Fakulta informačních technologií
Vysoké učení technické v Brně

ISA – Sieťové aplikácia a správa sietí
Generovanie NetFlow dát zo zachytenej sieťovej komunikácie

12.10.2022 Šimon Kadnár (xkadna00)

Obsah

1 Úvod

Cieľom projektu bolo vytvoriť aplikáciu v jazyku C, ktorá analyzuje súbory zachytených paketov a odosiela ich na príslušnú NetFlow zbernicu(kolektor). Program spracováva súbory typu *pcap*. Po spustení prejde celý súbor a odošle flowy packetov podľa zadaných voliteľných argumentov na príslušnú adresu alebo prípadne na základne nastavenú adresu.

2 Teória

2.1 Flow

Zoskupenie paketov ktoré majú rovnaké vlastnosti ako sú: protokol(TCP, ICMP, UDP), typ servisu(tos), zdrojovú a cieľovú adresu(DstAdress, SrcAdress), zdrojový port, cieľový port (SrcPort, DstPort) a hodnotu *Ingress interface* ktorú nepoznáme a teda nebude používaná v tomto projekte.

3 Návrh a implementácia

Program pozostáva z jedného súboru ktorý obsahuje jak potrebné datové štruktúry tak aj samotné telo algoritmu.

3.1 Štruktúy

Program pracuje celkovo so šiestimi štruktúrami. Tri slúžia pre získanie informácií z packetov(sniff_ip, sniff_tcp, sniff_udp, pre protokol icmp nebolo potrebné vytvárať štruktúru). Tieto štruktúry boli prevzaté z tohto tutoriálu . Obsahujú atributy ako by mali obsahovať bežné protokoly (teda sniff_ip má atributy: verziu, velikosť, tos,...) . Štvrtá slúži ako hlava linked-listu (zoznam) a piata je uzlom v liste. Šiestou je NetFlow v5 datagram(zlúčený Table B-3 a B-4) ktorý slúži pre export nazbieraných dát o flowe(názov štruktúry *NetFlow*).

3.2 linked-list

Obsahuje základné operácie:

- *init_linked_list()* – inicializácia listu,
- *insert_linked_list()* – vloženie prvku na začiatok zoznamu,
- *remove_node()* – odstránenie konkrétneho uzlu,
- *dispose_linked_list()* – odstránenie všetkých prvkov v zozname,
- *print_linkedList()* – ladiaci výpis celého zoznamu

Samotný uzol obsahuje protokol, zdrojovú a finálnu ip adresu, zdrojový a finálny port, Tos, čas príchodu prvého a posledného packetu ktorý je uložený ako *int* a môže dochádzať k nepresnostiam na nanosekundy, veľkosť zabranej pamäte, počet packetov, číslo daného flowvu a flagy protkolu tcp.

3.3 main

Načítanie vstupných argumentov je realizované cez funkciu *getopt()*, ktorá ich patrične prerozdelí.V prípade, že bola zadaná adresa s portom na ktorú majú byť odoslané záznamy dôjde k oddeleniu portu od adresy do premenných *port* a *adress* s ktorími sa pracuje neskôr vo funkcii *send_flow()*. Následne sa inicializuje linked list kde sa budú ukladať flowy packetov. Nasleduje cyklus v ktorom sa postupne prechádza vstupný súbor a cez funkciu *pcap_next()* sa získavajú jednotlivé pakety. Pokiaľ nedôjde k vráteniu packetu, cyklus končí, flowy ktoré neboli doposiaľ odoslané sa odošlú a program sa ukončuje. Telo cyklu sa skladá z funkcie

got_packet() ktorá je hlavnou častou programu.

3.4 got_packet

Nazačiatku dôjde k definícií ip hlavičky. Následne sa prejde linked-list pomocov cyklu a skontrolujú sa časy flowov v prípade, že niektorý z nich presahuje požadovanú časovú hranicu *-a* alebo *-i*, je odoslaný na Netflow kolektor, počet zabraných flowov v cachy(*flow_cache*) sa zmenší a odstráni sa zo zoznamu. Potom sa cez *switch* zistí protokol nového packetu a podľa typu protokolu sa zistí SrcPort, DstPort a flagy. Nasleduje cyklus ktorý prejde zoznam opäť a pokúsi sa nájsť flow kde by packet mohol patriť. Ak ho nájde dôjde k inkrementácií počtu packetov vo flowe, spracovaniu falgov a k navýšeniu d0ctest. V opačnom prípade ak packet nemá príslušný flow, skontroluje sa *flow_cache* a ak nieje prekročená pamäť vytvorí sa mu nový flow inak by došlo k odoslaniu a zmazaniu najstaršieho flowu aby uvoľnil miesto novému.

3.5 send_flow

Na začiatku dôjde k nahratiu dát o flowe do štruktúry *Netflow* (SysUptime je počítaný ako rozdiel času príchodu prvého packetu a najnovšieho v programe, unix_sec je nastavený ako čas najnovšieho packetu ktorý prišiel, unix_nsec je nastavený rovnako ako unix_sec akurát je použitá hodnota v unix milisekundách a je vynásobená 1000, hodnoty ktoré niesu známe sú nastavené ako 0). Ešte pred odosielaním dát dôjde k nahratiu celej štruktúry do bufferu ktorý sa odosiela na zadaný server. Postup odosielania dát bol prevzatý zo súboru echo-udp-client2.c vytvorený docentom Petrom Matúškom.

4 Návod na použitie

Po preložení súborom makefile cez príkaz make je vytvorený spustitelný program flow.
./flow [-f <file>] [-c <netflow_collector>[: <port >]] [-a <active_timer>] [-i <inactive_timer>] [-m <count >]

4.1 Argumenty a Parametre

-f <file>meno analyzovaného súboru v prípade, že nie je zadaný parameter –f vstup sa očkováva zo STDIN,
-c <neflow_collector:port>IP adresa alebo hostname NetFlow kolektoru pokiaľ nie je zadaný parameter –c tak IP je nastavená na 127.0.0.1:2055,
-a <active_timer>doba v sekundách po ktorej budú aktívne záznamy exportované na kolektor v prípade, že nieje uvedený parameter –a, je active_timer nastavený na 60s,
-i <seconds>doba v sekundách od príchodu posledného packetu ktorá ak bude prekročená tak budú neaktívne záznamy exportované na kolektor, ak nie je zadaný parameter –i východzia hodnota je nastavená na 10s,
-m <count>- velikost flow-cache. Pri dosiahnutí max. velikosti dôjde k exportu najstaršieho záznamu v cachi na kolektor ak nieje -m zadané, je použitá východzia hodnota 1024
-h zobrazí nápovedu.

Parametre je možné zadávať v ľubovoľnom poradí a sú brané ako voliteľné v prípade, že nebude niektorý z nich zadaný bude použitá východzia hodnota.

5 Literatúra

Netflow datagram: https://www.cisco.com/c/en/us/td/docs/net_mgmt/netflow_collection_engine/3-6/user/guide/format.html#wp1006108
Štruktúry: https://www.tcpdump.org/pcap.html
Odosielanie dát na server: https://moodle.vut.cz/pluginfile.php/502893/mod_folder/content/0/tcp/echo-server2.c?forcedownload=1