

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

Fakulta informačních technologií 2019 / 2020

IPK Projekt 2 Dokumentácia

Varianta ZETA- Sniffer paketov

Adam Žitňanský - xzitna02

Obsah

1	Úvod do problematiky				
	1.1	Packet sniffer			
	1.2	TCP			
	1.3	UDP			
2	Implementácia				
	2.1	Jazyk implementácie a použité nástroje			
	2.2	Podporované protokoly			
	2.3	Odchytávanie paketov			
	2.4	Filtrovanie paketov			
	2.5	Výpis paketu			
	2.6	Správanie v prípade chýb			
3	Testovanie				
4	Ref	Referencie			

1 Úvod do problematiky

1.1 Packet sniffer

Analyzátor paketov, tiež známý ako paketový sniffer, je počítačový program pomocou kterého je možné zachytávat a zaznamenávat komunikáciu v počítačovej sieti nebo v jej časti. V okamihu, keď dátové prúdy tečú cez sieť, tak analyzátor paketov zachytáva každý paket. Ten je potom v prípade potreby dekódovaný na surové dáta paketu, ktoré ukazujú hodnoty rôznych polí v pakete. Ďalej je analyzovaný jeho obsah na základe príslušnej RFC prípadne inej špecifikácie. [1].

1.2 TCP

Transmission Control Protocol (TCP) je najpoužívanejším protokolom transportnej vrstvy v sade protokolov TCP/IP používaných v sieti Internet. Špecifikovaný bol v RFC 793. Použitím TCP môžu aplikacie na počítačoch prepojených do siete vytvoriť mezi sebou spojenie, cez ktoré môžu obojsmerne prenášať dáta. Protokol garantuje spolahlivé doručovanie a doručovanie v správnom poradí. TCP tiež umožňuje rozlišovať a rozdelovať dáta pre viac aplikácí (napr. webový server a emailový server) bežiacich na jednom počítači. [2]

Obrázek 1: TCP paket 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 Bitv zdrojový port cílový port 32 potvrzovací číslo 64 96 offset dat rezervováno příznaky okénko 128 Urgent Pointer kontrolní součet 160 volby (volitelné) 192 volby (pokračování) výplň (do 32) 224 data

[2]

1.3 UDP

UDP protokol je protokol transportnej vrstvy orientovaný na správy a bol uvedený v RFC 768. V sade protokolov Internetu poskytuje UDP velmi jednoduché rozhranie medzi sieťovou vrstvou pod a aplikačnou vrstvou nad. UDP neposkytuje žiadne záruky doručenia a odesilatelova UDP vrstva si u raz už odoslaných správ neudržuje žiadny stav. UDP iba pridáva kontrolné súčty a schopnosť roztrieďovať UDP pakety mezi viac aplikací bežiacich na jednom počítači. [3]

Obrázek 2: UDP paket

+	bity 0 - 15	16 - 31
0	zdrojový port	cílový port
32	délka	kontrolní součet
64	data	

[3]

2 Implementácia

2.1 Jazyk implementácie a použité nástroje

Pre implementáciu som zvolil jazyk C s použitím doporučenej knižnice libpcap a deklarácí štruktúr a konštánt v hlavičkových súboroch netinet GNU C Library ktoré boli využité pri parsovaní paketu.

2.2 Podporované protokoly

Packet sniffer bol implementovaný v "obmedzene" verzí z hladicka podpory jednotlivých sieťových protokolov tj. z hladiska transportnej vrstvy sú podporované protokoly TCP a UDP a z hľadiska sieťovej vrstvy protokol IP(Ipv4 a ipv6) a z hľadiska linovej vrstvy Ethernet + Linux cooked header.

2.3 Odchytávanie paketov

Pakety sú odchytávané v réží knižnice pcap kde po otvorení zariadenia na ktorom sa má ochytávať (zadané v argumentoch) je spustený pcap_loop ktorý zabezbečí zachytenie zadaného počtu paketov a zavolanie callback funkcie pre každý takto zachytený paket kde je následne prevádzané jeho spracovanie (tj. parsovanie a výpis obsahu).

2.4 Filtrovanie paketov

Filtrovanie packetov na základe protokulu a portu je zabezpečené tak že sa podľa argumentov programu zostaví textový reťazec reprezentujúci tento filter podla syntaxe pcap, následne sa skompiluje (pcap_compile) a je apliovaný použítím pcap_setfilter nad daným zariadením.

2.5 Výpis paketu

Mimo závazných prvkov výpisu špecifikovaných v zadaní som výpisom porovnával s výpisom toolu tcpdump. Výpis paketu má nasledujúci formát:

```
ČAS SOURCE_IP: SOURCE_PORT > DEST_IP:DEST_PORT

počet_vypísaných_bajtov: výpis_bajtov_hexa výpis_bajtov_ASCII
```

Výpis bajtov hexa je vypisovaný po bajtoch oddelených medzerou a obsahuje výpis obsahu celého paketu. V ASCII výpise sú netlačíteľné znaky nahradené zankom '.' pričom o tlačitelnosti jednotlivých znakov sa rozhoduje na základe funkcie isprint.

2.6 Správanie v prípade chýb

V prípade že nastane chyba - (tj. napríklad chýba požadovaný argument, nepodarilo sa otvoriť zariadenie na ktorom sa majú zachytávať pakety) program vypíše na stderr informácie o chybe ktorá nastala a následne sa ukončí s nenulovým návratovým kódom. V prípade že sa jednalo o chýbajúci argument o rozhraní alebo zlyhalo otvorenie zariadenia, sú vypísané dostupné rozhrania. Vyhľadanie dostupných zariadení je realizované funkciou pcap_find_all_devs().

3 Testovanie

Program bol testovaný porovnaním príslušných častí výstupu s toolom tepdump. Testované boli kombinácie odchytávanie paketov na rôznych rozhraniach bez prídavných filtrov aj s nimi (podľa portu, prípadne štandardu - TCP aj UDP pakety. Z hľadiska prootkulu IP bola testovaná IPv4 aj IPv6.

Obrázek 3: IPv6 paket výstup ipk-snifferu

Obrázek 4: IPv6 paket výstup tcp-dump

Obrázek 5: IPv4 paket výstup ipk-snifferu

```
(base) [adam@localhost projekt2]$ sudo ./sniffer -i wlp4s0
14:56:28.582899 192.168.100.37 : 55514 > 162.159.136.234 : 443

0x0000 18 c5 8a fd f0 d4 44 85 00 f1 8e 34 08 00 45 00 .....D....4..E.
0x0010 00 55 e4 ca 40 00 40 06 05 81 c0 a8 64 25 a2 9f .U.@.@....d%..
0x0020 88 ea d8 da 01 bb ac 7e 33 a6 a7 2b e5 0e 50 18 .....~3.+..P.
0x0030 01 f5 53 8e 00 00 17 03 03 00 28 e5 1d e0 25 b4 ..s....(...%.
0x0040 7d 12 8b 09 92 7f 44 e8 6e cc a0 37 70 d7 35 93 }....Dn..7p.5.
0x0050 9f a0 05 82 a2 a4 b9 99 0b 35 49 le 7b 6d cb b4 ......51.{m..
0x0060 dc 89 2e ....
```

Obrázek 6: IPv4 paket výstup tcp- dump

4 Referencie

- [1] Wikipedia contributors. Analyzátor paketů Wikipedia, otevřená encyklopedie, 2020. https://cs.wikipedia.org/wiki/Analyztor_paket, [Pristupované online 2020-04-28].
- [2] Wikipedia contributors. Transmission control protocol Wikipedia, otevřená encyklopedie, 2020. https://cs.wikipedia.org/wiki/Transmission_Control_Protocol, [Pristupované online 2020-04-28].
- [3] Wikipedia contributors. User datagram protocol Wikipedia, otevřená encyklopedie, 2020. https://cs.wikipedia.org/wiki/User_Datagram_Protocol, [Pristupované online 2020-04-28].