VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA INFORMAČNÍCH TECHNOLOGIÍ
FACULTY OF INFORMATION TECHNOLOGY

ÚSTAV INFORMAČNÍCH SYSTÉMŮ
DEPARTMENT OF INFORMATION SYSTEMS

DNS RESOLVER

ISA PROJEKT

AUTOR PRÁCE **VILIAM CHUDÁČIK, xchuda06**
AUTHOR

BRNO 19. listopadu 2023

Obsah

Kapitola 1

Úvod

Výsledkem tohoto projektu je aplikace, která přijme parametry pro DNS dotaz pomocí spouštěcích argumentů, pomocí těchto parametrů vytvoří dotaz a následně ho odešle na DNS pomocí IPv4 nebo IPv6. Aplikace poté přehledně zobrazí přijatou odpověď uživateli.

Kapitola 2

Návrh aplikace

2.1 Řešená problematika

2.1.1 DNS zprávy

DNS dotazy a odpovědi se řídí několika různými RFC, které specifikují jak mají tyto zprávy vypadat. Zde např. RFC 1035[1], RFC 2535[2](Obeseleted by RFC4035[4]) a RFC3596[3] pro IPv6.

2.1.2 Hlavička zpráv

Každý DNS dotaz začíná 16-bitovým identifikátorem, který slouží pro spojení dotazů s odpověďmi.
Následuje 16-bitová sekce příznaků. Prvním příznakem je QR, který určuje, jestli se jedná o dotaz(0) nebo odpověď(1). Následuje 4-bitový operační kód, pro potřeby tohoto DNS Resolveru je použit pouze operační kód 0000 nebo-li Standardní dotaz. Dalším bitem je příznak autoritativní odpovědi, tento bit se nastavuje v odpovědi a určuje, zda dotazovaný DNS je autoritativním serverem pro dotazovanou doménu. Dalším bitem je TrunCation, určující, že odpověď nebo dotaz byl příliš dlouhý a tedy nemohl být odeslána přes UDP. Maximální délka dns dotazu nebo odpovědi je 512bajtů. Následuje bit pro určení zda by měl dotaz být proveden rekurzivně. V případě, že dotaz má být proveden rekurzivně, tak po té co DNS přijme dotaz a v cache nenajde odpověď, provede odeslání rekurzivního dotazu na root server pro získání odpovědi. Pokud se nejedná o rekurzivní dotaz, tak DNS hledá odpověď pouze v cache. Další bit je nastaven v odpovědi a určuje jestli server podporuje rekurzivní dotazy. Následuje jeden bit, který je rezervován pro další použití a musí být vždy nastaven na hodnotu 0. Následují dva bity AD a CD, které slouží pro zabezpečení, oba jsou zanedbatelné pro tento DNS resolver. Poslední 4 bity jsou nastavovány v odpovědi a slouží pro určení kódu odpovědi. Kód je 0 v případě, že nedošlo k chybě. Vyšší hodnoty značí tyto chyby: Error formátování(1), Server error(2), Neexistující doménové jméno(3), Neimplementováno(4), Odmítnuto(5). Hodnoty 6-15 jsou rezervovány pro budoucí použití

Poslední částí jsou čtyři 16 bitové hodnoty, označující počet dotazů, odpovědi, autoritativních záznamů a dalších záznamů.

2.1.3 DNS dotaz

DNS dotaz se skládá z hlavičky a části pro Dotaz. Část pro dotaz začíná doménovým jménem, který je rozdělen na jednotlivé bloky pomocí teček. Před každý blok je umístěno číslo reprezentující počet znaků, které budou následovat do dalšího bloku. Tečky se neukládají. Tím pádem např. "www.vutbr.cz"by bylo uloženo jako "3www5vutbr2cz0". Další je 16 bitový QTYPE, který určuje typ dotazu. U tohoto resolveru se používá 1 pro A záznam, 12 pro PTR záznam a 28 pro AAAA záznam. Poslední 16-bitové pole je QCLASS, které pro potřeby tohoto resolveru je vždy nastaven na 1 nebo-li IN(Internet).

2.1.4 DNS odpověď

DNS odpověď se skládá z hlavičky, přijatého dotazu a 3 částí pro odpověď. První je běžná odpověď, druhá jsou záznamy autoritativních serverů a třetí další záznamy. Části pro odpověď se skládají z pole pro doménové jméno, ke kterému tento záznam patří. Doménové jméno je zde ukládáno stejně jako v DNS dotazu, ale může navíc obsahovat ukazatele. Ukazatel je 16bitové celé číslo, které začíná dvěmi bity, které jsou nastavené na 1. Zbylých 14 bitů je offset od začátku přijaté zprávy, určující kde je třeba hledat zbytek domény. Následuje 16-bitové pole pro Typ, zde krom A, AAAA a PTR se můžeme také setkat s NS(2), CNAME(5) a SOA(6). Následuje pole CLASS, zde bude také vždy hodnoty 1 pro IN. Následuje 32-bitové TTL pole, reprezentující celé číslo se znaménkem. Tato hodnota určuje, po jakou dobu by se měl tento záznam považovat za platný. Následuje 16 bitová hodnota určující délku dat, které následují. Poslední částí jsou data, tato část se liší pro jednotlivé typy odpovědí. Struktura dat bude dále popsáno pro každý typ.

2.1.5 A

Záznam typu A slouží pro přiřazení IPv4 adresy k doméně. V části pro data najdeme pouze tuto IPv4 adresu uloženou bez teček.

2.1.6 AAAA

Záznam typu A slouží pro přiřazení IPv6 adresy k doméně. V části pro data najdeme pouze tuto IPv6 adresu uloženou bez dvojteček.

2.1.7 CNAME

Záznam typu CNAME slouží jako alias. V části pro data nalezneme doménové jméno, na které se dotazovaná doména odkazuje. Toto doménové jméno je uloženo stejným způsobem jako v dotazu. Po přistoupení na doménu s CNAME záznamem, dojde k přesměrování na tuto doménu. Společně s CNAME odpovědí může často přijít A záznam pro doménu obsaženou v části pro data.

2.1.8 NS

Záznam typu NS neboli NameServer určuje autoritativní server pro dotazovanou doménu. Záznam se nachází v autoritativní části. Data mají stejný formát jako CNAME. Záznam typu NS, dále způsobí, že server v části pro další záznamy odešle A nebo AAAA záznam pro každý NS záznam.

2.1.9 PTR

Záznam typu PTR slouží pro reverzní dotazy. Tedy pro to, abychom zjistili, jaké domény odkazují na dotazovanou IP adresu. Data mají stejný formát jako CNAME.

2.1.10 SOA

Záznam typu SOA neboli Start of authority a slouží pro zobrazení administrativních informací o DNS zóně. Záznam se nachází v autoritativní části. Data obsahují doménu primárního name serveru, email správce, 32bitové sériové číslo, 32bitové číslo určující interval obnovení, 32bitové pro opakovaný pokus, 32bitové číslo určující maximální platnost a 32bitové číslo určující minimální platnost. Doména je uložena stejně jako v dotazu, v e-mailu se vymění @ za . a poté je uložen stejně jako doména v dotazu. Všechny časové hodnoty jsou uloženy v sekundách.

Kapitola 3

Popis implementace

Při spuštění, dojde k ověření správnosti argumentů. Při spuštění je nutné použít parametr -s s IP adresou nebo doménový jménem DNS serveru a dále musí být zadána dotazované doménové jméno nebo IP adresa. O zpracování a kontrolu argumentů se stará funkce argPars obsažená v souboru arguments.cpp
Poté dojde k zavolání funkce dnsquery, která je obsažená v souboru dns.cpp. Na začátku se zavolá funkce createDNSQuery, kde vstupním parametrem je datová struktura se vstupními parametry. Po vytvoření dotazu, dojde k jeho odeslání na DNS server pomocí sendQueryIP4 nebo sendQueryIP6. Po přijetí je provedeno čtení a kontrola odpovědi funkcí responseParser. A následně dojde k vytištění přijaté odpovědi nebo chybové hlášky.

3.1 createDNSQuery

V této funkci dojde k vytvoření vektoru 8bitových bez znaménkových celých čísel. Do tohoto vektoru se budou postupně ukládat jednotlivé bajty dns dotazu. Jako první dojde k uložení vygenerovaného ID. Následně se vytvoří struktura pro hlavičku, do které se uloží jestli se jedná o standardní nebo rekurzivní dotaz, nastaví se počet otázek na 1 a počty odpovědí na 0 a následně dojde k vložení hlavičky do vektoru. Pokud se jedná o reverzní dotaz tak dojde k převodu IP adresy na vstupu na reverzní formát. Dále dojde k vložení čísla typu a třídy dotazu. Pro odeslání se využívá funkce sendQueryIP4 nebo sendQueryIP6.

3.2 sendQueryIP4/6

Po vytvoření dns dotazu, dojde ke kontrole adresy DNS serveru pomocí regexu. Pokud se jedná o IPv4 dojde k zavolání funkce pro odeslání dotazu přes IPv4, v případě IPv6 zase funkce pro odeslání přes IPv6. Pokud se nejedná ani o IPv4 ani o IPv6 tak dojde k zavolání funkce pro získání defaultních DNS serverů ze souboru /etc/resolv.conf. Poté se DNS resolver pokusí o získání záznamu typu A nebo AAAA pro zadaný DNS server za pomocí defaultního DNS serveru.
V případě IPv4 i IPv6 dojde k vytvoření struktury s adresou a k vytvoření socketu. Po odeslání dotazu dojde k nastavení time-out hodnoty pro přijmutí odpovědi na 2 sekundy. Pokud nedojde k přijetí odpovědi, tak dojde k vypsání chybové hlášky na chybový výstup a ukončení programu. V případě přijetí odpovědi se volá funkce responseParser.

3.3 responseParser

Po úspěšném přijetí odpovědi, dojde k porovnání ID přijaté odpovědi s ID dotazu.
Následně dochází ke kontrole jestli se jedná o odpověď na standardní dotaz. Poté se kontroluje nastavení bitu TrunCation, které by značilo, že došlo ke zkrácení odpovědi, v tom případě dojde k nastavení příznaku a navrácení do mainu, kde dojde k tisku odpovědi. Poté se přečte a nastaví příznak pro autoritativní a rekurzivní odpověď. Poté se přečte kód odpovědi a v případě, že je kód různý od nuly tak dojde k vypsání chyby. Poté dojde k uložení počtů otázek a odpovědí všech typů. Po přeskočení části s dotazem začne čtení části s odpovědí pomocí funkce ACNAME.
Pokud je přítomna část pro autoritativní odpovědi, tak nejdříve dojde k přečtení údajů společných pro SOA a NS typ záznamu. Pokud se jedná o SOA typ záznamu tak dojde k přečtení zbytku údajů, specifických pro tento typ záznamu. Poté dojde k přečtení případný dalších záznamů pomocí funkce ACNAME. V průběhu celého čtení dochází ke kontrole, aby nedošlo k čtení mimo alokovaný vektor.
Pomocnými funkcemi při čtení odpovědí je domainParser a bytesToInt.

3.4 domainParser

Funkce domainParser, slouží ke čtení doménových jmen z dns odpovědí. Vstupem do funkce je vektor odpovědi, aktuální pozice ve vektoru, error kód a počet přijatých bitů. Funkce čte z odpovědi doménové jméno dokud nenarazí na nulový bajt. V průběhu čtení mění délky následujících bloků za tečky. V případě, že narazí na pointer, který začíná hexadecimální hodnotou 0xC0 tak dojde k rekurzivnímu volání funkce domainParser začínající na offsetu, který byl přečten.

3.5 bytesToInt

Funkce bytesToInt slouží pro přečtení zadaného počtu bajtů z vektoru odpovědi a převedení této hodnoty na celé číslo, které je poté vráceno.

Kapitola 4

Testování

Testy jsou napsány v c++ a dají se spustit pomocí příkazu *make test*.
Testují se jednotlivé komponenty aplikace a to zpracování argumentů, vytváření dotazu a čtení dotazu.
Poté se testuje na pár příkladech funkčnost celého programu. Zde je problémové mít více automatických testů, jelikož se záznamy na DNS serverech můžou v průběhu času měnit.
Skutečný výstup se porovnává s očekávaným, který je uložen v kódu.

Kapitola 5

Popis programu a návod na obsluhu

Aplikace DNS Resolver slouží pro zaslání dotazů na DNS servery a pro následné zobrazení odpovědi uživateli.

5.1 Build aplikace

Aplikace se dá buildnout pomocí příkazu *make*. Pro build s možností debugování je příkaz *make debug*. Pro smazání všech binárních souborů, slouží příkaz *make clean*.

5.2 Použití aplikace

Aplikace se spustí pomocí příkazu *dns [-r] [-x] [-6] -s server [-p port] adresa*. Kde -s je povinný parametr, který musí být následován IPv4, IPv6 nebo doménou DNS serveru. Druhým povinným parametrem je *adresa*, jako adresu vyplňte dotazovanou doménu, nebo v případě reverzního dotazu IP adresu.
Pokud nejsou použity parametry jako -x nebo -6 tak server vrací odpovědi typu A, CNAME, NS a SOA.
Parametr -r, je pro rekurzivní dotazy. Pokud je tento parametr nastaven a dotazovaný server podporuje rekurzivní dotazy, bude server hledat odpověď dále na root serverech, v opačném případě bude server hledat odpověď pouze v cache.
Parametr -x, slouží pro reverzní dotazy. Tento parametr nesmí být použit spolus s parametrem -6. Při použití reverzního dotazu, se do argumentu *adresa* vyplňuje IPv4 nebo IPv6 adresa. Reverzní dotaz poté vrátí výčet domén, používající danou IP adresu v DNS záznamech.
Parametr -6, slouží pro navrácení záznamu typu AAAA, místo A.
Parametr -p, pokud použit, musí být následován portem na kterém bude DNS server odpovídat na DNS dotazy. V případě nepoužití tohoto parametru se použije port 53.
Pro zobrazení rychlé nápovědy lze použít argument *–help*, *-help*, *-?* nebo *-h*.

5.3 Výpis odpovědi

První řádek odpovědi obsahuje příznaky Authority, Recursive a Truncated. Následuje Question section, obsahující dotaz zaslaný na DNS. Poté následují sekce jednotlivých typů odpo-

vědí, kde odpovědi jsou ve formátu dotazovaná adresa, typ záznamu, třída záznamu a TTL. V případě A nebo AAAA záznamu je poslední položkou IP adresa. V případě CNAME, NS nebo PTR záznamu je poslední položkou doména. V případě SOA za TTL následuje primární name server, e-mailová adresa správce, sériové číslo záznamu, interval obnovení, interval pro opakování, maximální platnost a minimální platnost záznamu.

5.4 Řešení problémů

V případě, že dojde k jakémukoli problému nebo chybě, je uživatel informován chybovou hláškou na chybovém výstupu. Pokud dojde k chybě "Error creating socket", "Error sending data"nebo "setsockopt failed"prosím zkontrolujte vaše internetové připojení. Pokud posíláte dotaz na DNS server s IPv6 adresou, ověřte, že vaše internetové připojení podporuje IPv6 a to např. na odkaze https://ipv6test.google.com/. V případě chyby "Failed to receive data from dns"zkontrolujte vaše internetové připojení, správnost adresy a př. port dotazovaného DNS serveru. V případě chyby "Domain name is too long"zkontrolujte dotazovanou doménu, př. adresu DNS serveru. Maximální délka doménové adresy je 253 znaků.

Literatura

[1] *RFC1035*. RFC Editor, listopad 1987. Dostupné z: https://www.rfc-editor.org/info/rfc1035.

[2] 3RD, D. E. E. *RFC2535 Domain Name System Security Extensions*. RFC Editor, březen 1999. Dostupné z: https://www.rfc-editor.org/info/rfc2535.

[3] KSINANT, V., HUITEMA, C., THOMSON, D. S. a SOUISSI, M. *RFC3596 DNS Extensions to Support IP Version 6*. RFC Editor, říjen 2003. Dostupné z: https://www.rfc-editor.org/info/rfc3596.

[4] ROSE, S., LARSON, M., MASSEY, D., AUSTEIN, R. a ARENDS, R. *RFC4035 Protocol Modifications for the DNS Security Extensions*. RFC Editor, březen 2005. Dostupné z: https://www.rfc-editor.org/info/rfc4035.