DNS Resolver

Tomáš Sasák

October 24, 2019

Contents

1 Úvod do problematiky

Zadanie je nasledujúce, implementujte program DNS, ktorý bude zasielať dotazy na DNS servery a včitatelnej podobe vypisovať prijaté odpovede od daného DNS servera na štandartný výstup. Zostavenie a analýza DNS paketov musí byť implementovaná priamo v programe. Stačí považovať iba UDP komunikáciu.

1.1 DNS

Domain name systém (DNS), je systém, ktorý ukladá prístup k informácií o názve stroja a názve domény v istej databáze. Najdôležitejšie je, že poskytuje mechanizmus získania IP adresy pre každé meno stroja a naopak. DNS poskytuje dôležitú službu, pretože kým počítače a sieťový hardware pracujú s IP adresami, ľudia si lahšie pamatajú mená strojov a domém pri ich používaní. DNS tvojrí prostredníka medzi človekom a strojom.

1.2 DNS packet

DNS packet sa skladá z nasledujúcich častí

- Header (hlavička)
- Question (otázka)
- Answer (odpoveď)
- Authority (autorizovaná odpoveď)
- Additional (naviac odpovede)

1.2.1 Header (hlavička)

Táto časť má velkosť 12B a skladá sa z nasledujúcich častí

- `ID` - identifikačné číslo packetu (2B)
- `QR` - flag identifikujúci či sa jedná o otázku alebo odpoveď (1b)
- `OPCODE` - označuje variantu balíku (4b)
- `TC` - flag identifikujúci poškodený balík (1b)
- `RD` - flag identifikujúci či je vyžiadaná rekurzia (1b)
- `Z` - rezerované miesto (1b)
- `RA` - flag identifikujúci či je server dokáže vykonať rekurziu (1b)
- `QDCOUNT` - číslo identifikujúce počet otázok (2B)
- `ANCOUNT` - číslo identifikujúce počet odpovedí (2B)

- `NSCOUNT` - číslo identifikujúce počet autorizovaných odpovedí (2B)
- `ARCOUNT` - číslo identifikujúce počet naviac odpovedí (2B)

Hlavička ma vždy pevnú veľkosť a je súčasťou každého DNS packetu.

1.2.2 Question (otázka)

Táto časť ma premennú velkosť a je súčasťou každého DNS packetu. Skladá sa z následovných častí

- `NAME` - meno domény, ktorá má byť preložená
- `TYPE` - typ záznamu
- `CLASS` - trieda komunikácie

NAME Meno domény alebo IP adresa (pri reverznom DNS vyhľadávaní), ktorá musí byť podla normy rozdelená podla znaku "." (bodka), na dané štítky (labels). V časti otázka (question), sa ešte musí pridať pred každý štítok (label) pridať číslo, ktoré označuje kolko znakov obsahuje daný štítok.

TYPE Typ záznamu, ktorých je mnoho. V tomto projekte sú najviac používané

- `A` - záznam obsahujúci IPv4 adresu
- `AAAA` - záznam obsahujúci IPv6 adresu
- `PTR` - ukazateľ (pointer) na alias

CLASS Trieda komunikácie. v tomto projekte iba

- `IN` - komunikácia Internet

Tieto časti tvoria jednu DNS otázku (question), samozrejme otázok môže byť viac a takto by sa časti opakovali. Poznamenať treba, že ak sa jedná o reverznú otázku, v časti `NAME` je potrené danú adresu rozdeliť na štítky (labels) a pridať nové štítky ktoré značia že ide o reverznú otázku. Pri adresách IPv4, je adresa rozdelená podobne ako doména (čiže podla znaku ".") a štítky (labels) sú uložené pospiatočky, nakoniec su pridané 2 štítky, `in-addr` a `arpa`, vďaka ktorým, sa doména reverzne jednoduchšie vyhľadáva. Pri adresách typu IPv6, je adresa prevedená do dlhej podoby a každý hexadecimálny člen značí 1 štítok (label), nakoniec sú pridané 2 štítky (labels) `ip6` a `arpa`.

1.2.3 Answer (odpoved)

Reprezentuje odpoveď DNS serveru. Táto časť ma premennú velkosť. Je podobná otázke (question). Skladá sa z následujúcich častí.

- `NAME` - identické ako pri otázke (question)
- `TYPE` - identické ako pri otázke (question)
- `CLASS` - identické ako pri otázke (question)
- `TTL` - time-to-live, dĺžka platnosti odpovede (používané pre caching)
- `DL` - dĺžka následujúceho záznamu, ktorý následuje (záznamy rovnaké ako pri question)
- `dáta` - dáta záznamu

Ako v predchádzajúcej časti, tak aj odpovedí (answer) môže byť viac ako 1.

NAME Je dôležité poznamenať, že odpovede od DNS serveru, môžu byť komprimované, to znamená že namiesto značiek (labels), ktoré sa opakujú, je číselne daný offset od začiatku paketu kde sa daná značka nachádza po prvý krát. Tento príznak je naznačený tak, že na mieste prvých dvoch bitov bytu, ktorý označuje dĺžku labelu, sa nachádzajú bity 11.

TYPE V tejto sekcií je nutné poznamenať že pri odpovediach je program prispôsobiť rôznym typom odpovede. Tieto odpovede môžu byť rôzneho druhum, program dokáže rozpoznať tieto typy odpovedí

- `A` *
- `NS` *
- `MD` *
- `MF` *
- `CNAME` *
- `SOA` *
- `MB` *
- `MG` *
- `MR`
- `NULL`
- `WKS`

- `PTR` *
- `HINFO`
- `MINFO` *
- `MX` *
- `TXT`
- `SRV`
- `AAAA` *

Ale program dokáže spracovať obsah záznamov označených "*".

1.2.4 Authoritative Answer (autorizovaná odpoveď)

Reprezentuje odpoveď DNS serveru, ktorý je autorizovaný. Takýto DNS server obsahuje skutočné záznamy domén a IP adries, z ktorých je odpoveď vytvorená. Jej časti sú identické ako pri odpovedi (answer). Má premennú velkosť.

1.2.5 Additional (naviac odpovede)

Táto sekcia reprezentuje záznamy, ktoré priamo nemusia byť odpoveďou na otázku (question), ale môžu mať s odpoveďou niečo spoločné. Formát je rovnaký ako pri odpovedi (answer).

2 Implementácia

Všetká komunikácia prebieha pomocou protokolu UDP. Implementácia DNS resolveru sa skladá z nasledujúcich tried

- `Arguments`
- `DnsSender`
- `DnsParser`

2.1 Trieda Arguments

Trieda obsahuje metódu pre spracovanie vstupných argumentov, a obsahuje premenné ktoré vlastnia hodnotu argumentu.

Obsahuje následujúce metódy

- `parse_arguments` - metóda spracuje argumenty a inicializuje inštanciu triedy

A nasledujúce premenné

- `recursionDesired` - bool značiaci vyžiadanú rekurziu (parameter -r)
- `reverseQuery` - bool značiaci vyžiadanú reverznú otázku
- `ipv6` - bool značiaci či je požadovaný záznam AAAA (IPv6 adresa)
- `dnsServer` - reťazec, obsahujúci adresu/doménu DNS serveru
- `port` - číslo, obsahujúce port na ktorý je DNS paket odoslaný (štandartne 53)
- `target` - reťazec, obsahujúci doménu/adresu, ktorá je prekladaná

Spracovanie argumentov je implementované pomocou vstavanej funkcie `getopt`.

2.2 Trieda DnsSender

Trieda vytvára DNS paket obsahujúci otázku, vytvorí si daný socket, odošle tento packet na daný DNS server a príjme odpoveď.

Obsahuje následujúce metódy

- `send_query`
- `set_dns_socket`
- `create_dns_packet`
- `split_target`

A následujúce premenné

- `dnsSocket` - handle pre daný socket z ktoré je paket odoslaný a následne príjimaný

Metóda send_query Metóda ktorá odošle DNS paket na daný server a príjme odpoveď. (Skladá sa z nasledujúcich metód)

Metóda set_dns_socket Metóda zistí alebo overí IP adresu zadaného DNS serveru (prevencia problému so vajíčkom a sliepkou, pretože je treba pomocou DNS preložiť doménu DNS serveru na IP adresu), toto je vykonané vstavanou funkciou `getaddrinfo`, táto funckia naviac vracia správne nastavenia socketu pre komunikáciu s daným serverom, pomocou týchto nastavení sa vytvorí socket a nastaví sa. Pre prevenciu nekonečného čakania na odpoveď, ak by sa serveru niečo stalo alebo by vôbec nekomunikoval je použitá socket funkcia `connect`. Pretože protokol UDP je bezstavový a používa best-afford-delivery. Týmto sa dokáže predísť nekonečnému čakaniu na odpoveď.

Metóda create_dns_packet Metóda vytvorí DNS paket (typu otázka, query) ktorý obsahuje správne nastavenie podľa zadaných parametrov. Ak je požadovaný reverzný dotaz, je správnosť adresy skontrolovaná a podľa typu (IPv4/IPv6) je rozdelená na dané štítky a zabudovaná do hlavičky (štítky, tak ako je definované v teórii o štítkoch). Ak je zadaná doména, je rovnako rozdelená na štítky a zabudovaná, tak ako je popísané v teorií. Ďalšie časti paketu su nastavené podľa zadaného vstupu.

Metóda split_target Metóda je implementácia funkcie `explode()` z jazyku PHP. Podľa zadaného znaku, rozdelí reťazec na vektor tokenov a tento vektor vráti.

2.3 Trieda DnsParser

Trieda spracováva DNS packet odoslaný serverom ako odpoveď. A vypíše jeho obsah na štandartný výstup.

Obsahuje nasledujúce metódy

- `parse_dns_response`
- `parse_labels`
- `parse_answer`

Metóda parse_dns_response Metóda spracuje celý DNS packet. Metóda pretypováva packet a postupne sa po ňom posúva pomocou offsetov a volá nasledujúce pomocné funkcie. Medzi posúvaním po packete, vypisuje obsah packetu.

Metóda parse_labels Metóda ktorá slúži pre spracovanie štítkov (labels) a ich vypísanie na štandartný výstup, metóde je možné zadefinovať, či je v danej časti povolená komprimácia packetov (to znamená že sa v časti môže nachádzať pointer na štítok s daným offestom) metóda spracováva štítky až dokým nenarazí na 0x00 byte, čo znamená koniec štítkov. Ako už z popisu vyplýva, metóda taktiež dokáže spracovať komprimované štítky (labels) pomocou offsetov.

Metóda parse_answer Metóda ktorá slúži pre spracovanie posledných 3 častí DNS paketu (answer, autoritative a additional). Pretože forma týchto 3 častí je rovnaká. Metóda používa predchádzajúcu metódu `parse_labels` pre spracovanie štítkov (labels). Metóda podporuje spracovanie DNS záznamov uvedené v teórii (A, AAAA, CNAME, NS a PTR).

2.4 Hlavičkové súbory

Implementácia sa skladá z následujúcich hlavičkových súborov

- `dns_header`
- `dns_question`
- `dns_answer`
- `record_types`

Súbor dns_header Obsahuje reprezentáciu DNS hlavičky (header). Vzhľadom na to, že veľkosť hlavičky (header) je vždy pevná, je možné celú časť zapísať do štruktúry a jednoducho si packet pretypovať na danú časť. Obsahuje časti spomenuté v teorií.

Súbor dns_question Obsahuje pevné časti jednej DNS otázky (question), ktoré si kód, keď je treba pretypuje. Obsahuje položky uvedené v teorií.

Súbor dns_answer Obsahuje pevné časti jednej DNS z odpovedí (answer, authoritative a additional). Ako predtým, tak vzhľadom na to že obsahuje iba pevne dané časti odpovedi, kód si packet pretypováva podla potreby (napr. za záznamom, pred záznamom).

Súbor record_types Obsahuje makrá čísiel záznamov, pre lepšiu čitatelnosť kódu. Taktiež obsahuje makro pre maximálnu veľkosť DNS packetu.

Súbor soa_header Obsahuje poslednú časť, DNS záznamu typu `SOA`. Z tohoto vyplýva že táto časť je velkosťou nepremenná a preto môže byť definovaná ako štruktúra.

3 Spúšťanie programu

Prekladanie programu je pomocou Makefile a príkazu `make`. Spustenie je nasledujúce

```
./dns [-r] [-x] [-6] -s server [-p port] adresa
-r - požadovaná rekurzia
-x - reverzný dotaz
-6 - dotaz s záznamom AAAA (IPv6)
-s - doména/adresa DNS severu na ktorý je DNS paket odoslaný
-p - port na ktorý je DNS paket odoslaný
```

3.1 Príklad spustenia

```
$ ./dns -s kazi.fit.vutbr.cz www.google.com -r -p 53
Header section:
Type: Answer, Opcode: QUERY, Authorative answer: No, Trucanted: No,
Recursion desired: Yes, Recursion avaiable: Yes, Reply code: 0
Question section(1)
www.google.com, A, IN
Answer section(1)
www.google.com, A, IN, TTL: 300, 172.217.23.228
Authority section(0)
Additional section(0)

$ ./dns -s kazi.fit.vutbr.cz 172.217.23.228 -r -x
Header section:
Type: Answer, Opcode: QUERY, Authorative answer: No, Trucanted: No,
Recursion desired: Yes, Recursion avaiable: Yes, Reply code: 0
Question section(1)
228.23.217.172.in-addr.arpa, PTR, IN
Answer section(2)
228.23.217.172.in-addr.arpa, PTR, IN, TTL: 85558, prg03s06-in-f228.1e100.net
228.23.217.172.in-addr.arpa, PTR, IN, TTL: 85558, prg03s06-in-f4.1e100.net
Authority section(0)
Additional section(0)

$ ./dns -s 147.229.190.143 kazi.fit.vutbr.cz
Header section:
Type: Answer, Opcode: QUERY, Authorative answer: No, Trucanted: No,
Recursion desired: No, Recursion avaiable: Yes, Reply code: 0
Question section(1)
kazi.fit.vutbr.cz, A, IN
Answer section(1)
kazi.fit.vutbr.cz, A, IN, TTL: 4649, 147.229.8.12
Authority section(4)
```

```
fit.vutbr.cz, NS, IN, TTL: 3121, guta.fit.vutbr.cz
fit.vutbr.cz, NS, IN, TTL: 3121, rhino.cis.vutbr.cz
fit.vutbr.cz, NS, IN, TTL: 3121, gate.feec.vutbr.cz
fit.vutbr.cz, NS, IN, TTL: 3121, kazi.fit.vutbr.cz
Additional section(6)
gate.feec.vutbr.cz, A, IN, TTL: 4802, 147.229.71.10
gate.feec.vutbr.cz, AAAA, IN, TTL: 11102, 2001:67c:1220:9847::93e5:470a
guta.fit.vutbr.cz, A, IN, TTL: 316, 147.229.9.11
guta.fit.vutbr.cz, AAAA, IN, TTL: 316, 2001:67c:1220:809::93e5:90b
rhino.cis.vutbr.cz, A, IN, TTL: 3196, 147.229.3.10
rhino.cis.vutbr.cz, AAAA, IN, TTL: 3196, 2001:67c:1220:e000::93e5:30a
```

References

[1] RFC 1035 - DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION https://www.ietf.org/rfc/rfc1035

[2] RFC 2929 - Domain Name System (DNS) IANA Considerations https://tools.ietf.org/html/rfc2929

[3] RFC 3425 - Obsoleting IQUERY https://tools.ietf.org/html/rfc3425

[4] RFC 8501 - Reverse DNS in IPv6 for Internet Service Providers https://tools.ietf.org/html/rfc8501

[5] Domain name system - Wikipedia https://en.wikipedia.org/wiki/Domain_Name_System