
**VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ**
**BRNO UNIVERSITY OF TECHNOLOGY**

**FAKULTA INFORMAČNÍCH TECHNOLOGIÍ**
**FACULTY OF INFORMATION TECHNOLOGY**

**ÚSTAV INFORMAČNÍCH SYSTÉMŮ**
**DEPARTMENT OF INFORMATION SYSTEMS**


# ANALYZÁTOR PAKETŮ

**PACKET ANALYSIS**

**PROJEKTOVÁ DOKUMENTACE**
**PROJECT DOCUMENTATION**


**AUTOR PRÁCE** **PATRIK MICHALÁK**
**AUTHOR**


**BRNO 2017**

# Obsah

# Kapitola 1

# Úvod

Paket je blok dát, ktorý sa v sieti prenáša a spracováva ako celok. Má vlastnú štruktúru, ktorá obsahuje hlavičku a dáta. Hlavička obsahuje dôležité informácie o protokoloch, adresách, dĺžky v bitoch... S týmito informáciami pracujú rôzne programy, ktoré analyzujú pakety. Príkladom je konzolová aplikácia tcpdump alebo Wireshark, ktorý má aj grafické užívateľské rozhranie.

## 1.1 Návrh programu

Aplikácie na analýzu dát používajú aplikačné rozhranie pcap (packet capturing), ktoré slúži na odchytávanie sieťovej komunikácie. [**?**] V prípade programu isashark, ktorý je implementovaný pre unixové systémy použijeme knižnicu libcap. Program isashark načíta záznamy zo súboru a výstupom programu je reťazec záznamov analýzy vypísaný na štandardný výstup. Program umožňuje agregáciu a radenie záznamov. Isashark podporuje tieto protokoly:

- Vrstva sieťového rozhrania:
  - Ethernet
  - IEEE 802.1Q, IEEE 802.1ad
- Sieťová vrstva:
  - IPv4 (podporuje defragmentáciu)
  - IPv6 (podporuje rozšírené hlavičky)
  - ICMPv4
  - ICMPv6
- Transportná vrstva:
  - TCP
  - UDP

# Kapitola 2

# Popis implementácie

Program isashark pozostáva z hlavných a pomocných funkcií, ktoré sa sekvenčne vykonávajú až do dosiahnutia konečného tvaru štruktúr paketov a ich výpisu na štandardný výstup.

Štruktúra jendého paketu pozostáva s premenných, ktoré sú potrebné pre chod programu a konečný vypis. Tieto štruktúry sa uchovávajú v poli std::vector.

Funkcie zdrojového kódu:

- printErr() - vypis hlášky na stderr a ukončenie programu chybovým kódom
- printHelp() - výpis nápovedy a ukončenie programu
- printIcmp() - textový popis vysvetlujúci typ a kód ICMP
- hexToString() - pomocná funkcia na prevod hexadecimálneho čísla na decimálne
- clearPacketStruct() - pomocná funkcia, ktorá priradí do využitej štruktúry paketu prázdne hodnoty
- formatMac() - pomocná funkcia na formátovanie mac adresy do šesť dvojíc hexadecimálnych čísel
- printAggrArray() - konečný výpis agregovaných paketov s alebo bez limitu
- printPacketArray() - štandardný konečný výpis paketov s alebo bez limitu
- packetDefragmentation() - spájanie fragmentov paketu
- skipExtHeader() - preskočenie rozšírenej hlavičky v IPv6 pakete [**?**]
- switchL4() - analýza hlavičky paketu na úrovni transportnej vrstvy
- switchL3() - analýza hlavičky paketu na úrovni sieťovej vrstvy
- aggregatePackets() - agregovanie paketov podľa kľúča
- sortPackets() - zoradenie paketov podľa kľúča
- getOutput() - sekvencia volania funkcií po spracovaní paketov zo vstupu
- parseArgs() - spracovanie a kontrola validnosti argumentov
- main() - hlavná funkcia programu, spracovanie paketov zo vstupu

## 2.1 Sekvencia programu

1. Program spracuje argumenty a nastaví hodnoty z prepínačov.
2. Následne sa vykoná cyklus, ktorý zo všetkých zadaných súborov spracuje záznamy paketov a vrámci toho volá funkcie pre anylýzu paketu na úrovni sieťovej a transportnej vrstvy.
3. Nasleduje spájanie prípadných fragmentov paketu.
4. Ak je nastavený prepínač a kľúč pre agregáciu, vykoná sa funkcia pre agregovanie.
5. Ak je nastavený prepínač a kľúč pre zoradenie, vykoná sa funkcia pre zoradenie paketov.
6. Na štandardný výstup sa vypíšu spracované záznamy paketov. Ak je nastavený prepínač na limit výpisu, vypíše sa iba požadovaný počet záznamov.

## 2.2 Defragmentácia paketu

Defragmentácia paketov nastáva hneď po načítaní všetkých záznamov zo vstupných súborov. Fragment definujeme ako paket, ktorý má nastavený bit "viac fragmentov", v zdrojovom kóde isashark označený ako bool moreFrag, alebo nenulové pole "offset fragmentu", v kóde ako integer offsetFrag. Ak má paket nastavený bit moreFrag ale offsetFrag je nulové, potom je tento fragment počiatočným fragmentom. Tento fragment obsahuje v štruktúre platné záznamy po sieťovú vrstvu. Ak má paket moreFrag ako false ale offsetFrag je nenulové, tento fragment je koncovým fragmentom. Offset koncového fragmentu + dĺžka jeho dát nám udáva celkovú dĺžku znovuzložených dát defragmentovaného paketu.

Hlavný cyklus prechádza všetky záznamy paketov. Ak sa objaví paket ktorý nemá nastavený ani moreFrag ani offsetFrag, paket sa vloží do pomocného poľa defragmentovaných paketov a pokračuje sa na ďalší paket. Ak sa nájde paket, ktorý je počiatočný fragmentom, vytvorí sa buffer, kde sa budú ukladať jednotlivé dáta fragmentov a inicializuje sa bitová maska, ktorá slúži na kontrolu úplnosti dát. Následne sa v cykle hľadajú pakety, ktoré majú s počiatočným fragmentom spoločnú identifikačnú hodnotu, zdrojovú a cieľovú IP adresu. Dáta fragmentov sa následne ukladajú do bufferu na pozíciu danú offsetom fragmentu. Po znovuzložení dát z fragmentov sa podľa bitovej masky kontroluje úplnosť paketu. Ak je paket úplný, analyzujú sa dáta na úrovni sieťovej vrstvy a paket sa vloží do pomocného poľa defragmentovaných paketov.

## 2.3 Agregácia a radenie paketov

Agregácia podľa kľúča sa vykonáva cyklom, kde pre všetky pakety kontrolujeme, či pre daný paket už existuje agregačný záznam, kde by sa mohol priradiť. Ak už záznam existuje, inkrementuje sa počet paketov v zázname a pripočíta dĺžka paketu. Ak záznam neexistuje, vytvorí sa nový záznam a pridá do pola záznamov.

Zoradenie záznamov podľa kľúča je uskutočnené bubble sort algoritmom inšpirované predmetom Algoritmy. [**?**]

# Kapitola 3

# Základné informácie o programe

Isashark je offline konzolová aplikácia na analýzu, agregovanie a radenie záznamov sieťovej prevázky obsahujúci vybrané protokoly z rodiny TCP/IP.

Program je možné spúšťať s nasledujúcimi argumentmi:

```
isashark [-h] [-a aggr-key] [-s sort-key] [-l limit] [-f filter-expression] file
```

kde:

- -h Vypísanie nápovedy a ukončenie programu.
- -a aggr-key Agregovanie záznamov paketov podľa agregačného kľúča. Agregčný kľúč môže byť:
  - srcmac - zdrojová MAC adresa
  - dstmac - cieľová MAC adresa
  - srcip - zdrojová IP adresa
  - dstip - cieľová IP adresa
  - srcport - číslo zdrojového transportného portu
  - dstport - číslo cieľového transportného portu
- -s sort-key Radenie záznamov paketov podľa kľúča:
  - packets - počet paketov (efekt iba pri agregovanom výpise)
  - bytes - počet bajtov
- -l limit Nezáporné celé číslo určujúce limit počtu vypísaných záznamov.
- -f filter-expression Program spracuje iba pakety, ktoré vyhovujú danému filter-expression.
- file Cesta k súboru alebo viac súborom vo formáte pcap.

# Literatura

[1] pcap.
URL https://sk.wikipedia.org/wiki/Pcap

[2] Cisco: IPv6 Extension Headers Review and Considerations. 2006.
URL https://www.cisco.com/en/US/technologies/tk648/tk872/technologies_white_paper0900aecd8054d37d.pdf

[3] Prof. Ing. Honzik CSc., J.: Opora predmetu Algoritmy. 2017.
URL https://wis.fit.vutbr.cz/FIT/st/course-files-st.php?file=%2Fcourse%2FIAL-IT%2Ftexts%2FOpora-IAL-2017-verze-17-B.pdf&cid=11418