IPK - Dokumentácia k snifferu paketov

Richard Seipel - xseipe00

April 2021

Obsah

1 Popis

Program zachytáva pakety na určenom rozhraní, zisťuje a vypisuje čas ich odoslania, adresu odosielateľa a prijímateľa, ich dĺžku a obsah. Zachytávané pakety je možné filtrovať alebo obmedziť ich počet.

2 Spustenie

2.1 Načítavané argumenty

Hlavný a povinný argument je rozhranie, na ktorom majú byť pakety zachytávané. Ďalej môže byť zadaný argument, udávajúci koľko paketov sa má zachytiť a ďalšie argumenty nastavujúce filtrovanie. Pri nezadaní počtu paketov je vypísaný jeden paket. Pri nezadaní rozhrania sú vypísané všetky dostupné rozhrania a program je ukončený.

2.2 Filtrovanie

Filter nastavuje, aké pakety majú byť zachytené. Napríkad podľa čísla portu alebo protokolu. Je zložený funkciou get_filter() podľa zadaných argumentov. Následne je skompilovaný funkciou pcap_compile() a nastavený funkciou pcap_setfilter() knižnice libpcap.

2.3 Spustenie

Zachytávanie paketov je realizované vo funkcii main, kde sú vytvorené základné štruktúry a načítané argumenty. Na vytvorenie pcap súboru je použitá podľa manuálu modernejšia funkcia pcap_create() a ďalej je nastavený promiskuitný mód a timeout na vypisovanie packetov z bufferu. Po aktivovaní súboru je prípadne skompilovaný a nastavený filter. Následne je spustené samotné zachtyávanie funkciou pcap_loop(), ktorej trvanie záleží od hodnoty argumentu pre počet zachytených paketov. Parametrom tejto funkcie je obslužná funkcia print_packet_info(), ktorá sa priebežne stará o formátovaný výpis zachytených paketov. Pred ukončením programu je ešte uvoľnená pamäť pre filter a uzavretý pcap súbor.

3 Výpis paketu

3.1 Prečítanie adries a portov

Funkcia print_packet_info() najprv zistí z ethernetovej hlavičky nasledujúcu hlavičku a následne podľa nej správnymi odsadeniami v pakete vyhľadá adresu odosielateľa a prijímateľa, prípadne aj porty na ktorých komunikácia prebieha. Tieto dáta sú ukladané do štruktúry header_data. Táto štruktúra je ďalej použitá vo volanej funkcii print_packet_header() pre zostavenie a vypísanie hlavičky paketu a na vypísanie jeho formátovaného obsahu slúži posledná volaná funkcia print_packet().

3.2 Formátovanie hlavičky

Čas je získaný vo funkcii get_time() z hlavičky paketu vo forme epoch času a následne formátovaný podľa normy RFC3339. Rovnako z hlavičky je získaná dĺžka paketu. Následne sú tieto dáta vrátane dát štruktúry header_data správne formátované a vypísané.

3.3 Formátovanie obsahu paketu

Obsah paketu je rozdelený do 16 bajtových blokov, začínajúcich offsetom bloku, následne jeho hexadecimálnym zápisom a nakoniec zápisom v ascii s nečitateľnými znakmi nahradenými bodkou.

4 Testovanie

Program bol testovaný po celý dobu vývoja súčasným spúštaním programu wireshark a následným porovnávaním oboch výstupov. Tento program mi taktiež pomáhal pochopiť zloženie rôznych paketov pri návrhu riešenia.