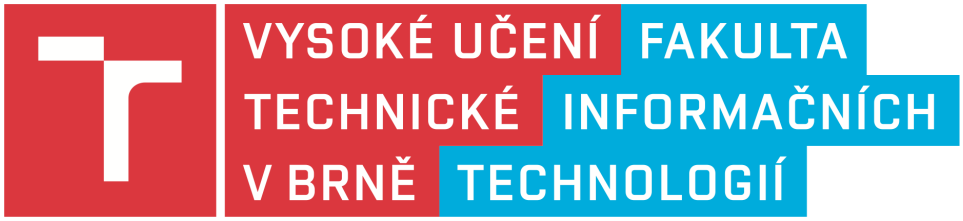


VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

ISA - Síťové aplikace a správa sítí

Projekt -Varianta: Generování NetFlow dat ze zachycené síťové komunikace

Lucia Makaiová (xmakai00)

12. novembra 2022

Obsah

1 Úvod

1.1 Popis projektu

Netflow exportér *flow* je C++ applikácia, slúžiaca na analýzu siete. Aplikácia vytvára NetFlow záznamy zo zachytených sieťových dát vo formáte pcap a následne tieto záznamy posiela ďalej netflow kolektoru pomocou User Datagram Protokolu (UDP). Implementovaná je najpoužívanejšia verzia -Netflow v5.

1.2 Formát volania programu

./flow [-f <súbor>] [-c <netflow_kolektor>[:<port>]] [-a <aktívny_časovač>] [-i <neaktívny_časovač>] [-m <počet>]

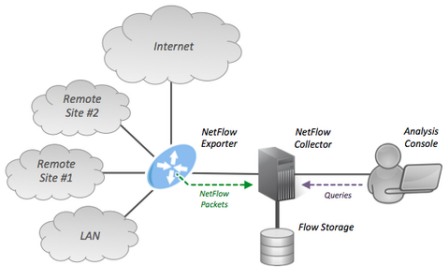
Činnosť programu je možné dodatočne špecifikovať použitím nasledujúcich argumentov pri volaní programu(na poradí argumentov nezáleží):

- -f -meno **súboru** na analýzu (default = STDIN)
- -c -**IP adresa**, alebo **hostname** NetFlow kolektoru, volitelne aj UDP **port** (default = 127.0.0.1:2055)
- -a - interval v sekundách, po ktorom sú **aktívne** flows exportované (default = 60)
- -i - interval v sekundách, po ktorom sú **neaktívne** flows exportované (default = 10)
- -m - **veľkosť flow-cache** (default = 1024)

2 Prehlad naštudovaných informácií z literatúry

NetFlow protokol bol vyvinutý spoleočnosťou Cisco Systems. Jeho hlavným účelom je monitorovanie sieťovej prevádzky na základě IP tokov. [5] Hlavné komponenty:

- **Flow exportér** -agreguje pakety do jednotlivých flow záznamov a posiela ich na kolektor
- **Flow kolektor** -zodpovedá za príjem a spracovanie dát z exportéru
- **Aplikácia na analýzu**



Obr. 1: Architektúra Netflow [5]

V projekte implementujeme exportovanie flow záznamov, ktoré bežne vykonáva router.
Základom NetFlow technológie je **IP tok** (flow) -sekvencia paketov na základe ktorej sú generované NetFlow štatistiky. Táto sekvencia má zhodnú päticu údajov:

- **Cieľová a zdrojová IP adresa** -v prípade ICMP protokolu sú tieto hodnoty nulové
- **Cieľový a zdrojový port**
- **Číslo protokolu**

O každom toku sú následne zaznamenávane údaje v NetFlow štruktúre a každým ďalším príchodzým paketom sú potrebné údaje aktualizované.

Bytes	Contents	Description
0-1	version	NetFlow export format version number
2-3	count	Number of flows exported in this packet (1-30)
4-7	SysUptime	Current time in milliseconds since the export device booted
8-11	unix_secs	Current count of seconds since 0000 UTC 1970
12-15	unix_nsecs	Residual nanoseconds since 0000 UTC 1970
16-19	flow_sequence	Sequence counter of total flows seen
20	engine_type	Type of flow-switching engine
21	engine_id	Slot number of the flow-switching engine
22-23	sampling_interval	First two bits hold the sampling mode; remaining 14 bits hold value of sampling interval

Bytes	Contents	Description
0-3	srcaddr	Source IP address
4-7	dstaddr	Destination IP address
8-11	nexthop	IP address of next hop router
12-13	input	SNMP index of input interface
14-15	output	SNMP index of output interface
16-19	dPkts	Packets in the flow
20-23	dOctets	Total number of Layer 3 bytes in the packets of the flow
24-27	First	SysUptime at start of flow
28-31	Last	SysUptime at the time the last packet of the flow was received
32-33	srcport	TCP/UDP source port number or equivalent
34-35	dstport	TCP/UDP destination port number or equivalent
36	pad1	Unused (zero) bytes
37	tcp_flags	Cumulative OR of TCP flags
38	prot	IP protocol type (for example, TCP = 6; UDP = 17)
39	tos	IP type of service (ToS)
40-41	src_as	Autonomous system number of the source, either origin or peer
42-43	dst_as	Autonomous system number of the destination, either origin or peer
44	src_mask	Source address prefix mask bits
45	dst_mask	Destination address prefix mask bits
46-47	pad2	Unused (zero) bytes

Obr. 2: Netflow hlavička a Netflow záznam toku [2]

3 Použité knižnice

- arpa/inet.h -funkcie htons a ntohs na prevod medzi hostiteľským a sieťovým poradím bajtov
- pcap.h -načítavanie paketov zo súboru a práca s nimi
- getopt.h -parsovanie argumentov príkazového riadka
- netdb.h -použitie funkcie gethostbyname pre rozoznanie IP adresy na ktorú sa majú exportovať toky
- sys/socket.h -tvorba sieťového socketu pre export a pripojenie k nemu
- map -agregácia paketov patricich do rovnakého toku

Knižnice pre prácu s hlavičkami pomocou typecastovania paketu:

- netinet/ether.h
- netinet/tcp.h
- netinet/udp.h
- netinet/ip_icmp.h

4 Časti programu a popis ich implementácie

4.1 Spracovanie argumentov -arguments.cpp

Na spracovanie argumentov príkazového riadka slúži modul arguments.cpp. Argumenty sú spracované pomocou knižnice getopt[3]. Keďže argumenty nemajú dlhé verzie, použitá bola krátka verzia getopt príkazu.
Následne prebiehajú v module aj kontroly vstupov a pre argument -c definujúci hostname je dodatočne naimplementovaná funkcia na spracovanie argumentu v rôznych možných tvaroch.
Funkcia ako prvé otestuje či sa nachádza v definícii hostname aj port a prípadne overí jeho korektnosť. Následne pre zvyšok vstupu zistí či ide o platnú ip adresu, poprípade o hostname ktoré sa na ňu pomocou funkcie gethostbyname() dá previesť.

4.2 Inicializácia klienta pre exportovanie flows -netflow_generator.cpp

Netflow záznamy sú bežne exportované protokolom UDP a zachytené pomocou Netflow kolektoru. Implementácia udp klienta je inšpirovaná príkladom z prednášky[4]. Funkcia client vytvorí nové spojenie a samotný export, teda zasielanie dát cez UDP prebieha jendotlivo pri každom zavolaní funkcie export_flow v hlavnom module flow.

4.3 Hlavná smyčka spracovávania paketov.

Spracovanie paketov zabezpe4uje knižnica pcap.[1] Na začiatku programu sa otvorí uživateľom špecifikovaný pcap súbor na čítanie pomocou funkcie pcap_open_offline. Pred samotným spracovaním paketov sa skompiluje a aplikuje filter paketov ktorý zabezpečí podporu protokolov TCP, UDP a ICMP. Pakety ktoré používajú iný protokol budú odfiltrované.

Zachytenie samotného paketu prebieha pomocou funkcie *pcap_next*, ktorá vracia ukazateľ typu const u_char na nasledujúci paket v súbore. Táto funkcia a následné spracovanie packetu a jeho protokolov (na spojovej, sieťovej a transportnej vrstve) sa nachádza v hlavnej smyčke programu ktorá sa vykonáva pokým sa program nedostane na koniec súboru. Informácie z hľavičiek protokolov paketu sa získavaju použitím knižníc pre dané protokoly a typecastovaním paketu na štruktúru odpovedajúcu danému protokolu.

4.4 Agregovanie a vytváranie NetFlow tokov

Na agregáciu tokov je použitá štruktúra map. Map je triedené asociatívne pole, ktoré obsahuje páry kľúč – hodnota s jedinečnými kľúčmi. [6]

Ako kľúč je používaná štruktúra obsahujúca vyššie spomínanú päticu, podľa ktorej sa pakety do tokov agregujú. Pre použitie heterogénnej štruktúry na mieste klúča bolo nutné dodefinovať operácie ekvivalencie a porovnávania dvoch klúčov.

Spočiatku dodefinované funkcie nerozlišovali smer komunikácie a spájali teda do jedného toku komunikáciu aj z opačného smeru. Z tohoto dôvodu bolo nutné dodatočne ošetriť funkcie operátorov proti tejto chybe. To bolo docielené násobením cieľovej adresy a portu aby sme pre komunikáciu z opačného smeru dostali odlišnú hodnotu kľúča.

```
//custom comparison operator
bool operator<(const Netflow_base& other) const{
    if (srcaddr + 2*dstaddr + srcport + 2*dstport + prot <
        other.srcaddr + 2*other.dstaddr + other.srcport +2*other.dstport + other.prot) {
        return  true;
    }
    return false;
}
//custom equality operator (dstaddr and port multiplied by to do differ between bidirectional communication)
bool operator==(const Netflow_base& other) const{
    if (srcaddr + 2*dstaddr + srcport + 2*dstport + prot ==
        other.srcaddr + 2*other.dstaddr + other.srcport +2*other.dstport + other.prot) {
        return  true;
    }
    return false;
}
```

Obr. 3: Dodefinované operátory na porovnanie a ekvivalenciu štruktúry

Autor

Lucia Makaiová [xmakai00]

Literatúra

[1] CARSTENS, T.: PROGRAMMING WITH PCAP. [online], rev. 20. leden 2022.
URL https://www.tcpdump.org/pcap.html

[2] Cisco Systems, I.: NetFlow Export Datagram Format. [online], 2022.
URL https://www.cisco.com/c/en/us/td/docs/net$_m$gmt/net$flow_c$ollection$_e$ngine/3 – 6/user/guide/format.html

[3] Free Software Foundation, I.: Example of Getopt. [online], 1993–2022.
URL https://www.gnu.org/software/libc/manual/html_node/Example-of-Getopt.html

[4] Matousek, P.: Echo-udp-client2. [online], 2016.
URL https://moodle.vut.cz/

[5] Wikipedia, I., Wikimedia Foundation: NetFlow. [online], 2022.
URL https://en.wikipedia.org/wiki/NetFlow

[6] WikiSysop: Cpp container map. [online], 2011.
URL https://en.cppreference.com/w/cpp/container/map