

Programování síťové služby

DHCP server

Obsah

1. Teoretický úvod.....	3
2. Návrh aplikácie.....	3
3. Popis implementácie.....	4
4. Použitie.....	5
Zoznam použitej literatúry.....	6

1. Teoretický úvod

Protokol DHCP (Dynamic Host Configuration Protocol - protokol dynamického pridelovania adries) je štandardný sieťový protokol definovaný v RFC 2131. Slúži na dynamické pridelovanie IP adries v sieti, zároveň klientom odovzdáva ďalšie konfiguračné informácie.

DHCP je postavený na modeli klient-server, kde server prideluje klientom IP adresy a distribuuje konfiguračné parametre. Klient a server komunikujú prostredníctvom DHCP správ, schéma ich presného formátu je uvedená v RFC 2131 (1). Pri komunikácii sa využívajú nasledujúce druhy DHCP správ:

DHCPDISCOVER – Klient zasiela správu na broadcast aby lokalizoval servery v sieti.

DHCPOFFER - Server odpovedá na DHCPDISCOVER a poskytuje konfiguračné parametre siete.

DHCPREQUEST – Klient žiada parametre od jedného zo serverov a zároveň odmieta ponuky všetkých ostatných.

DHCPACK – Server posiela klientovi vyžiadané konfiguračné parametre spolu s pridelenou IP adresou.

DHCPNAK – Server oznamuje klientovi že jeho dáta o seti nie sú správne.

DHCPRELEASE – Klient oznamuje serveru že sa vzdáva pridelenej adresy.

Keďže ide o implementáciu jednoduchého DHCP servera správy typu DHCPDECLINE a DHCPINFORM nie sú podporované.

2. Návrh aplikácie

Aplikácia sa dá z pohľadu návrhu rozdeliť do troch základných častí.

Prvou je nekonečný cyklus servera, ktorý načúva na štandardnom porte, zachytáva správy z broadcastu a odosiela vytvorené odpovede. Táto časť je samozrejme spojená s tvorbou socketov a nastavovaním ich parametrov.

Druhou časťou je analýza prijatých DHCP správ, tvorba odpovedí a riadenie reakcií servera na vzniknuté situácie. Špeciálnou podmnožinou tejto časti je spracovávanie a zapisovanie sekcie

„options“, ktorá je špecifická svojou variabilitou.

Poslednou zo základných častí návrhu je správa IP adries. Tu je potrebné analyzovať parametre siete zo vstupných argumentov programu – adresa siete, maska, prefix, použiteľný rozsah a evidovať stav pridelených adries.

Nevyhnutné je tu rozlíšiť tri stavy ktoré môžu pre konkrétnu IP adresu nastať:

- reserved – adresa je rezervovaná pre určitého klienta (prebieha komunikácia o pridelení)
- assigned – adresa je pridelená niektorému z klientov
- free/expired – adresa je voľná na použitie (nebola vôbec alokovaná alebo uvoľnená)

Samozrejme je potrebné kontrolovať či požadovaná adresa patrí do siete, alebo nedošlo k vyčerpaniu všetkých použiteľných adries. V tomto prípade sa ponúka využiť objektovo orientovaný prístup a zodpovednosť za správu adries prideliť jedinému objektu, ktorý bude ponúkať množinu jednoduchých metód na zmenu stavu adries a získanie voľnej adresy.

3. Popis implementácie

Jadrom programu je modul `dhserver`. Okrem nekonečného cyklu, v ktorom server zachytáva správy klientov obsahuje tento modul funkcie na odosielanie správ na broadcast respektíve unicast v prípade že ide o obnovenie pridelennej adresy pred vypršaním platnosti a spracovanie argumentov programu.

Ďalej sa tu nachádza funkcia na získanie všetkých potrebných dát po prijatí správy a konštrukciu odpovede s výnimkou spracovávania „options“, ktoré sa nachádza v samostatnom moduli nazvanom `options`.

Proces analýzy prijatej správy a tvorby odpovede je do značnej miery paralelný. V priebehu analýzy dochádza zároveň k zapisovaniu do bufferu na odoslanie a ku kopírovaniu určitých dát prijatej správy.

Keďže sekcia „options“ môže mať variabilné poradie jednotlivých položiek na jej spracovanie a vytváranie je využitý jednosmerne viazaný zoznam obohatený o funkcie na vytvorenie na základe bufferu bytov a zapísanie obsahu do bufferu.

Na účel práce so štruktúrou protokolu a zasielaním/prímaním správ formou jednoduchého pola

bytov je veľmi praktická práca s ukazateľmi jazyka C. Do pola bytov sa zapisujú jednotlivé položky prostredníctvom ukazateľa na štruktúru **struct bootp**, čím je zaručený ich zápis na správny ofset v poli bez nutnosti presúvania v pamäti pred odoslaním po sieti.

Server je ukončený po prijatí signálu SIGINT, z toho dôvodu bolo potrebné ošetriť správne uvoľnenie všetkých alokovaných zdrojov. Po zachytení signálu SIGINT je zavolaná obslužná funkcia, ktorá prestaví hodnotu globálnej premennej **quit**, táto zmena spôsobí ukončenie cyklu servera. Následne dôjde ku korektnému uvoľneniu všetkých alokovaných zdrojov a program skončí s návratovým kódom 0.

Správu IP adries implementuje trieda **AddressPool** z modulu **address_pool**. Objekt tejto triedy uchováva dôležité dáta – adresu siete, masku, broadcastovú adresu a vektor alokovaných adries a poskytuje množinu atomických operácií na zmeny a zisťovanie stavu jednotlivých adries.

Alokovanie pamäte pre všetky použiteľné adresy v sieti by mohlo mať za následok veľké plitvanie pamäťou pri použití kratšieho prefixu a malého počtu klientov v sieti. Z tohoto dôvodu je pamäť pre adresu alokovaná až v prípade že je adresa naozaj vyžiadaná na pridelenie. Použiteľné adresy pre ktoré záznam nie je vytvorený sú implicitne v stave free.

Dôležité konštanty typov DHCP správ a options sa nachádzajú v hlavičkových súboroch **constants.hh** a **message_codes.hh**.

4. Použitie

Spustiteľný program sa vytvorí príkazom **make**.

Príklad spustenia:

```
./dserver -p 10.0.0.0/24 [-e 10.0.0.2,10.0.0.5]
```

Parametre:

- **-p <ip_adresa/maska>** rozsah prideloovaných IP adries
- **-e <ip_adresy>** rezervované IP adresy ktoré sa nepridelujú oddelené čiarkou (voliteľný parameter)

Pri spustení programu sa predpokladá že stroj na ktorom server beží má v sieti pridelenú prvú použiteľnú IP adresu z rozsahu zadaného parametrom -p.

Zoznam použitej literatúry

- [1] <https://tools.ietf.org/html/rfc2131>
- [2] <https://tools.ietf.org/html/rfc951>