bol zvolený spôsob, ktorým je použitie každého n-tého paketu.

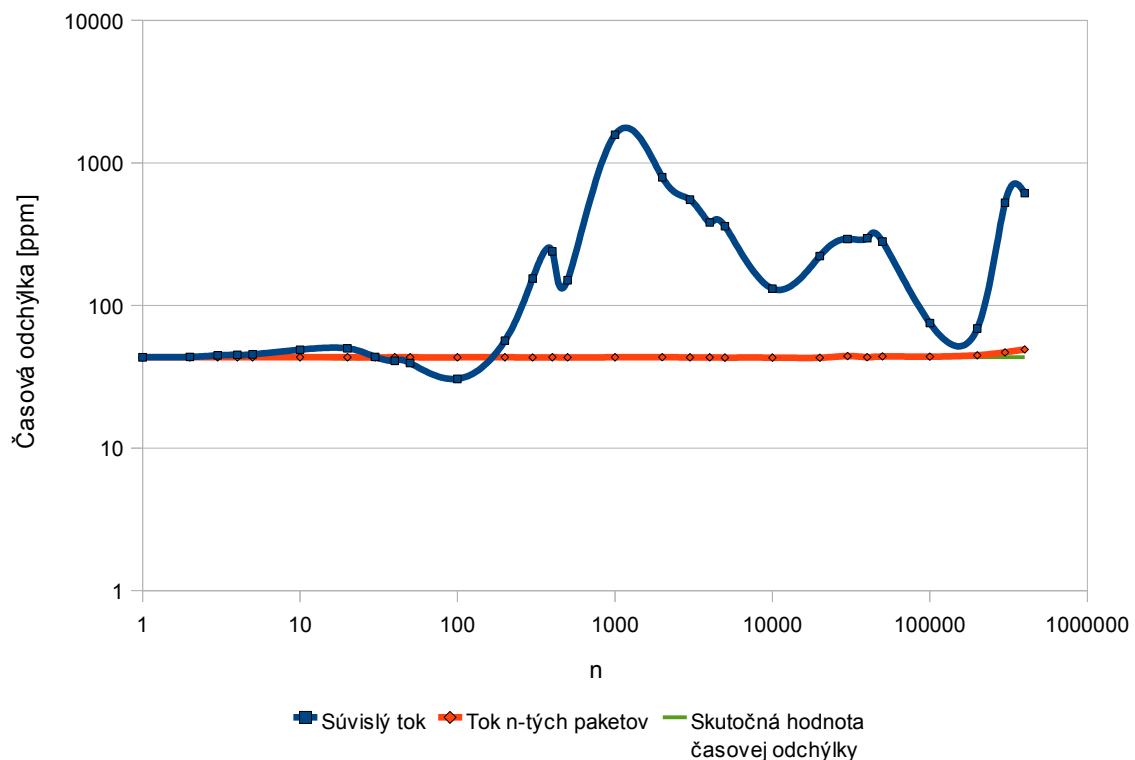
Výhodou použitia každého n-tého paketu je možnosť určiť časovú odchýlku z výrazne menšieho počtu paketov, pričom rovnomerne reprezentujú hodnoty z celého toku. Pritom účinne zabraňujú prejavaniu lokálnych výkyvov, napríklad zdržaniu na sieti. V nasledovnej tabuľke (Tabuľka 3) sú takto vypočítané údaje porovnané s výsledkami, ktoré boli dosiahnuté výpočtom zo súvislej časti toku s rovnakým počtom paketov.

n	Počet paketov	Tok n-tých paketov		Súvislý tok	
		Frekvencia [Hz]	Časová odchýlka [ppm]	Frekvencia [Hz]	Časová odchýlka [ppm]
1	1495903	1000	43,3279	1000	43,3279
2	747952	1000	43,3286	1000	43,5959
3	498635	1000	43,3019	1000	44,6970
4	373976	1000	43,3292	1000	45,0318
5	299181	1000	43,3159	1000	45,4193
10	149591	1000	43,3162	1000	48,9963
20	74796	1000	43,3200	1000	50,0290
30	49864	1000	43,2570	1000	43,5381
40	37398	1000	43,3367	1000	40,9762
50	29919	1000	43,3103	1000	39,4518
100	14960	1000	43,2988	1000	30,5630
200	7480	1000	43,3298	1000	56,5859
300	4987	1000	43,2067	1000	154,4881
400	3740	1000	43,3048	1000	239,5600
500	2992	1000	43,2734	1000	150,2821
1000	1496	1000	43,3145	998	1576,5779
2000	748	1000	43,3546	999	793,4833
3000	499	1000	43,2589	999	552,2557
4000	374	1000	43,2656	1000	381,9517
5000	300	1000	43,1323	1000	359,6119
10000	150	1000	43,1092	1000	131,3578
20000	75	1000	43,0305	1000	222,2649
30000	50	1000	44,1992	1000	292,8342
40000	38	1000	43,4099	1000	296,4150
50000	30	1000	43,9154	1000	280,9251
100000	15	1000	43,8194	1000	75,3499
200000	8	1000	44,8030	1000	68,9958
300000	5	1000	46,9724	1001	524,8418
400000	4	1000	49,1645	1001	614,3079
500000	3	1000	47,8768	973	27289,6211

Tabuľka 3: Výsledky frekvencie a časovej odchýlky pri využití každého n-tého paketu a súvislého toku s rovnakým počtom paketov pri toku s veľkým počtom paketov

Tento test bol uskutočnený na toku, ktorý pozostával z približne 1,5 milióna paketov. Z uvedených hodnôt je evidentné, že použitím tohto postupu dosiahneme správne výsledky (s výnimkou najvyšších hodnôt n). Pre výpočet frekvencie timestamp clock je tento postup

jednoznačne využitelný. Vďaka zaokrúhľovaniu na celé číslo je takto vypočítaná frekvencia presne stanovená pre každé n . Aj odlišnosť vypočítanej odchýlky pre všetky hodnoty $n \leq 100\,000$ od skutočnej odchýlky ($n = 1$) je malá a aj pri využití len jednej stotisíciny paketov z tohto toku získavame dostatočne presnú hodnotu časovej odchýlky. Naopak použitím súvislej časti toku dosahujeme značne nesprávne výsledky, ktoré majú akceptovateľnú presnosť len pre $n = 1$ a $n = 2$. Získané výsledky sú prehľadne zobrazené v nasledovnom grafe (Graf 7.2).



Graf 7.2: Závislosť časovej odchýlky na n

Z grafu vidíme, že odhad časovej odchýlky pomocou toku n -tých paketov je takmer konštantný. Naopak časová odchýlka určená zo súvislej časti toku má značne náhodný priebeh a dosahuje dobré výsledky len pri malej hodnote n , teda pri využití veľkej časti paketov pôvodného toku. Vynechávanie paketov sa teda ukazuje ako správny prístup k danej problematike.

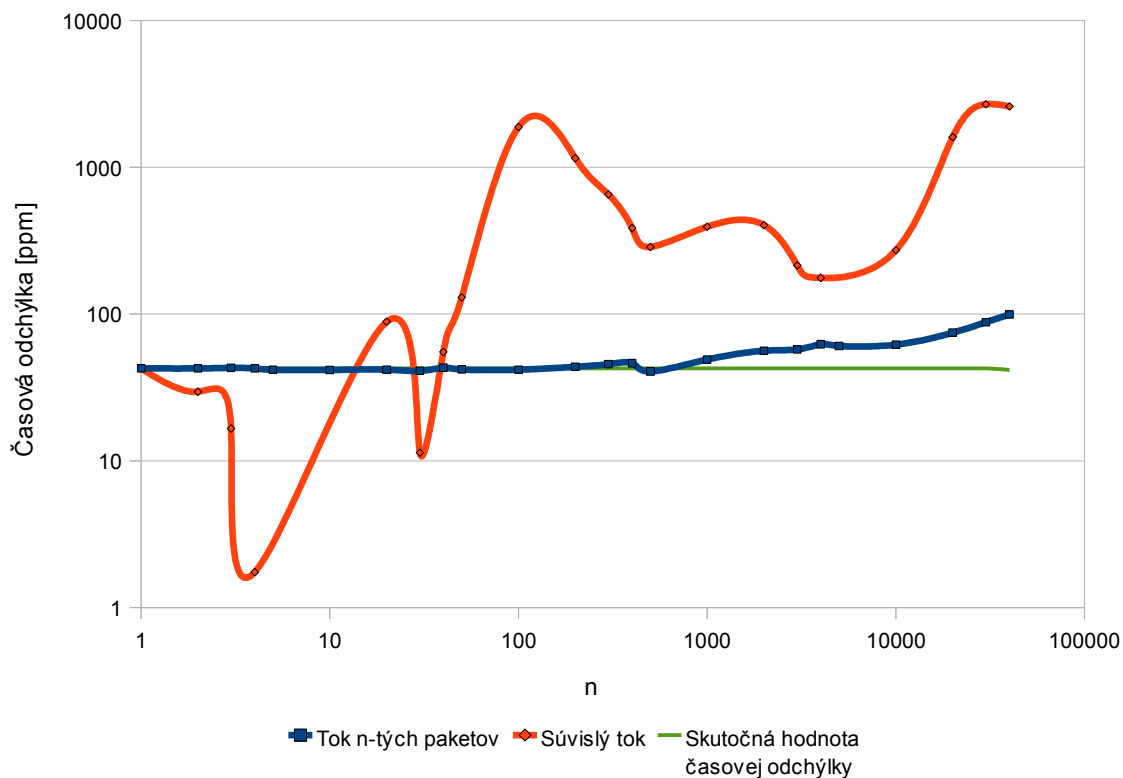
Rovnaké, prípadne veľmi podobné výsledky boli dosiahnuté aj pri ďalších testoch s obdobne veľkými tokmi. Nasledovný test s rovnakým postupom bol vykonaný na toku s oveľa menším množstvom paketov, pozostával zo 125 tisíc paketov. Cieľom testu bolo overiť použiteľnosť postupu aj pre malé toky a jeho výsledky sú zobrazené v nasledovnej tabuľke (Tabuľka 4).

n	Počet paketov	Tok n-tých paketov		Súvislý tok	
		Frekvencia [Hz]	Časová odchýlka [ppm]	Frekvencia [Hz]	Časová odchýlka [ppm]
1	125098	1000	42,6188	1000	42,6188
2	62549	1000	42,5958	1000	29,5825
3	41700	1000	43,0194	1000	16,5961
4	31275	1000	42,6522	1000	1,7438
5	25020	1000	41,7073	1000	-11,5859
10	12510	1000	41,6225	1000	-120,9461
20	6255	1000	41,7709	1000	88,6744
30	4170	1000	41,0281	1000	11,3653
40	3128	1000	43,0841	1000	55,2192
50	2502	1000	41,9217	1000	129,6935
100	1251	1000	41,8024	998	1 883,7721
200	626	1000	43,7202	999	1 154,3108
300	417	1000	45,5810	999	652,6023
400	313	1000	46,2361	1000	384,9097
500	251	1000	40,6563	1000	286,5733
1000	126	1000	48,9918	1000	393,3316
2000	63	1000	56,2009	1000	403,7212
3000	42	1000	57,4255	1000	214,4664
4000	32	1000	62,2607	1000	176,4360
5000	26	1000	60,6584	1000	-1,5594
10000	13	1000	61,8616	1000	272,2372
20000	7	1000	74,6541	998	1 606,7676
30000	5	1000	88,0264	997	2 688,1514
40000	4	1000	99,3041	997	2 594,5993
50000	3	1000	57,8203	1239	-238 623,3274
100000	2	1000	57,7500	1273	-272 543,6893

Tabuľka 4: Výsledky frekvencie a časovej odchýlky pri využití každého n-tého paketu a súvislého toku s rovnakým počtom paketov pri toku s malým počtom paketov

Výsledky testu opäť ukazujú, že určenie frekvencie timestamp clock je nenáročná úloha, ktorá si vyžaduje malé množstvo údajov. Čo sa týka časovej odchýlky, dosiahnuté hodnoty nie sú také stabilné, ako tomu bolo pri predchádzajúcom teste s veľkým tokom. Prijateľné výsledky sú dosahované pre hodnoty $n \leq 100$. Ukazuje sa teda, že použiteľnosť tohto postupu závisí na veľkosti toku, ktorého časovú odchýlku skúmame. Výsledky s využitím súvislých tokov sú opäť nepoužiteľné.

Získané výsledky sú vykreslené v nasledovnom grafe (Graf 7.5). Z grafu je evidentné, že pri tokoch s malým počtom paketov nie je možné použiť výpočet časovej odchýlky zo súvislej časti toku. Aj možnosti využitia toku n-tých paketov je v tejto situácii limitovaná. Pre hodnoty $n > 100$ sú už výsledky časovej odchýlky nepresné aj pri tomto postupe. Aj napriek tomu je však schopnosť určiť časovú odchýlku zo stotiny paketov veľkým úspechom.



Graf 7.3: Závislosť časovej odchýlky na n

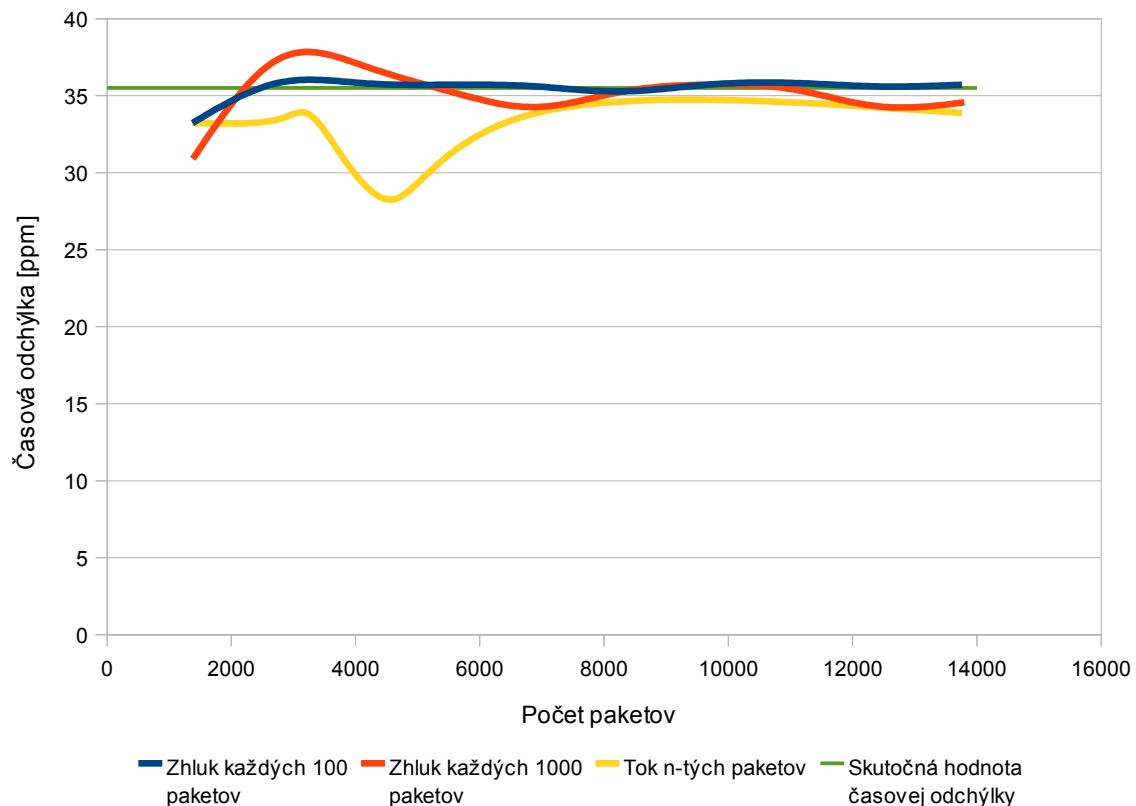
Z uvedených výsledkov je možné vyvodit' niekoľko záverov. Pakety je možné vynechávať najmä pri výpočte frekvencie, ktorú je možné spoľahlivo určiť už z malého množstva zachytených údajov. Je to spôsobené faktom, že frekvencia je celé číslo a pri zaokrúhľovaní získanej hodnoty sa stratia drobné nepresnosti. Pri vynechávaní paketov za účelom urýchlenia výpočtu časovej odchýlky je vhodné prispôbiť voľbu hodnoty n predpokladanej veľkosti toku.

Do úvahy ešte pripadá možnosť nevyužívať len každý n -tý paket, ale rovnomerne rozložené zhluky niekoľkých paketov. Výsledky tohto prístupu sú uvedené v nasledovnej tabuľke (Tabuľka 5).

Zhluk každých 100 paketov			Zhluk každých 1000 paketov			Tok n -tých paketov	
Veľkosť zhlukov	Počet paketov	Časová odchýlka [ppm]	Veľkosť zhlukov	Počet paketov	Časová odchýlka [ppm]	Počet paketov	Časová odchýlka [ppm]
1	1376	33,23	10	1380	30,89	1376	33,23
2	2752	35,83	20	2760	37,39	2752	33,46
3	4128	35,81	30	4140	36,93	3440	33,07
4	5504	35,71	40	5520	35,28	4587	28,26
5	6880	35,61	50	6900	34,27	6880	33,85
6	8256	35,28	60	8280	35,26	-	-
7	9632	35,71	70	9660	35,72	-	-
8	11008	35,84	80	11040	35,42	-	-
9	12384	35,61	90	12420	34,31	-	-
10	13760	35,71	100	13800	34,57	13760	33,89

Tabuľka 5: Výsledky časovej odchýlky pri použití zhlukov paketov a toku n -tých paketov

Výsledky sú usporiadané s ohľadom na počet paketov, ktoré boli využité pre výpočet časovej odchýlky pri jednotlivých meraniach. Na nasledovnom grafe (Graf 7.4) sú zobrazené odhady časovej odchýlky v závislosti na počte paketov.



Graf 7.4: Závislosť časovej odchýlky na počte paketov pri použití zhlukov paketov v porovnaní s využitím toku n-tých paketov v toku s malým počtom paketov

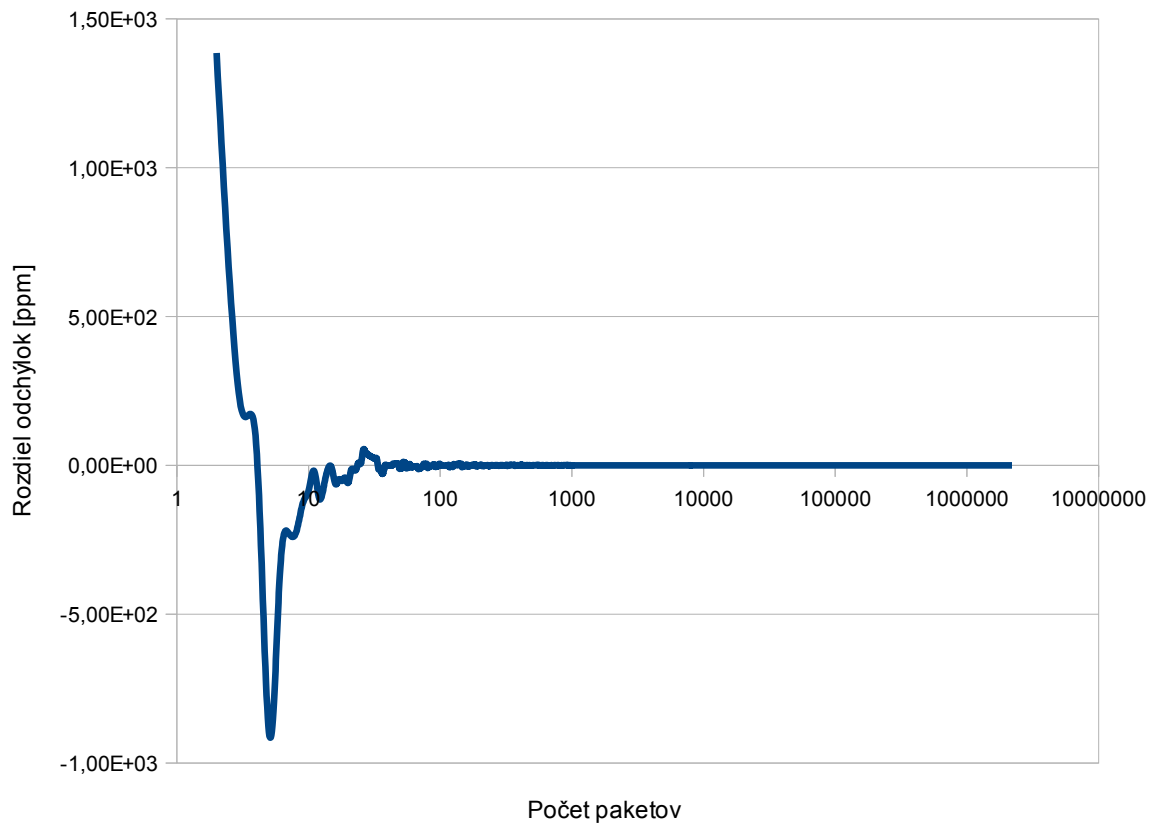
Z priebehu funkcií je zjavné, že aj keď postup využívajúci tok n-tých paketov dosahuje potrebnú presnosť až pri hodnotách okolo 8 000 paketov, využitím malých zhlukov paketov opakujúcich sa pravidelne každých 100 paketov môžeme dosiahnuť potrebnú presnosť už pri polovičnom počte paketov.

V tejto podkapitole boli predstavené dva spôsoby vynechávania paketov, vďaka ktorým je možné získať pomerne presnú hodnotu časovej odchýlky aj z malého množstva dát. Pre menšie toky je odporúčané použiť zhluky paketov, pri väčších tokoch zase využiť toky n-tých paketov.

7.4 Priebežná aktualizácia časovej odchýlky

Využitím priebežnej aktualizácie časovej odchýlky môže byť jej hodnota neustále k dispozícii bez potreby ukladania si veľkého množstva dát a náročných výpočtov. Táto možnosť je využiteľná napríklad pre efektívnu detekciu zariadenia, ktoré sa nachádza v zozname blokovaných zariadení.

Nasledovný graf (Graf 7.5) predstavuje závislosť rozdielu časovej odchýlky získanej pomocou rekurzívnej metódy najmenších štvorcov a časovej odchýlky vypočítanej z dát priamo metódou najmenších štvorcov od počtu paketov.



Graf 7.5: Rozdiel časových odchýlok vypočítaných pomocou rekurzívnej metódy najmenších štvorcov a obyčajnej metódy najmenších štvorcov

Z grafu je jasné, že už pri počte približne 100 paketov je rozdiel hodnôt časových odchýlok dostatočne malý. Tieto výsledky potvrdzujú možnosť využitia rekurzívnej metódy najmenších štvorcov na priebežnú aktualizáciu hodnoty časovej odchýlky. Výsledky predchádzajúcich testov ukázali, že získanie presnej frekvencie timestamp clock sledovaného zariadenia je pomerne jednoduchá operácia, ktorá vyžaduje malé množstvo dát. Aj táto skutočnosť prispieva k použiteľnosti priebežnej aktualizácie časovej odchýlky už pri malom množstve zachytených dát.

Pre túto prácu navrhnutá alternatívna metóda priebežnej aktualizácie časovej odchýlky použitím váženého priemeru nedosiahla vo vykonaných testoch potrebné parametre a preto nie je vhodná pre potreby takto presných výpočtov.

8 Záver

Táto diplomová práca sumarizuje poznatky o meraní času v moderných počítačových sieťach. Objasňuje spôsob fungovania hodín v bežnej elektronike a ich vlastnosti. Taktiež popisuje dôvod vzniku a vlastnosti časových odchýlok, tzv. clock skew. Tie je možné detekovať zo zachytenej komunikácie sledovaného zariadenia. Zachytené dáta však musia obsahovať pakety nesúce časové známky, čo je splnené len pri paketoch protokolov TCP a ICMP a len za určitých podmienok. Z týchto údajov je možné pomocou popísanej metódy identifikovať zariadenia komunikujúce v sieti Internet. Slúžia k tomu metódy lineárnej regresie, pomocou ktorých sú získané dáta preložené regresnou priamkou.

Výsledkom práce je aplikácia, teda presnejšie skript, ktorý implementuje popísanú metódu vzdialeného označenia fyzických zariadení. Boli v nej využité dva prístupy k výpočtu hodnoty časovej odchýlky. Jednou možnosťou bolo jej určenie po zachytení celého toku, prípadne jeho dostatočne veľkej časti. Druhá možnosť spočívala v priebežnej aktualizácii tejto hodnoty pre zaistenie jej neustálej dostupnosti.

Prínosom práce sú vykonané testy, ktorých výsledky sú prehľadne interpretované v grafoch a tabuľkách. Významnú skupinu testov tvoria testy vplyvu vynechávania paketov, ktoré umožňujú zníženie množstva potrebných dát a výpočtov pri určovaní hodnoty časovej odchýlky. Na ich základe sú stanovené závery a odporúčania pre budúce použitie tejto metódy.

8.1 Pokračovanie práce

Možným pokračovaním tejto práce je implementácia popísanej metódy do zariadenia s obmedzenými zdrojmi, napríklad FPGA. Druhou možnosťou je optimalizácia regresných metód z hľadiska pamäťovej a časovej náročnosti. Ďalšou možnosťou je navrhnúť bezpečnostný systém, ktorý by využíval danú metódu.

Literatúra

- [1] Arkin O., Yarochkin F., „*XProbe2 – A ‘Fuzzy’ Approach to Remote Active Operating System Fingerprinting*“, 2002
- [2] Boggs P. T., Rogers J. E.. „*Orthogonal Distance Regression*“. In P.J. Brown and Wayne A. Fuller, editor, *Contemporary Mathematics*. American Mathematical Society, Providence, Rhode Island, 1990.
- [3] Dana P., Penrod B., „*The Role of GPS in Precise Time and Frequency Dissemination*“, GPS World, 1990
- [4] DARPA, „*Transmission Control Protocol*“, RFC 793, 1981
- [5] Dreher A., Mohl D., „*Precision Clock Synchronization*“, *White Paper*, The Standard IEEE 1588
- [6] Ettercap homepage, <http://ettercap.sourceforge.net>, 2012
- [7] Fajmon B., Růžicková I., „*Matematika 3*“, Ústav matematiky FEKT VUT Brno, 2005
- [8] *GMT and Other Time Systems Explained*
<http://www.timeanddate.com/time/gmt-utc-time.html> [cit. 30.12. 2011]
- [9] Gross R., „*The excitation of the Chandler Wobble*“, Jet Propulsion Laboratory, NASA, 1999
- [10] Hayes, Monson H. „9.4: Recursive Least Squares“. *Statistical Digital Signal Processing and Modeling*. Wiley. p. 541. ISBN 0-471-59431-8., 1996
- [11] Jacobson V., Braden R., Borman D., „*TCP Extensions for High Performance*“, RFC 1323, 1992
- [12] Kohno T., Broido A., Claffy K. C., „*Remote Physical Device Fingerprinting*“. IEEE Trans. Dependable Secur. Comput., 2005, 93-108.
- [13] Lombardi A., Nelson L., Novick A., Zhang V., „*Time and Frequency Measurements Using the Global Positioning System*“, NIST, 2001
- [14] Markovsky I, Van Huffel S., „*Overview of total least squares methods*“. Signal Processing, vol. 87, pp. 2283-2302, 2007
- [15] Mills D., Delaware U., Martin J., Burbank J., Kasch W., „*Network Time Protocol Version 4: Protocol and Algorithms Specification*“, RFC 5905, 2010
- [16] Mills D., „*The Network Time Protocol (NTP) Distribution*“, 2011
<http://www.eecis.udel.edu/~mills/ntp/html/index.html>
- [17] Mills D., „*Simple Network Time Protocol (SNTP) Version 4 for IPv4, IPv6 and OSI*“, RFC 4330, 2006
- [18] Mills D., „*Network Time Protocol (Version 3) Specification, Implementation and Analysis*“, RFC 1305, 1992

- [19] Moon S. B., Skelly P., Towsley D., „*Estimation and Removal of Clock Skew From Network Delay Measurements*” Proc. INFOCOM Conf., 1999
- [20] *NIST Automated Computer Time Service (ACTS)*, 2010
<http://www.nist.gov/pml/div688/grp40/acts.cfm> [cit. 30.12. 2011]
- [21] Nmap Security Scanner, <http://nmap.org>, 2012
- [22] OS Fingerprinting through RTOs, <http://www.planb-security.net/wp/ring.html>, 2012
- [23] Paxson V., „*On Calibrating Measurements of Packet Transit Times*” Proc. SIGMETRICS Conf., 1998
- [24] Postel J., „*Daytime Protocol*“, RFC 867, 1983
- [25] Postel J., Harrenstein K., „*Time Protocol*“, RFC 868, 1983
- [26] Postel J., „*Internet Control Message Protocol*“, RFC 792, 1981
- [27] Project details for p0f, <http://freecode.com/projects/p0f>, 2012
- [28] Taylor B. N., Thompson A., „*The International System of Units (SI)*“, NIST Special Publication, p. 19, 2008
- [29] Veitch D., Babu S., Pásztor A., „*Robust Synchronization of Software Clocks Across the Internet*” Proc. Fourth ACM SIGCOMM Conf. Internet Measurement, 2004
- [30] Weibel H., „*The Second Edition of the High Precision Clock Synchronization Protocol*“, Technology Update on IEEE 1588, 2008
- [31] Wikipedia Contributors, „*Coordinated Universal Time*“, Wikipedia, The Free Encyclopedia, 2011
http://en.wikipedia.org/w/index.php?title=Coordinated_Universal_Time&oldid=467031579
- [32] Wikipedia contributors. „*TCP/IP stack fingerprinting*“, Wikipedia, The Free Encyclopedia, 2012,
http://en.wikipedia.org/w/index.php?title=TCP/IP_stack_fingerprinting&oldid=462420582

Zoznam príloh

Príloha A – Zhrnutie algoritmu rekurzívnej metódy najmenších štvorcov

Príloha B – Obsah CD

Príloha C – CD so zdrojovými kódmi

Príloha A – Zhrnutie algoritmu rekurzívnej metódy najmenších štvorcov

Parameters :

- p = Filter order
- α = Exponential weighting factor
- δ = Value used to initialize $P(0)$

Initialization :

$$w_0 = 0$$
$$P(0) = \delta^{-1} I$$

Computation :

For $n = 1, 2, \dots$ *compute*

$$z(n) = P(n-1)x^*(n)$$

$$g(n) = \frac{z(n)}{\lambda + x^T(n)z(n)}$$

$$\alpha(n) = d(n) - w_{n-1}^T x(n)$$

$$w_n = w_{(n-1)} + \alpha(n)g(n)$$

$$P(n) = \frac{P(n-1) - g(n)z^H(n)}{\lambda}$$

Príloha B – Obsah CD

/script.py	implementácia metódy
/README	návod na použitie skriptu
/xkrbam00.pdf	diplomová práca