VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta informačních technologií

Sieťové aplikácie a správa sieti

2022/2023

Generovanie NetFlow dát

zo zachytenej sieťovej komunikácie

Michal Uhrecký (xuhrec00)

Brno, 7. 11. 2022

Abstrakt

Cieľom tejto práce je prezentovať NetFlow exportér, ktorý zo zachytených sieťových dát vo formáte pcap vytvorí záznamy NetFlow, ktoré odošle na kolektor.

Obsah

1. Úvod

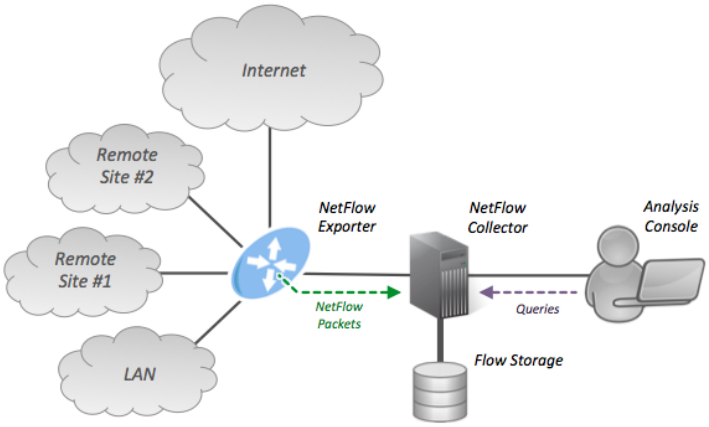
Sledovaním jazdy automobilov dokážeme pozorovať vyťaženosť ciest, zdroj a cieľ automobilov. Dokážeme identifikovať upchaté oblasti, vieme stanoviť dennú aktivitu a v prípade trestného činu sa dajú použiť záznamy. Podobne ako v cestnej doprave, ide sledovať a monitorovať tok sieťových dát. Záznamy o tokoch sa využívajú pre správu siete. V prípade bezpečnostného incidentu je možné identifikovať komunikáciu v danom čase. Je možné použiť štatistické dáta o tokoch na identifikáciu neobvyklých prenosov a tiež aj na analýzu prevádzky siete.[1]

2. NetFlow

NetFlow je sieťový protokol vytvorený spoločnosťou Cisco pre zbieranie informácií IP prevádzky a monitorovanie sieťového toku. Analyzovaním NetFlow dát je možné získať obraz prevádzky siete.[2]

2.1 Základné prvky systému NetFlow

Základné prvky systému NetFlow sú exportér, komunikačný protokol NetFlow, kolektor a nástroje pre zobrazenie dát. Exportér je sonda/router pre získavanie štatistík o tokoch. Komunikačný protokol NetFlow slúži na komunikáciu medzi exportérom a kolektorom. Kolektor je používaný ako zariadenie, ktoré ukladá záznamy o tokoch.[3]



Obrázok 1: Princíp NetFlow[4]

[1] MATOUŠEK, Petr. Monitorování toků NetFlow: Síťové aplikace a správa sítí (prezentácia).
[2] What Is NetFlow?. *SolarWinds* [online]. [cit. 2022-11-03]. Dostupné z: https://www.solarwinds.com/netflow-traffic-analyzer/use-cases/what-is-netflow
[3] MATOUŠEK, Petr. Monitorování toků NetFlow: Síťové aplikace a správa sítí (prezentácia).
[4] NetFlow architecture. In: *Wikipedia* [online]. [cit. 2022-11-03]. Dostupné z: https://en.wikipedia.org/wiki/NetFlow#/media/File:NetFlow_Architecture_2012.png

2.2 NetFlow exportér

NetFlow exportér je sieťové zariadenie (prípadne softvér), ktoré monitoruje prechádzajúcu prevádzku siete. Príkladné zariadenie môže byť router alebo firewall. Exportér zhromažďuje dáta respektíve sieťové pakety do tokov a exportuje dané toky na kolektory keď rozhodne, že toky expirovali.[5]

Exportér rozhoduje, ktoré toky sú nové podľa spoločných vlastností jednotlivých paketov. Vlastnosti môžu byť rôzne podľa konkrétneho exportéra. Často to je týchto 7 vlastností: zdrojové rozhranie, zdrojová IP adresa, cieľová IP adresa, protokol, typ služby (ToS), zdrojový port, cieľový port (obrázok 2). Ak príde paket, ktorý má tieto spoločné vlastnosti už s uloženým tokom, tak ho aktualizuje, inak vytvára nový tok.[6]

Expirácia uložených tokov je založená na neaktívnom časovači, aktívnom časovači, pamäti uložených tokov, RST alebo FIN TCP príznaku. Neaktívny časovač meria čas od posledného paketu v danom toku. Aktívny meria čas od prvého paketu v danom toku. Ak jeden z časovačov vyprší, tak sa tok exportuje na kolektor. V prípade plnej pamäte sa exportuje najstarší tok. Ak sa daný tok skončil, napríklad u TCP príznak RST či FIN, tak sa tiež tok exportuje.[7]

Exportér podporuje aj filtrovanie a vzorkovanie dát. Vzorkovať sa môže podľa poradia, času, veľkosti. Dôvodom býva zníženie nárokov na hardware. Filtrovanie slúži na klasifikáciu prevádzky na základe hodnôt v hlavičke paketu. Vzorkovanie a filtrovanie sa môže kombinovať.[8]

SrcIf	SrcIPadd	DstIf	DstIPadd	Proto	ToS	Flgs	Pkts	SrcPort	SrcAS	DstPort	DstAS	Bytes/Pkt	Active	Idle
Fa1/0	173.100.21.2	Fa0/0	10.0.227.12	UDP	80	10	11000	162	5	162	15	1528	1745	4
Fa1/0	173.100.3.2	Fa0/0	10.0.227.12	TCP	40	0	2491	21	196	21	15	740	41,5	1
Fa1/0	173.100.20.2	Fa0/0	10.0.227.12	UDP	80	10	10000	161	180	161	15	1428	1145,5	3
Fa1/0	173.100.6.2	Fa0/0	10.0.227.12	TCP	40	0	2210	25	180	25	15	1040	24,5	14

Obrázok 2: Zlučovanie paketov[9]

[5] NETFLOW COMPONENTS: NETFLOW EXPORTER, NETFLOW COLLECTOR AND NETWORK ANALYZER [online]. [cit. 2022-11-05]. Dostupné z: https://www.firewall.cx/networking-topics/protocols/netflow/1257-netflow-basics-netflow-components-rfc-history.html

[6] MATOUŠEK, Petr. Monitorování toků NetFlow: Síťové aplikace a správa sítí (prezentácia).

[7] Tamtiež.

[8] Tamtiež.

[9] Obrázok prevziaty z prezentácie: MATOUŠEK, Petr. Monitorování toků NetFlow: Síťové aplikace a správa sítí.

3. Implementácia

V tejto časti budem popisovať návrh a implementáciu môjho NetFlow exportéru. Cieľom bolo vytvoriť program s názvom „flow“ v jazyku C/C++, ktorý načíta zo súboru vo formáte pcap dáta a spracuje ich ako záznamy tokov, ktoré odošle na kolektor. Na komunikáciu medzi exportérom a kolektorom som sa rozhodol použiť protokol *NetFlow v5*. Jeho formát je možné vidieť na obrázkoch 3 a 4.

V5 Flow record format

Bytes	Fields	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) byte
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

Obrázok 3: NetFlow formát záznamu[10]

[10] Obrázok prevziaty z: NetFlow V5 formats. *IBM* [online]. [cit. 2022-11-05]. Dostupné z: https://www.ibm.com/docs/en/npi/1.3.0?topic=versions-netflow-v5-formats

V5 header format

Bytes	Fields	Description
0-1	version	NetFlow export format version number
2-3	count	Number of flows that are exported in this packet (1-30)
4-7	SysUptime	Current time in milliseconds since the export device started
8-11	unix_secs	Current count of seconds since 0000 Coordinated Universal Time 1970
12-15	unix_nsecs	Residual nanoseconds since 0000 Coordinated Universal Time 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

Obrázok 4: NetFlow formát hlavičky[11]

[11] Obrázok prevziaty z: NetFlow V5 formats. *IBM* [online]. [cit. 2022-11-05]. Dostupné z: https://www.ibm.com/docs/en/npi/1.3.0?topic=versions-netflow-v5-formats

3.1 Spracovanie argumentov

Argumenty, ktoré je potrebné spracovať sú: názov súboru, hostname/IP a port kolektoru, aktívny časovač, neaktívny časovač, veľkosť pamäti tokov.

Každý argument má východziu hodnotu v prípade nezadania. Ak nie je zadaný súbor, očakáva sa *štandardný vstup*. Východzia hodnota kolektoru na ktorý sa posiela je *127.0.0.1:2055*. Aktívny časovač je nastavený na *60 sekúnd* a neaktívny na *10 sekúnd*. Veľkosť pamäti je nastavená na *1024 tokov*.

Argumenty sú spracované pomocou jedného *for* cyklu.

3.2 Spracovanie pcap súboru

Na spracovanie pcap súboru používam knižnicu *pcap.h*. Na otvorenie pcap súboru používam funkciu *pcap_fopen_offline()*, kde načítam súbor zadaný v argumente alebo štandardný vstup.[12]

V otvorenom súbore som schopný spracovávať pakety pomocou *while* cyklu a funkcie *pcap_next()*. Vo vnútri cyklu spracovávam jeden paket. V ňom rozhodnem či daný paket budem spracovávať pomocou príznakov, ktoré si nastavím na základe protokolu lebo podporované protokoly sú: *TCP*, *UDP* a *ICMP*.[13]

[12] CARSTENS, Tim. PROGRAMMING WITH PCAP. *TCPDUMP & LIBPCAP* [online].[cit. 2022-11-06]. Dostupné z: https://www.tcpdump.org/pcap.html

[13] Tamtéž.

3.3 Spracovanie paketu

V prvom rade je potrebné uviesť, kde chcem spracovaný paket ukladať. Vytvoril som štruktúru *NetFlow v5* paketu podľa *NetFlow v5* protokolu podľa obrázkov 3 a 4.

```
typedef struct netflow_v5_flow_format
{
    u_int32_t srcaddr; // získaná z iphdr
    u_int32_t dstaddr; // získaná z iphdr
    u_int32_t nexthop; // nepoznám = 0
    u_int16_t input; // nepoznám = 0
    u_int16_t output; // nepoznám = 0
    u_int32_t dPkts; // pri každom pridanom pakete inkrementujem
    u_int32_t dOctets; /* pri každom pridanom pakete pridám   .
bajty na tretej vrstve získané z iphdr */
    u_int32_t First; // získaný zo systémového času
    u_int32_t Last; // získaný zo systémového času
    u_int16_t srcport; // získaný z tcphdr/udphdr alebo 0 (ICMP)
    u_int16_t dstport; // získaný z tcphdr/udphdr alebo 0 (ICMP)
    u_int8_t pad1; // 0
    u_int8_t tcp_flags; // získané z tcphdr
    u_int8_t prot; // získaný z iphdr
    u_int8_t tos; // získaný z iphdr
    u_int16_t src_as; // nepoznám = 0
    u_int16_t dst_as; // nepoznám = 0
    u_int8_t src_mask; // nepoznám = 0
    u_int8_t dst_mask; // nepoznám = 0
    u_int16_t pad2; // 0
} NETFLOW_V5_FLOW_FORMAT;
```

```
typedef struct netflow_header
{
    u_int16_t version; // používam verziu 5
    u_int16_t count; // vždy exportujem iba 1 tok
    u_int32_t SysUptime; /* čas od prvého paketu získam z metadát
o pakete */
    u_int32_t unix_secs; // získam z metadát o pakete
    u_int32_t unix_nsecs; // získam z metadát o pakete
    u_int32_t flow_sequence; // po každom exporte inkrementujem
    u_int8_t engine_type; // 0
    u_int8_t engine_id; // 0
    u_int16_t sampling_interval; // 0
} NETFLOW_HEADER;



typedef struct net_flow_packet
{
    NETFLOW_HEADER header;
    NETFLOW_V5_FLOW_FORMAT record;
} NET_FLOW_PACKET;
```

Potom som si vytvoril pole štruktúr *NetFlow v5* paketu o veľkosti zadanej v argumente alebo východzej 1024, kde budem ukladať jednotlivé toky.

```
NET_FLOW_PACKET flow_cache[flow_cache_size];
```

Na získanie informácií z paketu využívam štruktúry *ether_header*, *iphdr*, *udphdr* a *tcphdr*, ktoré sú dostupné v sieťových knižniciach, avšak kvôli preložitelnosti na serveri merlin som si štruktúry tcphdr a udphdr nakopíroval do môjho hlavičkového súboru.[14]

[14] Odkaz na sieťové hlavičkové súbory: *Index of /usr/include/netinet* [online]. [cit. 2022-11-07]. Dostupné z: https://sites.uclouvain.be/SystInfo/usr/include/netinet/

Spracovanie paketu začína preskúmaním pomocou *for* cyklu už uložených tokov, kde sa porovnajú časy tokov s časom aktuálneho paketu a rozhodne sa o ich expirácii na základe časovačov. Kedže je implementácia pamäti tokov pomocou pola, tak pri exporte paketu je potrebné pole preindexovať, aby bola zaplnená vzniknutá „diera". V tom istom cykle je kontrolované aj či daný paket treba zaradiť ako nový tok alebo aktualizovať už existujúci. Táto kontrola je na základe 6 spoločných vlastností a to: *zdrojová IP adresa*, *cieľová IP adresa*, *zdrojový port*, *cieľový port*, *protokol* a *typ služby*. V prípade nového a starého toku využívam príznak *is_new_flow*.

Potom prebieha kontrola pomocou podmienky na plnosť pamäti tokov. V prípade, že exportér nemal čo vyexportovať na základe časovačov a ďalší paket je nový tok, tak je potrebné vyexportovať najstarší tok. Najstarší tok som zvolil ten, ktorý mal najstarší čas posledného paketu. Prebiehajú podobné 2 *for* cykly, kde v prvom cykli sa zistí najstarší paket a v druhom ho exportujem, kde zasa následuje preindexovanie.

Exportér nepodporuje kontrolu RST či FIN príznaku u TCP.

Pomocou vyššie spomínaného príznaku *is_new_flow* rozhodujem či aktualizujem alebo vytvorím tok. Ak aktualizujem tok, tak aktualizujem jeho bajty na 3. vrstve, čas posledného paketu, počet paketov, TCP príznaky a systémový čas (čas od úplne prvého paketu). Inak vytvorím nový tok uložený v štruktúre *NET_FLOW_PACKET*.

3.4 Exportovanie toku

NetFlow je implementovaný nad transportným protokolom UDP. Preto si vytváram UDP socket *sock*, cez ktorý exportujem jednotlivé toky.

Ak ostali v pamäti toky po spracovaní paketov, tak sa na záver exportujú.[15]

[15] UDP Server-Client implementation in C. *GeeksforGeeks* [online]. [cit. 2022-11-07]. Dostupné z: https://www.geeksforgeeks.org/udp-server-client-implementation-c

4. Testovanie

Testovanie prebiehalo pomocou nástrojov zo sady *nfdump,* a to *nfdump* a *nfcapd*. Tiež som porovnával výstup môjho exportéra a exportéra *softflowfd*. Výstupy boli niekedy rozdielne, lebo exportér softflowfd má niektoré iné implementačné detaily na základe ktorých exportuje.

```
michal-ubuntu@DESKTOP-U3MVS18:~/isaprojekt$ nfdump -r nfcapd.202210271750
Date first seen          Event  XEvent Proto      Src IP Addr:Port          Dst IP Addr:Port     X-Sr
2022-09-28 00:32:50.126 INVALID  Ignore ICMP    100.64.208.103:0     ->    66.254.114.41:0.0
2022-09-28 00:32:50.137 INVALID  Ignore ICMP     66.254.114.41:0     ->   100.64.208.103:0.0
Summary: total flows: 2, total bytes: 336, total packets: 4, avg bps: 2643, avg pps: 3, avg bpp: 84
Time window: 2022-09-28 00:32:50 - 2022-09-28 00:32:51
Total flows processed: 2, Blocks skipped: 0, Bytes read: 288
Sys: 0.000s flows/second: 0.0        Wall: 0.001s flows/second: 1249.2
```

Obrázok 5: Výstup testovaného súboru

5. Návod na použitie

Program sa preloží pomocou pridaného Makefilu na linuxovom systéme pomocou príkazu *make*. Je potrebné mať počúvajúci kolektor na konkrétnom porte. Je možné použiť kolektor *nfcapd*, ktorý počúva na porte 2055 pomocou príkazu *nfcapd -T all -l . -I any -p 2055* (viac na obrázku 6). Následne sa z druhého terminálu spustí program s voliteľnými argumentami (viac na obrázku 7).

./flow [-f <file>] [-c <netflow_collector:port>] [-a <active_timer>] [-i <inactive_timer>] [-m <count>]

-f <file> - meno analyzovaného súboru alebo STDIN

-c <neflow_collector:port> - IP adresa, alebo hostname NetFlow kolektoru, voliteľne aj UDP port (127.0.0.1:2055, pokiaľ nie je špecifikované)

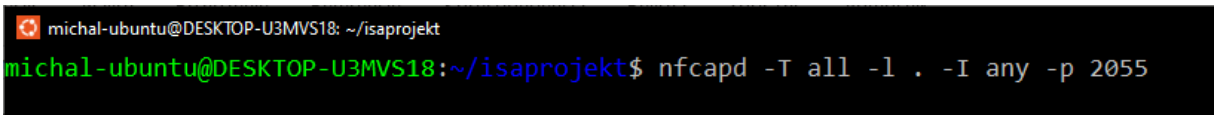
-a <active_timer> - interval v sekundách, po ktorom sa exportujú aktívne záznamy na kolektor (60, pokiaľ nie je špecifikované)

-i <seconds> - interval v sekundách, po jeho vypršaní sa exportujú neaktívne záznamy na kolektor (10, pokiaľ nie je špecifikované)

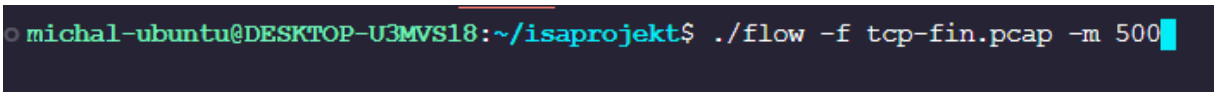
-m <count> - veľkosť pamäti tokov. Pri dosiahnutí max. veľkosti dôjde k exportu najstaršieho záznamu v pamäti na kolektor (1024, pokiaľ nie je špecifikované).

Všetky parametry sú brané ako voliteľné. Pokiaľ niektorý z parametrov nie je uvedený, použije sa miesto neho východzia hodnota.

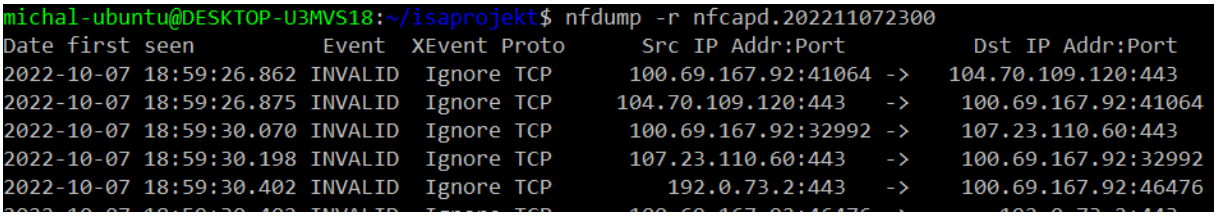
Vygeneruje sa mi nfcapd súbor, ktorý je možné vizualizovať pomocou nástroja nfdump príkazom *nfdump -r nfcapd.datum* (viac na obrázku 8).



Obrázok 6: Spustenie kolektora na porte 2055



Obrázok 7: Spustenie programu so súborom tcp-fin.pcap s veľkosťou pamäti 500



Obrázok 8: Zobrazenie súboru pomocou nfdump

6. Záver

Monitorovanie sieťovej prevádzky je veľmi dôležité a preto mi tento projekt prišiel ako veľmi zaujímavý. Naučil som sa ako sa prakticky združujú pakety, ktoré majú spoločné vlastnosti a tým pádom mám oveľa prehľadnejšie záznamy o pohyboch na sieti. V projekte bolo veľmi užitočné, že som už mal skúsenosti so spracovaním paketu, kedže minulý rok v predmete Počítačové siete a komunikácie som skúšal vytvoriť packet sniffer, ktorý mal spracovanie paketu veľmi podobné. Zo začiatku bolo náročné pochopiť ako pracovať s časmi v tomto projekte, ale skúšaním na konkrétnych súboroch mi to začalo byť jasnejšie. Keďže som si zvolil implementáciu NetFlow paketu ako pole štruktúr, tak bola výzva vymyslieť indexovanie daného pola po exporte paketu. Verím, že sa mi podarilo úspešne naimplementovať funkčný NetFlow exportér.

Zdroje

CARSTENS, Tim. PROGRAMMING WITH PCAP. TCPDUMP & LIBPCAP [online].[cit. 2022-11-06]. Dostupné z: https://www.tcpdump.org/pcap.html

MATOUŠEK, Petr. Monitorování toků NetFlow: Síťové aplikace a správa sítí (prezentácia).

NetFlow architecture. In: Wikipedia [online]. [cit. 2022-11-03]. Dostupné z: https://en.wikipedia.org/wiki/NetFlow#/media/File:NetFlow_Architecture_2012.png

NETFLOW COMPONENTS: NETFLOW EXPORTER, NETFLOW COLLECTOR AND NETWORK ANALYZER [online]. [cit. 2022-11-05]. Dostupné z: https://www.firewall.cx/networking-topics/protocols/netflow/1257-netflow-basics-netflow-components-rfc-history.html

NetFlow V5 formats. IBM [online]. [cit. 2022-11-05]. Dostupné z: https://www.ibm.com/docs/en/npi/1.3.0?topic=versions-netflow-v5-formats

Odkaz na sieťové hlavičkové súbory: Index of /usr/include/netinet [online]. [cit. 2022-11-07]. Dostupné z: https://sites.uclouvain.be/SystInfo/usr/include/netinet/

UDP Server-Client implementation in C. GeeksforGeeks [online]. [cit. 2022-11-07]. Dostupné z: https://www.geeksforgeeks.org/udp-server-client-implementation-c

What Is NetFlow?. SolarWinds [online]. [cit. 2022-11-03]. Dostupné z: https://www.solarwinds.com/netflow-traffic-analyzer/use-cases/what-is-netflow