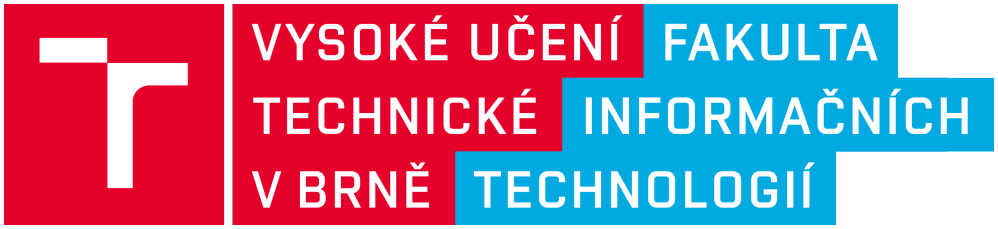

Projektová dokumentácia

Monitorování DHCP komunikace

ISA - Síťové aplikace a správa sítí

20. listopadu 2023 Adrián Bobola (xbobol00)

Obsah

1 Úvod

Témou projektu bolo vytvoriť program, ktorý umožní získať štatistiku vyťaženia sieťového prefixu z pohľadu počtu alokovaných IP adries.

Pri spustení je potrebné definovať sieťové prefixy, pre ktoré chceme generovať štatistiku vyťaženia. Program následne vytvára štatistiku z pcap súboru, prípadne zachytáva online komunikáciu na zadanom sieťovom rozhraní a vypisuje ju v spustenom okne aplikácie. Pri prekročení 50% vyťaženia nejakého z monitorovaných prefixov dôjde k zápisu tejto události do syslogu.

2 Návrh a popis implementácie

DHCP je sieťový protokol, ktorý umožňuje zariadeniam v sieti dynamicky získať konfiguračné informácie - napr. IP, maska, gateway.

Klient odošle `DHCP REQUEST` broadcastom na lokálnu sieť.
Následne servre odpovedajú klientovi správou `DHCP OFFER`, ktorá obsahuje možnú pridelenú IP.
Klient si zvolí požadovaný DHCP server a odpovedá mu správnou `DHCP REQUEST`.
Vybraný DHCP server potvrdzuje priradenie klienta správou `DHCP ACK`.

Pre potreby tejto aplikácie nám tak postačuje analyzovať správu `DHCP ACK`.
Celá aplikácia je naimplementovaná v súbore `dhcp-stats.cpp`.
V tejto kapitole nájdete stručný popis jednotlivých časti implementácie.

2.1 Štart programu, funkcia int main()

Jedná sa o automaticky spúšťanú funkciu po spustení aplikácie.
Po jej spustení dôjde k inicializácii balíčkov knihovne `ncurses`[4] a k vytvoreniu nového okna pre výstup programu, v ktorom sa budú zobrazovať štatistiky vyťaženia.
Po inicializácii okna aplikácie dôjde k spracúvaniu vstupných parametrov pomocou pomocnej funkcie `parse_params()` viď 2.2.

Následne dôjde k pokusu o načítanie uloženého .pcap súboru so zachytenými paketmi pomocou funkcie `pcap_open_offline`[5], prípadne začne aplikácia zachytávať pakety zo zadaného sieťového rozhrania zo vstupných parametrov pomocou funkcie `pcap_open_live`[5].
Pre online zachytávanie paketov som zvolil nastavenie `"promiskuitného režimu"` na vypnuté, pretože nepredpokladám potrebu zachytávať pakety v sieti, ktoré nepatria konkrétnemu zariadeniu. Parameter timeout `"to_ms"` je nastavený na hodnotu 1000 ms.

Pre aplikáciu je nastavená filtrácia paketov na protokol `UDP` a port `67`. Ostatné protokoly a porty sa zahadzujú. Protokol UDP a port 67 je typickým označením DHCP servera podľa RFC 2131[2], preto bol použitý v tomto projekte. Ďalšie filtre na pakety sa aplikujú vo funkcii `parse_packet()`, detaily v kapitole 2.3.

Následuje funkcia `pcap_loop()`[5], ktorá predá načítaný paket ďalšej funkcii `parse_paket()`, viď 2.3. Po spracúvaní všetkých zachytených paketov zo súboru dôjde k ukončeniu čítania súboru/online zachytávania paketov pomocou funkcie `pcap_close()`[5].

2.2 Načítanie vstupných argumentov, funkcia parse_params()

Funkcia načíta zadané argumenty pri spustení programu. K tomu využíva pomocnú funkciu `getopt()`[3].

Pre úspešne spustenie programu musí byť zadaný práve jeden parameter - parameter „-r“ alebo parameter „-i“.

Po zadaní IP/Prefixu dôjde ku kontrole či je daná IP adresa validná. To zabezpečuje funkcia `inet_addr()`. Taktiež dochádza ku kontrole, či je prefix z rozsahu 0 - 32.

2.3 Spracúvanie paketov, funkcia packet_parser()

Do funkcie vstupujú pakety splňujúce požiadavku na `UDP` protokol a port `67`.
Na začiatku sa kontroluje položka `options` v DHCP pakete.
Ak je nájdený `option 53 - DHCP Message Type`[1], dôjde ku kontrole či sa jedná o `DHCPACK` paket.
Ak to nie je `DHCPACK` paket, tak pre ďalšie spracúvanie nie je daný paket relevantný.

Pokiaľ sa jedná o `DHCPACK` paket, dôjde ku kontrole `yiaddr` paketu. Hodnota `yiaddr` sa nesmie rovať adrese `0.0.0.0`. Ak by sa hodnota `yiaddr` rovnala adrese `0.0.0.0` znamenalo by to, že sa jedná o odpoveď na `DHCPINFROM`, čo tiež nechceme započítavať v štatistikách.

Ak sa daný paket úspešne dostal cez všetky overenia tak nás zaujíma, či náhodou danú `yiaddr` adresu už nemáme uloženú z predchádzajúcich paketov. Týmto zamedzíme vzniku duplicít a nesprávnemu počítaniu vyťaženia daného prefixu.

Ak daná adresa ešte nie je započítaná v štatistike vyťaženia prefixu a zatiaľ spĺňa všetky potrebné požiadavky paketu, dôjde k jej uloženiu a volaniu pomocnej funkcie `print_statistics()`.
Daná funkcia vyhodnotí `yiaddr` adresu z paketu viď 2.4.

2.4 Vytváranie štatistiky vyťaženia prefixu, funkcia print_statistics()

Funkcia po zavolaní prechádza všetky IP adresy s prefixom zadané v argumente pri spustení programu. Na začiatku dôjde k resetovaniu štatistiky pred výpočtom nových hodnôt.
Program si vypočíta `masku subnetu` zadanej IP adresy a prislušného prefixu. Taktiež si vypočíta `broadcast adresu` a `adresu siete`.

Následne dôjde k prechádzaniu všetkých kandidátov adries zo získaných paketov. Na obe adresy sa aplikuje `maska subnetu` a dôjde k porovnaniu výsledných hodnôt.
Ak sa výsledné hodnoty zhodujú znamená to, že daná IP adresa patrí do zadaného prefixu.
Je nutné ešte overiť, či sa náhodou nejedná o IP `adresu siete` alebo `adresu broadcastu`.

Ak adresa paktu splnila všetky požiadavky, môže byť započítaná v štatistike vyťaženia daného sieťového prefixu. Zvýši sa počet alokovaných adries o 1 a prepočíta sa hodnota vyťaženia prefixu.

Následne dôjde k aktualizácii údajov vypísaných v okne aplikácie. Po aktualizácii výpisu dochádza ešte ku kontrole hodnoty vyťaženia daného prefixu. Ak je vyťaženie daného prefixu väčšie ako hodnota 50.00, dôjde k výpisu tejto informácie do okna a k zápisu informácie do `syslog`. Pri implementácii bola zvolená syslog správa typu `LOG_NOTICE`[6].

3 Návod na použitie

3.1 Kompilácia programu

Pre kompiláciu je potrebné zadať príkaz `make`. Po úspešnej kompilácii dôjde k vytvoreniu spustiteľného súboru `dhcp-stats`.
Aplikácia je písaná v jazyku C++ a požaduje std=C++17.

3.2 Spúšťanie programu

Aplikáciu je možné spustiť pomocou príkazu:

```
./dhcp-stats [-r <file-name>] [-i <interface-name>] [<ip-prefix>]
```

- `-r <file-name>`
  Meno pcap súboru, z ktorého sa bude vytvárať štatistika.
  Pre uzavreie okna aplikácie stačí stlačiť ľubovoľnú klávesu.
  Bez zadaného parametra je povinné použiť parameter -i.

- `-i <interface-name>`
  Názov sieťového rozhrania, na ktorom bude aplikácia naslúchať a z neho vytvárať štatistiku.
  Uzavreie okna aplikácie je potrebné signálom SIGINT, napr. „Ctrl+C“.
  Bez zadaného parametra je povinné použiť parameter -r.

- `<ip-prefix>`
  Rozsah siete, pre ktorý sa bude vytvárať štatistika. Požaduje sa zadávať v tvare `IP/prefix`.
  V prípade použitia viac prefixov je nutné jednotlivé adresy oddeľovať medzerou.
  Príklad: 192.168.0.1/28 192.168.0.1/20 172.1.1.1/24

4 Záver

Implementované bolo celé zadanie projektu bez rozšírení.
Pri testovaní projektu som nezaznamenal žiadne problémy ani chyby.

Testovanie aplikácie pre generovanie štatistiky zo sieťového rozhrania som testoval pomocou zachytávania paketov rozhrania localhost a odosielania paketov pomocou nástroja `tcpreplay` na localhost.

Kompilácia a spustiteľnosť aplikácia bola testovaná na školskom serveri Merlin.fit.vutbr.cz.

5 Použitá literatúra

[1] DHCP Options and BOOTP Vendor Extensions, RFC-2132. [online], [vid. 2023-11-20]. Dostupné z: https://datatracker.ietf.org/doc/html/rfc2132

[2] Dynamic Host Configuration Protocol, RFC-2131. [online], [vid. 2023-11-20]. Dostupné z: https://datatracker.ietf.org/doc/html/rfc2131

[3] getopt(3) — Linux manual page. [online], [vid. 2023-11-20]. Dostupné z: https://man7.org/linux/man-pages/man3/getopt.3.html

[4] NCURSES Programming HOWTO. [online], [vid. 2023-11-20]. Dostupné z: https://tldp.org/HOWTO/NCURSES-Programming-HOWTO/

[5] PCAP(3PCAP) MAN PAGE. [online], [vid. 2023-11-20]. Dostupné z: https://www.tcpdump.org/manpages/pcap.3pcap.html

[6] syslog(3) — Linux manual page. [online], [vid. 2023-11-20]. Dostupné z: https://man7.org/linux/man-pages/man3/syslog.3.html