Vysoké učení technické v Brne

Fakulta informačných technológií

Síťové aplikace a správa sítí

Filtrující DNS resolver

Radoslav Eliáš (xelias18)

Obsah:

Použitie

Preklad:

Preklad sa prevedie príkazom `make`.

Spustenie programu:

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

Alebo

```
make run OPTIONS="-s server [-p port] -f filter_file [-v]"
```

Parametre:

`-s:` ipv4/ipv6 adresa alebo doménové meno DNS servera kam sa dotaz prepošle.

`-p:` port na ktorom bude server počúvať. Ak nie je špecifikovaný tak sa použije port 53.

`-f:` názov súboru ktorý obsahuje nežiadúce domény. Ak daný súbor neexistuje, všetky dotazy sa budú preposielať bez filtrovania.

`-v:` príznak na výrečnosť programu. Server bude oznamovať na výstup akú činnosť vykonáva.

Príklady spustenia:

```
./dns -s 8.8.8.8 -p 5353 -f filterfile.txt
./dns -s dns.google.com -p 1234 -f filterfile.txt -v
```

Alebo

```
make example
```

Upratanie:

`make clean`: vymaže spustiteľný súbor vytvorený prekladom.

Implementácia

Jazyk: C++

Prostredie: Linux

Program podporuje komunikáciu cez UDP protokol na transportnej vrstve a výhradne DNS dotazy typu A. Implementácia neuvažuje viac ako 1 položku(otázku) v časti queries.

Použité knižnice:

`iostream, getopt.h, unistd.h, stdio.h, stdlib.h, string.h, fstream`
Práca so súbormi, reťazcami, parsovanie argumentov, interné reprezentácie atď.

`netinet, arpa, netdb.h, socket.h, types.h`
Hlavičky sieťových protokolov, funkcie na komunikáciu po sieti, práca so socketmi...

Štruktúra:

Program je celý v jednom zdrojovom súbore `dns.cpp`. Po spustení sa naparsujú argumenty zadané užívateľom, otvorí socket na danom porte a začne načúvanie pre dotazy.
Po zachytení dotazu program skontroluje príznaky v hlavičke a vyparsuje dotazovanú doménu. Ak typ dotazu nie je A, server odošle klientovi jeho dotaz s nastaveným príznakom RCODE na 4, teda NOT IMPLEMENTED.
Potom sa zavolá funkcia `filter()`, ktorá porovná dotazovanú doménu s listom nežiadúcich domén. Podľa návratovej hodnoty funkcie sa klientovi pošle odpoveď s príznakom RCODE = 5, teda REFUSED ak je doména na zozname blokovaných. Inak sa predá riadenie funkcii `forward_query()`.
Následne sa vytvorí štruktúra a socket pre DNS server na ktorý bude dotaz preposlaný. Podporované formáty zadaného serveru sú ipv4/ipv6 adresa alebo doménové meno. Po preposlaní dotazu sa odpoveď v nezmenenej podobe prepošle klientovi a začne načúvanie pre ďalší dotaz.
Program načúva a beží, kým ho užívateľ nepreruší, napr. cez `Keyboard Interrupt`(ctrl+c).
Ak nastane chyba pri inicializácii serveru a socketu pre načúvanie program skončí a vráti hodnotu -1. Chyba pri spracovaní a odpovedaní na dotaz vypíše na chybový výstup zodpovedajúcu hlášku, dotaz zahodí a začne znova načúvať.

Testovanie

Priložený je shell script `tests.sh` ktorý otestuje základnú funkcionalitu programu. Je spustiteľný z Makefilu príkazom `make test`. Príklad časti výstupu skriptu na operačnom

```
starting server...
dns server: 8.8.8.8
port: 6789
filterfile2.txt
...querying wis.fit.vutbr.cz
expected: status: NOERROR
output:   status: NOERROR
-->   TEST PASSED
...querying docs.google.com
expected: status: NOERROR
output:   status: NOERROR
-->   TEST PASSED
...querying 8.8.8.8 PTR(not supported query type)
expected: status: NOTIMP
output:   status: NOTIMP
-->   TEST PASSED
```

systéme ubuntu:

Testovanie bolo vykonané na lokálnom stroji s OS ubuntu a školských serveroch `merlin` a `eva`. Skript spustí server ako proces na pozadí a potom posiela DNS dotazy pomocou nástroja `dig`. Z výstupu tohto programu sa potom vyberie hodnota príznaku `status` a porovná s očakávaným výsledkom. Dodatočné testovanie bolo vykonané rovnakým spôsobom pomocou dvoch terminálov vystupujúcich ako server a klient.
Príklad:

```
/m/c/U/R/D/s/I/ISA-project >>> ./dns -s 8.8.8.8 -f filterfile2.txt -p 9134 -v
Creating socket file descriptor...
listening for query...
query recieved...
parsing...
checking query type flag...
comparing domain to blacklist...
creating socket for dns resolver...
forwarding query to dns resolver...
waiting for response from dns resolver...
response from resolver recieved...
sending response to client...
listening for query...
```

```
/m/c/U/R/D/s/I/ISA-project >>> dig @localhost -p 9134 seznam.cz

; <<>> DiG 9.16.1-Ubuntu <<>> @localhost -p 9134 seznam.cz
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 39895
;; flags: qr rd ra ad; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;seznam.cz.                     IN      A

;; ANSWER SECTION:
seznam.cz.              101     IN      A       77.75.75.172
seznam.cz.              101     IN      A       77.75.75.176

;; Query time: 72 msec
;; SERVER: 127.0.0.1#9134(127.0.0.1)
;; WHEN: Mon Nov 16 15:03:15 CET 2020
;; MSG SIZE  rcvd: 70
```

Pre overenie jednotlivých krokov serveru je k dispozícii prepínač `-v`, ktorý umožní dodatočný výpis pre lokalizovanie chyby programu.
Východzí port pre DNS je 53, ktorý je ale v unixovom systéme rezervovaný. Preto na spustenie serveru na tomto porte je vyžadované byť prihlásený ako `superuser`, teda napríklad: `sudo ./dns -s 8.8.8.8 -f filterfile2.txt`. Táto vlastnosť bola otestovaná iba na lokálnom stroji, vzhľadom na právomoci študenta na školských serveroch.
Korektná práca s pamäťou bola testovaná pomocou nástroja `valgrind`:

```
/m/c/U/R/D/s/I/ISA-project >>> valgrind ./dns -s 8.8.8.8 -f filterfile2.txt -p 1234
==15029== Memcheck, a memory error detector
==15029== Copyright (C) 2002-2017, and GNU GPL'd, by Julian Seward et al.
==15029== Using Valgrind-3.15.0 and LibVEX; rerun with -h for copyright info
==15029== Command: ./dns -s 8.8.8.8 -f filterfile2.txt -p 1234
==15029==
==15029== error calling PR_SET_PTRACER, vgdb might block
^C==15029==
==15029== Process terminating with default action of signal 2 (SIGINT)
==15029==    at 0x4B7252A: recvfrom (recvfrom.c:27)
==15029==    by 0x10B433: main (dns.cpp:336)
==15029==
==15029== HEAP SUMMARY:
==15029==     in use at exit: 0 bytes in 0 blocks
==15029==   total heap usage: 15 allocs, 15 frees, 99,604 bytes allocated
==15029==
==15029== All heap blocks were freed -- no leaks are possible
==15029==
==15029== For lists of detected and suppressed errors, rerun with: -s
==15029== ERROR SUMMARY: 0 errors from 0 contexts (suppressed: 0 from 0)
```

Zdroje

https://www.geeksforgeeks.org/socket-programming-cc/
web.cecs.pdx.edu/~jrb/tcpip/sockets/ipv6.src/udp/udpclient.c
linux man pages
https://www.ietf.org/rfc/rfc1035.txt
http://www.cplusplus.com/reference
príklady sieťových aplikácii - súbory k predmetu ISA