17.10.2020

FIT VUT – ISA

PROGRAMOVÁNÍ SÍŤOVÉ SLUŽBY - FILTRUJÍCÍ DNS RESOLVER

SASÍN JONÁŠ (XSASIN05)

Obsah

1. Rozbor problematiky

DNS

DNS je hierarchický systém, který ukládá informace o názvech strojů a domén v decentralizované databázi DNS serverů. Jeho hlavním úkolem je převod doménového jména na IP adresu uzlu sítě a naopak (později další funkce).

Právě převod doménových jmen na IP adresy je úkolem zadání. DNS server má mít navíc schopnost *filtrovat* dotazy na nežádoucí domény, které se nemají zpracovávat. Odborně řečeno má být tedy schopen zpracovávat DNS dotazy typu A pro domény, které se nevyskytují v přiloženém seznamu nechtěných domén.

Implementovaný server tuto činnost vykonává poměrně triviálně. Očekává ve smyčce dotaz od klienta na předem stanoveném portu. Po přijmutí DNS dotazu zkontroluje jeho hlavičku, typ dotazu a hlavně jméno domény. Pokud je vše v pořádku, přeposílá DNS dotaz na předem adresou určený server, který dotaz zpracuje a zpátky odešle odpověď. Přijatou odpověď už náš server pouze přepošle původnímu tazateli, tedy našemu klientovi. Server čeká na další dotaz.

Struktura DNS packetu

Packet se skládá z částí:

- Header (hlavička)
- Question (otázka)
- Answer (odpověď)
- Authority (autoritativní server)
- Additional (dodatečná informace)

Rozeberu pouze pro nás implementačně důležité části, abych zde pouze neopisoval celé RFC.

Header

Přesnou strukturu hlavičky lze najít např. v RFC 1035 (nebo soubor dnsPacketStruct.hpp), pro nás důležité části ale jsou:

- QR – (1 bit) – Identifikující otázku (0), nebo odpověď (1)
- QDCOUNT – (16 bit) – specifikuje počet otázek

Pro hlavičku je také důležité, že má pevnou velikost (12 B).

Question

Otázka se skládá z následujících částí:

- NAME – jméno domény (proměnlivá délka)
- TYPE – typ dotazu
- CLASS – třída komunikace (pro nás pouze IN – Internet)

Kvůli jménu domény má otázková část paketu proměnlivou velikost, proto není možné triviálně udělat pouze typecast do struktury. Jméno se musí parsovat samostatně a typecast se provede až pro zbytek otázky beze jména.

Name

Jméno domény je rozdělené na jednotlivé štítky pomocí čísla, které reprezentuje, kolik znaků následující štítek obsahuje. Domény proto převedeme na tvar, na který jsme zvyklí, nahrazením odpovídajících čísel tečkou (“.“). (3fit5vutbr2cz -> fit.vutbr.cz)

Type

Typ záznamu/dotazu – je jich mnoho, my v projektu implementujeme pouze typ A – záznam obsahující IPv4 adresu. Pro případný reverzní dotaz používáme typ PTR, který ale není součástí zadání.

2. Popis implementace

Zdrojový kód je pro přehlednost rozdělen do více souborů. Popíšu tedy jednotlivé funkce, které jsou v průběhu programu volány. Pro jednoduchost samotné implementace jsem se rozhodl nepoužít objektového návrhu.

paramParse.cpp:

parse_args

Funkce naplní strukturu pro uložení argumentů, kontroluje jejich syntaktickou správnost. Pro jednoduché zpracování argumentů programů je použita funkce getopt. Funkce vrací číslo indikující úspěšnost zpracování přepínačů.

Funkce také tiskne --help neboli nápovědu pro užití programu.

getUnwantedDomains

Funkce načte ze souboru specifikovaného přepínačem -f nežádoucí domény. Domény čte po řádku ze souboru, řádky začínající # a prázdné řádky ignoruje, ostatní ukládá do vektoru stringů. Funkce vrací vektor těchto stringů.

createSock.cpp:

setClientSock

Vytvoří UDP socket na specifikovaném portu (přepínač -p) pro komunikaci s klientem, provede bind a vrací deskriptor daného socketu. V případě neúspěchu vrací EXIT_SOCK_FAILURE neboli záporný deskriptor.

setResolverSock

Vytvoří UDP socket na implicitním DNS portu 53 pro komunikaci se serverem (na tomto socketu vystupuje náš program jako klient). Pomocí funkce getaddrinfo zjistíme, zda je server možno kontaktovat. Otevřeme socket a funkcí connect otestujeme, je-li možné se k serveru připojit. Funkce opět vrací popisovač otevřeného socketu a v případě neúspěchu EXIT_SOCK_FAILURE.

dns.cpp

main

Volá funkce pro zpracování argumentů, nastavení socketů a funkci startServer, která řeší hlavní implementační část programu.

startServer

Ve smyčce očekává zprávy od klientů na klientském socketu. Po přijetí dotazu proběhne parsování DNS packetu funkce parseDnsPacker. V případě zjištění chyby nebo nežádoucí domény se v hlavičce paketu nastaví odpovídající rcode a přepošle se zpátky klientovi. Pokud se nezjistí žádný problém, přepošle se dotaz na předem specifikovaný server (otevřený server socket). Obdržená odpověď serveru se přepošle opět klientovi, který tak dostává odpověď na svůj dotaz a náš server opět čeká na další.

parseDns.cpp

parseDnsPacket

Funkce zpracuje přijatý paket. Nejprve zpracuje hlavičku, poté část z otázkou (nebo otázkami), volá funkci parse_domain a check_domain pro filtrování nežádoucích domén. Kontroluje typ dotazu.

V případě úspěchu vrací funkce 0, v případě nepodporovaného dotazu (jiný než A) vrací NOTIMP(4) a v případě nežádoucí domény REFUSED(5).

Před návratem z funkce musí funkce převést zpět všechny hodnoty paketu na správné pořadí bitů.

parseDomain

Funkce převede proměnlivě dlouhé jméno domény do srozumitelného tvaru a vrací ukazatel za konec jména, tedy za část NAME otázky. Poté je možně zbylou část Question části paketu type-castovat do struktury, protože její délka již není proměnlivá.

checkDomain

Prochází seznam nežádoucích domén a porovnává je se jménem domény v DNS dotazu. Kontroluje tedy, zda dotazovaná doména není nežádoucí, nebo jestli není pod-doménou některé z nežádoucích domén.

setAnswerErr

Volá se v případě neúspěchu funkce parseDnsPacket. Nastaví rcode hlavičky dns paketu na odpovídající flag. Nastaví taky qr hlavičky na 1(odpověď), aby mohl být dotaz zaslán zpět klientovi.

dnsPacketStruct.hpp

Obsahuje strukturu hlavičky DNS paketu dns_header a strukturu neměnné části otázky paketu (bez *name*) dns_question.

3. použití programu

Přeložení a spuštění

Program je možno přeložit pomocí přiloženého Makefile příkazem ***make.***

Použití (okopírováno ze zadání):

./dns -s server [-p port] -f filter_file

Pořadí parametrů je libovolné. Popis parametrů:

- -s: IP adresa nebo doménové jméno DNS serveru (resolveru), kam se má zaslat dotaz.
- -p port: Číslo portu, na kterém bude program očekávat dotazy. Výchozí je port 53.
- -f filter_file: Jméno souboru obsahující nežádoucí domény.

Chybové zprávy

Předčasné ukončení

Program běží v nekonečné smyčce a ukončit je ho možné např. příkazem ctrl+C.

Při spuštění programu může program skončit předčasně s návratovým kódem:

- 0
  - špatně zadané parametry programu
  - volání --help
  - soubor obsahující nežádoucí domény se nepodařilo otevřít
- 1
  - chyba při tvorbě socketů

Každé předčasné ukončení je doprovázeno jednoduchým chybovým hlášením.

Chyba při zpracování DNS paketu

V takové situaci program mlčí a posílá paket zpět klientovi s odpovídajícím rcode v hlavičce paketu.

Rcode:

- REFUSED(5) – nežádoucí doména
- NOTIMP (4) – jiný než “A“ dotaz – není implementováno

4. Zdroje

1) RFC 1035 DOMAIN NAMES - IMPLEMENTATION AND SPECIFICATION
   https://tools.ietf.org/html/rfc1035
2) Domain Name System – Wikipedia
   https://cs.wikipedia.org/wiki/Domain_Name_System