isa-top: Aplikace pro získání statistik o síťovém provozu

Dokumentace projektu do předmětu ISA, FIT VUT 2024

Autor: Michal Pavlíček (xpavlim00)
Email: xpavlim00@stud.fit.vutbr.cz
Vedoucí: Ing. Matěj Grégr, Ph.D.

Datum: 17. listopadu 2024

Obsah

Kapitola 1

Úvod

Tato dokumentace popisuje implementaci a teorii nutnou pro pochopení fungování nástroje isa-top. Tento program funguje na rozhraní příkazového řádku a zpracovává a zobrazuje v reálném čase komiunikaci na dané síti mezi jednotlivými hosty.

Program je inspirován nástrojem iftop, oproti kterému však nabízí jednodušší a přehlednější rozhraní zaměřené na základní statistiky síťového provozu.

1.1 Uvedení do problému

Získávání statistik o síťovém provozu je klíčovým nástrojem pro správu a optimalizaci počítačových sítí. Administrátoři potřebují mít přehled o tom, jak je síť využívána, kteří uživatelé nebo služby spotřebovávají nejvíce síťových prostředků a zda síť funguje podle očekávání.

Monitorování síťového provozu v reálném čase umožňuje rychle odhalit různé problémy. Může se jednat o nadměrné zatížení sítě jednotlivými uživateli, výpadky spojení nebo neobvyklé vzory komunikace, které mohou indikovat bezpečnostní problémy. Například náhlý nárůst provozu může znamenat probíhající útok typu DoS (Denial of Service) nebo přítomnost škodlivého softwaru.

1.2 Cíle projektu

Hlavní cíle projektu isa-top jsou následující:

- Vytvořit konzolovou aplikaci zachycující provoz na síti
- Podporovat IPv4 a IPv6 komunikaci
- Srozumitelně zobrazit rychlosti mezi jednotlivými hosty uživateli
- Důsledně aplikaci otestovat

Kapitola 2

Teorie

2.1 Síťová komunikace

Síťová komunikace je organizována do vrstev podle protokolu TCP/IP. Program isa-top na vrstvě síťového rozhraní zachycuje pouze rozhraní Ethernet, na síťové vrstvě poté pouze protokoly IPv4 a IPv6 a podle informací v jejich hlavičce počítá informace o přenosech na síti.

Pro získání velikosti přenesených dat program využívá informace z hlavičky zachyceného paketu poskytnuté knihovnou libpcap (`pcap_pkthdr->len`), která obsahuje skutečnou délku paketu na síti včetně všech hlaviček. Tato hodnota je klíčová pro přesný výpočet přenosových rychlostí, protože zahrnuje kompletní velikost síťové komunikace včetně režie protokolů.

2.1.1 IPv4 a IPv6

Protokoly IPv4 a IPv6 oba v rámci TCP/IP slouží především k adresaci a jednoznačně identifikují zařízení v síti. IPv4 k tomuto účelu používá 32 bitové adresy a IPv6 128 bitové adresy. Dále tyto protokoly umožňují fragmentaci dat a nezávislost na fyzickém přenosu.

V programu isa-top jsou statistiky přenesených bitů a paketů ukládány na základě stejných IPv4 nebo IPv6 adres. Pokud nastane komunikace mezi dvěma adresami, které už se v minulosti objevily, je do této komunikace přidána nová statistika.

Směr komunikace je určen podle první zachycené komunikace. Přijaté a poslané statistiky se poté přidávají do statistik odpovídajícím způsobem. Pokud tedy první zachycený paket šel z adresy A na adresu B, a následně je zachycen paket z B do A, jsou data přijatá z této druhé komunikace přidána do statistik první komunikace jako odeslaná a naopak.

2.1.2 Transportní protokoly

Na transportní vrstvě, existující nad IPv4 a IPv6 protokoly, existují další protokoly, jakými jsou například TCP, UDP (oba využívají aplikační porty), ICMP a další.

Transportní vrstva zajišťuje identifikaci aplikací pomocí portů a komunikaci mezi aplikacemi na cílových zařízeních.

- TCP - zajišťuje spolehlivý přenos dat. Obsahuje čísla portů pro identifikaci služeb.
- UDP - jednoduchý a rychlý přenos dat, není spolehlivý. Také využívá porty pro rozlišení služeb.

- ICMP/ICMPv6 - protokoly pro řízení a signalizaci chyb v síti. Nepoužívají porty.

Program isa-top identifikuje číslo použitého protokolu, který je následně přeložen na název protokolu a vypsán uživateli na obrazovku.

2.1.3 Promiskuitní režim

V běžném režimu síťová karta přijímá pouze pakety určené pro její MAC adresu. V promiskuitním režimu jsou zachyceny všechny pakety procházející síťovým rozhraním, což umožňuje zachytávání paketů na celé síti.

2.1.4 Filtrace paketů

Knihovna libpcap poskytuje mechanismus BPF (Berkeley Packet Filter), který umožňuje efektivní filtrování paketů. Program isa-top využívá filtr "ip or ip6"pro zachycení pouze IPv4 a IPv6 provozu.

2.1.5 Zpracování hlaviček

Při zachycení paketu je nutné postupně analyzovat hlavičky jednotlivých protokolů:

- Ethernet hlavička - identifikace typu následujícího protokolu
- IP hlavička - získání zdrojové a cílové adresy
- Transportní hlavička - získání portů a typu protokolu

2.2 Vlákna

Pro zajištění spolehlivého běhu programu jsou použita vlákna.

Vlákna v počítači běží současně a sdílí paměťový prostor. To může vést k tzv. race condition, což zjednodušeně značí současný přístup k datům. Z toho důvodu je zapotřebí využít semaforů, které zajišťují bezkonfliktní průběh.

V programu isa-top běží současně dvě vlákna, jedno pro zachytávání paketů, druhé pro vypisování statistik uživateli.

2.3 Spojový seznam

Spojový seznam je dynamická datová struktura uchovávající datové položky stejného typu provázané vzájemnými ukazateli. Typem spojového seznamu je oboustranný spojový seznam, kde každá položka ukazuje nejen na svého následovníka, ale také na svého předchůdce.

Program isa-top využívá oboustranný spojový seznam společně s bublinkovým řazením pro uchovávání a správné řazení statistik. Každá statistika je na svém správném místě ve spojovém seznamu a pokud je zachycena nová komunikace, jejíž adresy odpovídají dané statistice, je tato statistika probublána výše na své nové místo.

Kapitola 3

Design a Implementace

3.1 Přehled Architektury

Program isa-top je strukturován do několika logických částí:

- Zachycování paketů (packet_handler.c)
- Výpočet a výpis statistik (stats.c)
- Řazení (linked_list.c)
- Parsování argumentů (cli.c)

3.2 Klíčové komponenty

3.2.1 Parsování argumentů

`cli.c` se stará o:

- Kontrolu a uchování argumentů z příkazové řádky
- Možnosti konfigurace

3.2.2 Zachycování paketů

`packet_handler.c` zařizuje:

- Odchyt paketů za pomocí knihovny libpcap
- Identifikace paketů
- Extrakce IP adres a portů
- Kalkulace velikosti dat

3.2.3 Řazení a datové struktury

Datové struktury v `linked_list.c`:

- Implementace obousměrného spojového seznamu
- Implementace automatického řazení pomocí bublinkového algoritmu

3.2.4 Zobrazování statistik

Modul statistik v `stats.c`:

- Sčítání identických komunikací
- Formátování dat pro zobrazení
- Aktualizace UI

3.3 Implementační Detaily

3.3.1 Multithreading

Aplikace používá dvě vlákna:

```
capture_thread: Průběžné zachytávání paketů
stats_thread: Aktualizace uživatelského rozhraní
```

3.3.2 Řadící algoritmy

Řazení komunikací v programu isa-top probíhá postupně dle zachycené komunikace. Všechny komunikace jsou uloženy ve struktuře CommunicationInfo, která uchovává informace o IP adresách, přenosech, počtech paketů a podobně.

Podle této struktury jsou v rámci spojového seznamu komunikace umístěny výše, pakliže přenesly po přijetí dalšího paketu v dané komunikaci více dat nebo paketů.

3.3.3 Knihovna ncurses

Pro zobrazení statistik uživateli v příkazové řádce byla použita knihovna ncurses.

Kapitola 4

Návod na použití

4.1 Instalace

Požadavky:

- libpcap knihovna
- ncurses knihovna
- C compiler (gcc/clang)
- make

Build příkaz:

```
make
```

4.2 Možnosti spuštění v příkazové řádce

Základní syntaxe:

```
sudo ./isa-top -i interface [-s b|p] [-t interval] [-c] [-h]
```

Možnosti:

- `-i interface`: Síťové rozhraní, na kterém zachytávat pakety
- `-s b|p`: Řazení podle bitů/s nebo paketů/s
- `-t interval`: Jak často aktualizovat statistiky, v sekundách
- `-c`: Kumulativní statistiky (jinak jsou každý interval nulovány)
- `-h`: Zobraz help zprávu

4.3 Popis uživatelského rozhraní

Na displeji je zobrazeno:

- Páry zdrojových a cílových adres IP:port

- Typ protokolu (TCP, UDP, ICMP...)
- Průměrné rychlosti přenosu (Rx/Tx)
- Průměrné počty paketů

Kapitola 5

Testování

Testování bylo provedeno pomocí:

- Předdefinovaných PCAP souborů
- Skutečného sledování síťového provozu

Testy lze najít v souboru `pcap_file_tests.c` a spustit pomocí make test.

5.1 Testovací případy

5.1.1 ICMP Provoz

Testováno souborem `icmp-data.pcap`, který obsahuje příklad několika komunikací s portem ICMPv4 i ICMPv6. Byly vytvořeny automatické testy zkoumající následující:

- Kontrola správného počtu zachycených paketů
- Kontrola správných přenosových rychlostí
- Správná detekce protokolů ICMP

5.1.2 Smíšený provoz

Test pomocí zachycení náhodné síťové komunikace po krátkou dobu je uložen v souboru `test_capture.pcap`. Následně byla komunikace prozkoumána v programu Wireshark a její výstupy programově porovnány s programem isa-top.

- Zachycování pouze IPv4 a IPv6 paketů
- Kontrola portů
- Správný počet zachycených paketů
- Korektní řazení dle paketů a přenesených bitů

5.1.3 Internet Speed Test

Byly prováděny průběžné testy, kdy byl výstup programu isa-top porovnáván k výsledkům testu rychlosti internetu.

5.2 Výsledky

Testy prokázaly následující:

- Počítání paketů zobrazuje přesné výsledky
- Správné rozlišování portů
- Přesné počítání přenosových rychlostí

Kapitola 6

Literatura

Program isa-top byl implementován s pomocí následujících zdrojů:

6.1 Knihovny a manuály

- PCAP library programming manual. The Tcpdump Group. Dostupné z: https://www.tcpdump.org/manpages/pcap.3pcap.html
- NCURSES Programming Guide. GNU Project. Dostupné z: https://invisible-island.net/ncurses/ncurses.html
- Using getopt to parse command line options. GNU Project. Dostupné z: https://www.gnu.org/software/libc/manual/html_node/Example-of-Getopt.html
- Vlákno (informatika). Wikipedia. Dostupné z: https://cs.wikipedia.org/wiki/Vl%C3%A1kno_(informatika)
- Lineární seznam. Wikipedia. Dostupné z: https://cs.wikipedia.org/wiki/Line%C3%A1rn%C3%AD_seznam
- Bublinkové řazení. Wikipedia. Dostupné z: https://cs.wikipedia.org/wiki/Bublinkov%C3%A9_%C5%99azen%C3%AD
- Ethernet frame. Wikipedia. Dostupné z: https://en.wikipedia.org/wiki/Ethernet_frame
- iftop - nástroj pro analýzu síťového provozu. Dostupné z: https://www.ex-parrot.com/pdw/iftop/
- Wireshark User's Guide. Wireshark Foundation. Dostupné z: https://www.wireshark.org/docs/