VYSOKÉ UČENÍ TECHNICKÉ

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

IPK projekt č.2

DNS Lookup nástroj

Autor: Jan Kočí

Obsah

Úvod

Cílem projektu bylo vytvořit aplikaci realizující překlad mezi doménovými jmény a jejich IP adresami, pomocí dotazování DNS serveru s použitím linuxových soketů a protokolu UDP.

Spouštění

./ipk-lookup [-h] … vypiš nápovedu programu

./ipk-lookup -s server [-T timeout] [-t type] [-i] name

- h (help) - volitelný parametr, vypíše nápovědu programu
- s (server) - DNS server (IPv4 adresa), na který se budou odesílat dotazy
- T (timeout) - timeout pro dotaz v sekundách, výchozí hodnota 5 sekund
- t (type) - typ dotazovaného dotazu: A (výchozí), AAAA, PTR, CNAME, NS
- i (iterative) - vynucení iterativního způsobu rezoluce
- name - překládané doménové jméno nebo IP adresa (pokud type = PTR)

Pokud není zadán parametr -i, aplikace vykonává pouze rekurzivní dotazy.
Je-li parametr -i zadán, aplikace postupuje iterativním způsobem. Nejprve zašle dotaz, hledající NS server root serveru, následně dotaz pro IP adresu root serveru. Poté zasílá postupně dotazy pro jednotlivé části doménového jména, respektive IP adresy, kdy opět nejprve nalezne NS server dané části domény a další dotaz poté deleguje na tento nově nalezený NS server.

Návratová hodnota

Program vrací:
- hodnotu 0, v případě úspěšného ukončení,
- hodnotu 1, v případě výskytu chyby
- hodnotu 2, v případě špatně zadaných parametrů

Výstup programu

Program vypisuje na standartní výstup výsledky, popřípadě dílčí výsledky, překladu.
Vypsány jsou pouze záznamy podporovaných typů (A, AAAA, PTR, NS, CNAME).
Výsledek překladu je vypsán v následujícím formátu:

<name> [ ]+ IN [ ]+ <type> [ ]+ <answer> \n

Problematika DNS

Úkolem DNS serverů je realizovat vzájemný překlad (mapování?) mezi doménovými jmény a jejich IP adresami.

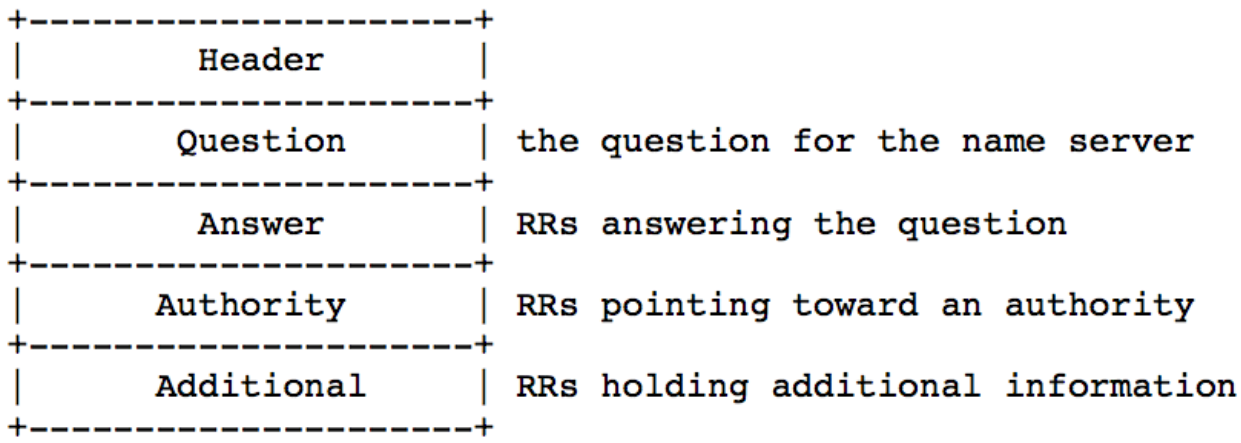
Hiearchie DNS serverů

DNS system využívá velké množství serverů, organizovaných do hierarchické struktury. Žádný DNS server tak neobsahuje všechna známá mapování adres, mapování jsou rozmístěna napříč všech DNS serverů.
Na nejvyšší úrovni hierarchie se nachází root server, pod ním se v hierarchii nachází TLD (Top Level Domain) servery a autoritativní servery.

Iterativní dotaz pak vypadá tak, že se nejdříve ptáme root serveru na IP adresu TLD serveru, poté kontaktujeme takto nalezený TLD server na IP adresu autoritativniho serveru a tak dále.

Struktura DNS zprávy

Struktrura DNS zprávy je stejná jak pro dotazy tak pro odpovědi. Na začátku zprávy se nachází hlavička (Header) a za ní již samotné DNS dotazy. V případě, že je zpráva odpověď, nachází se DNS odpovědi ihned za posledním dotazem.

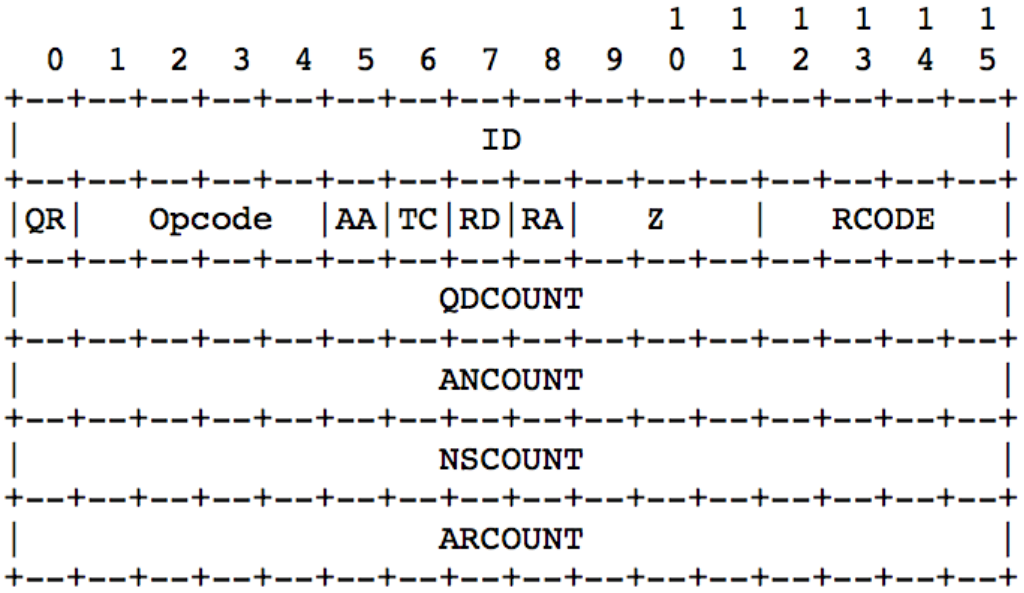


[1] struktura DNS zprávy

1.Hlavička

Hlavička je pevné velikosti 12B a obsahuje:

- ID = identifikátor zprávy
- Flags = pole bitových příznaků
- QDCOUNT = počet DNS dotazů, které zpráva obsahuje
- ANSCOUNT = počet DNS odpovědí (pokud je zpráva dotaz má hodnotu 0)
- NSCOUNT = počet odpovědí s autoritativními servery
- ARCOUNT = počet dodatečných odpovědí

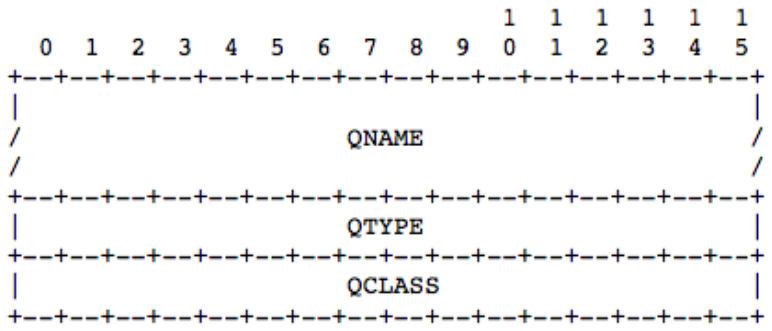


[2] struktura DNS hlavičky

Každý řádek představuje 16 bitů.

2. DNS dotaz

Struktura DNS dotazu je již proměnlivé velikosti.



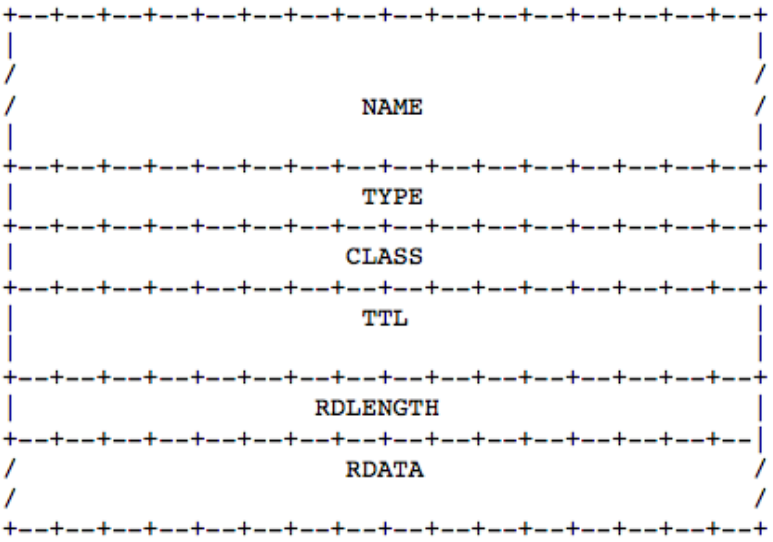
- QNAME = zakódované doménové jméno
- QTYPE = ty dotazu
- QCLASS = třída dotazu (IN pro Internet)

[3] struktura DNS

3.DNS záznam

DNS záznam představuje strukturu stejnou pro všchny typy odpovědí, ať už additional nebo authority. DNS záznamy jsou navíc přímo používány na serverech, kde mapují doménová jména na IP adresy.

- NAME = doménové jméno
- TYPE = specifikace typu dat
- CLASS = třída data v RDATA
- TTL = doba po kterou může být záznam v cachy než by měl být zničen
- RDLENGTH = velikost dat
- RDATA = data obsahující dané mapování



[4] struktura DNS záznamu

Formátování dat

Data, která ukládáme do DNS zprávy musejí být zakódována v přesně specifikovaném formátu.
Příklad zakódování:

- doménové jméno = tečky oddělující části jména jsou nahrazeny bajtem reprezenrujícím počet znaků v dané skupině.
  www.fit.vutbr.cz => zakóduje jako => 3www3fit5vutbr2cz

- IPv4 adresa = části adresy jsou obráceny a je přidána koncovka .in-addr.arpa
  128.43.24.2 => 2.24.43.128.in-addr.arpa
  - Poté je adresa zakódována stejným způsobem jako doménové jméno

- IPv6 adresa = je rozložena po jednotlivých číslovkách, ty jsou odděleny tečkami a nakonec je přidána koncovka .ip6.arpa
  2001:67c:1220:8b0::93e5:b013 =>
  3.1.0.b.5.e.3.9.0.0.0.0.0.0.0.0.b.8.0.0.2.2.1.c.7.6.0.1.0.0.2.ip6.arpa
  - Poté je adresa zakódována stejným způsobem jako doménové jméno

Komprese dat

Abychom redukovali velikost zpráv, používáme speciální reprezentace dat, umožňující jejich komplesi. Tato komprese je uplatňována například u odpovědí. Pokud odpověď obsahuje doménové jméno, je toto jméno již součástí zprávy a je tedy již ve zprávě zakódováno. V takovém případě bude odpověď obsahovat pouze ukazatel na již zakódované jméno.
Data tedy mohou být reprezentována jako:

- zakódovaná data
- ukazatel na zakódovaná data
- část zakódovaných dat a ukazatel na zbytek již zakódovaných dat

Ukazatel je reprezentován 16bity a vždy začíná bity 11.
To indikuje, že se jedná právě o ukazatel. Pokud se jedná o zakódovaná data jsou první 2 bity naopak vždy 00. Zbývajících 14 bitů ukazatele pak představuje offset od začátku zprávy, kde se již zakódovaná data nacházejí.

Implementace

Rozděleno podle jednotlivých souborů.

- ipk-lookup.c = hlavní kostra programu, realizace komunikace s DNS servery
- args_parser.c = zpracování argumentů příkazové řádky
- dns_convert.c = funkce pro konverze mezi binárním formátem dat v DNS zprávě a jejich tisknutelnou reprezentací
- iterative.c = funkce pro iterativní způsob dotazování

Závěr

Cílem projektu bylo vytvořit aplikaci pro komunikaci s DNS servery. Celkové řešení však zahrnovalo mnohem více problematik, které bylo potřeba řešit. Vytvoření a zpracování DNS zpráv vyžadovalo realizaci funkcí operujících přímo s binárními daty, funkce realizující konverze mezi binárními daty a jejich tisknutelnou reprezentací a nebo naopak překlad řetězce znaků do formátu definovaného pro DNS zprávy.
Projekt byl poměrně náročný, ale jeho řešení bylo velice zajímavé a velmi přínosné.

Zdroje

- [1], [2], [3], [4] P. Mockapetris, RFC 1035: Domain names - implementation and specification, https://tools.ietf.org/html/rfc1035

- Kurose, Ross. Computer networking a top-down approach