Vysoké učení technické v Brně
Fakulta informačních technologií

Síťové aplikace a správa sítí

Monitorování DNS komunikace

17. listopadu 2024 Alisher Mazhirinov

Obsah

1 Úvod

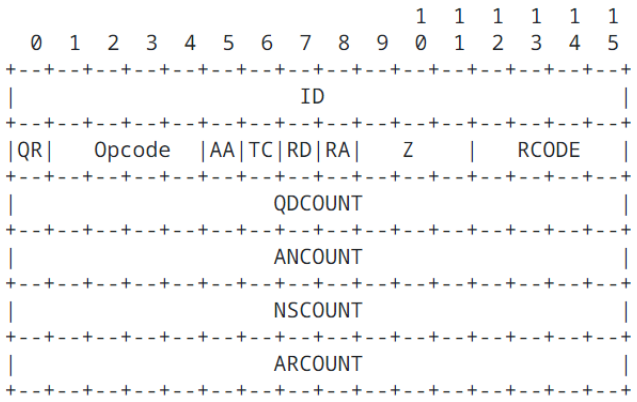
Cílem projektu je implementovat program, který bude monitorovat DNS komunikaci na zvoleném rozhraní nebo z existujícího záznamu ve formátu `PCAP`. Nástroj zpracovává DNS zprávy a vypisuje zjištěné informace. Kromě toho umožňuje zjistit, jaká doménová jména se objevila ve zprávách a hledat překlady doménových jmen na `IPv4/6` adresy.

2 Specifikace

2.1 Domain Name System

Základním úkolem služby DNS je převod doménových adres na IP adresy. Doménové adrese se často říká také doménové jméno. Každé síťové rozhraní v Internetu obsahuje jednoznačný identifikátor, jímž je IP adresa. Pro porovnání či prefixové vyhledávání se používá 32bitové (či 128bitové u IPv6 adres) číslo. IP adresu zjistíme z doménového jména dotazem na server DNS [11].

2.2 DNS hlavička



Obrázek 1: DNS hlavička

Sekce header je vždy přítomna. Záhlaví obsahuje pole, která určují, které ze zbývajících sekcí jsou přítomny, a také uvádějí, zda je zpráva dotazem nebo odpovědí, standardním dotazem nebo nějakým jiným operačním kódem.
Sekce otázek obsahuje pole, která popisují otázku názvovému serveru. Tato pole jsou typu dotazu, třídy dotazu a názvu domény dotazu. Část odpovědí obsahuje RR, které odpovídají na otázku. Sekce autority obsahuje RR, které ukazují na autoritativní jmenný server. Sekce dalších záznamů obsahuje RR, které se vztahují k dotazu, ale nejsou striktně odpovědí na otázku.

Flagy:

- QR (1 bit): je nastaveno na 0 pro dotazy a 1 pro odpovědi.
- OPCODE (4 bity): určuje typ dotazu (0 pro standardní dotaz).
- AA (1 bit): nastaveno na 1 v odpovědích od autoritativních jmenných serverů.
- TC (1 bit): nastaveno na 1, pokud je odpověď zkrácena kvůli její velikosti.
- RD (1 bit): nastaveno na 1, pokud klient chce, aby server prováděl rekurzivní rozlišení.

- RA (1 bit): nastaveno na 1, pokud server podporuje rekurzi.
- Z (3 bity): vyhrazeno pro budoucí použití. Na nových strojích obsahuje Z, AD a CD.
- RCODE (4 bity): označuje výsledek dotazu (0 pro stav bez chyby).

2.3 Zaznamy DNS

Pro ukládání informací v datovém prostoru DNS slouží záznamy DNS. Záznamy jsou uloženy v textové podobě v zónových souborech na serverech DNS. Všechny typy záznamů mají stejný formát, obecný formát definovaný standardem RFC 1035 [2]. Formát záznamu obsahuje položky `NAME`, `TYPE`, `CLASS`, `TTL`, `RDLENGTH` a `RDATA`. Položka `RDATA` se liší podle typu záznamu, kde pro daný typ (např. A či MX) obsahuje odpovídající informace.

Resource Records Format	Example
Name (variable length)	www.fit.vutbr.cz
Type (16 bits)	CNAME
Class (16 bits)	IN (0x0001)
TTL (32 bits)	4106 (1 h 8 min 26 s)
RDLENGTH (16 bits)	9
RDATA (variable length)	tereza.fit.vutbr.cz

Obrázek 2: Format DNS zaznamu[12]

2.4 Popis podporovaných záznamů DNS

2.4.1 Zaznam A

`A` znamená adresa a toto je nejzákladnější typ DNS záznamu: označuje IP adresu dané domény. Nejběžnějším použitím záznamů A je vyhledávání IP adres: přiřazování názvu domény k adrese IPv4. To umožňuje zařízení uživatele připojit se k webové stránce a načíst ji, aniž by si uživatel pamatoval a zadával skutečnou IP adresu.[3]
Pr:

```
google.com. IN A
```

2.4.2 Zaznam AAAA

Záznamy DNS `AAAA` odpovídají názvu domény k adrese IPv6. DNS AAAA záznamy jsou přesně jako DNS A záznamy, kromě toho, že ukládají IPv6 adresu domény místo její IPv4 adresy.[4]
Pr:

```
google.com. IN AAAA
```

2.4.3 Zaznam NS

Záznam `NS` (nebo záznam nameserveru) je záznam DNS, který obsahuje název autoritativního jmenného serveru v doméně nebo zóně DNS. Když klient požádá o IP adresu, může najít IP adresu zamýšleného cíle ze záznamu NS pomocí vyhledávání DNS.[14]
Pr:

```
fit.vutbr.cz. 0 IN NS kazi.fit.vutbr.cz.
```

```
fit.vutbr.cz. 0 IN NS rhino.cis.vutbr.cz.
fit.vutbr.cz. 0 IN NS guta.fit.vutbr.cz.
fit.vutbr.cz. 0 IN NS gate.feec.vutbr.cz.
```

2.4.4 Zaznam MX

Záznamy MX umístěné v zonových souborech DNS, což jsou jednoduché textové soubory sdružující všechny záznamy pro konkrétní doménu, poskytují e-mailovým klientům informace o doméně, pod kterou lze přistupovat k poštovnímu serveru.[10]
Pr:

```
fit.vutbr.cz. 0 IN MX 10 kazi.fit.vutbr.cz.
fit.vutbr.cz. 0 IN MX 20 eva.fit.vutbr.cz.
```

2.4.5 Zaznam SOA

Záznam SOA obsahuje informace týkající se uložení autoritativních dat pro danou zónu. Většinou je to první záznam v zónovém souboru. Každá zóna má právě jeden záznam SOA. Záznam SOA obsahuje jméno primárního serveru DNS pro danou doménu. Dále obsahuje kontakt na správce domény – jeho emailovou adresu.[13]
Pr:

```
google.com. 0 IN SOA ns1.google.com. dns-admin.google.com.
```

2.4.6 Zaznam CNAME

Kanonický název nebo záznam CNAME je typ záznamu DNS, který mapuje název aliasu na skutečný nebo kanonický název domény. Záznamy CNAME se obvykle používají k mapování subdomény, jako je www nebo mail, na doménu hostující obsah této subdomény. For example, a CNAME record can map the web address www.example.com to the actual web site for the domain example.com.[7]
Pr:

```
ocsp.sectigo.com. 3227 IN CNAME ocsp.comodoca.com.cdn.cloudflare.net.
ocsp.comodoca.com.cdn.cloudflare.net. 160 IN A 104.18.38.233
ocsp.comodoca.com.cdn.cloudflare.net. 160 IN A 172.64.149.23
```

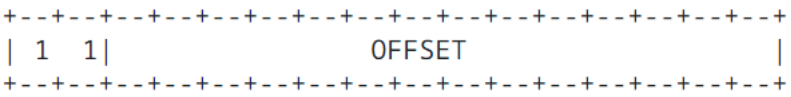
2.4.7 Zaznam SRV

Záznam SRV obvykle definuje symbolický název a transportní protokol používaný jako součást názvu domény. Definuje prioritu, váhu, port a cíl pro službu v obsahu záznamu.[5]
Pr:

```
sip.tcp.example.com. 86400 IN SRV 10 60 5060 bigbox.example.com.
```

2.5 Komprese paketu DNS

Aby se zmenšila velikost zpráv, doménový systém využívá a kompresní schéma, které eliminuje opakování doménových jmen ve zprávě. V tomto schématu je celý název domény nebo seznam štítků na konci názvu domény nahrazen ukazatelem na předchozí výskyt stejného jména.
První dva bity jsou jedničky. To umožňuje rozlišit ukazatel ze štítku, protože štítek musí začínat dvěma nulovými bity, protože štítky jsou omezeny na 63 oktetů nebo méně. Pole OFFSET určuje posun od začátku zprávy.[2]



Obrázek 3: Schéma komprese

3 Implementace

Program `dns-monitor` je implementovan v C, program je plně kompatibilní s operačním systémem Linux. Program podporuje format `Ethernet` a `Linux cooked Capture` verzi `v1`, IPv4/6 adresy. Program podporuje nasledujici DNS zaznamy: `A`, `AAAA`, `NS`, `MX`, `SOA`, `CNAME`, `SRV`.[1, 8, 9] Ostatni typy program ignoruje.

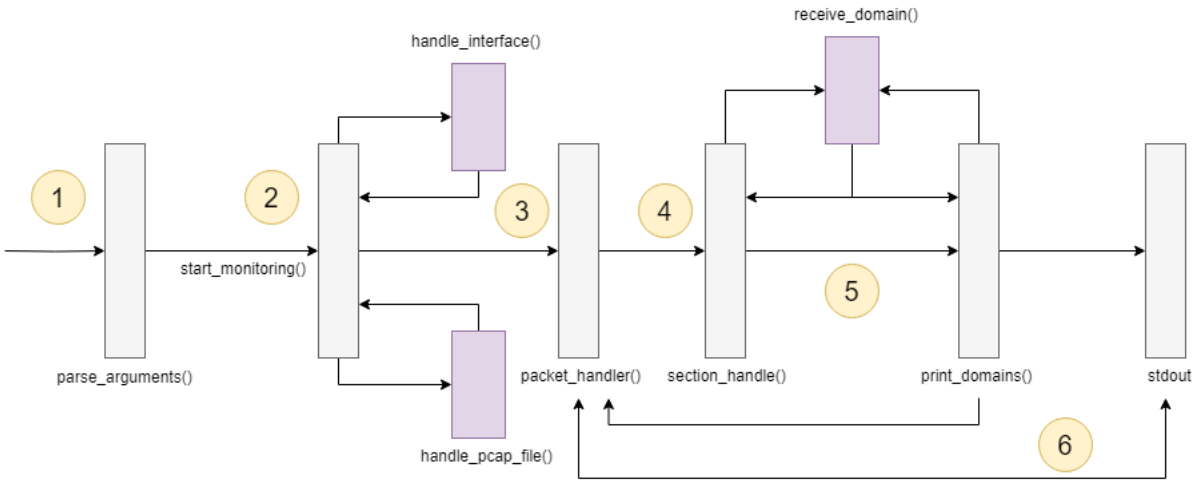
3.1 CLI

Program lze spustit pomoci nasledujiciho prikazu:
`./dns-monitor (-i <interface> | -p <pcapfile>) -v [-d <domains>] [-t <translation>]`

Pro monitorovani lze vybrat bud interface nebo soubor PCAP a nelze vybrat hned oba argumenty. Argument `-v` zapina tak zvany rezim "verbose", ktery umoznuje vypsat kompletni vypis k DNS zpravam. Argumenty `-d` a `-t` jsou volitelne a slouzi pro zapis domenovych jmen a prekladu domenovych jmen do souboru.

3.2 Logika aplikace

Tento obrazek znazornuje beh programu.



Obrázek 4: dns-monitor

3.2.1 Kontrola vstupních argumentu

Program startuje nejprve kontrolou vstupních argumentů, v případě jakékoli odchylky od očekávaného vstupu bude uživateli vypsán návod na správné spuštění. Po kontrole budou parametry uloženy do struktury `InputData` pro snadný přístup v průběhu programu.

3.2.2 Nastavení filtru

Dale probiha kontrola, jestli se jedna o soubor PCAP nebo o interface. Na zaklade toho bude volana funkce `handle_pcap_file()` nebo `handle_interface()`. Tyto funkce slouzi pro zpracováni PCAP souboru nebo pro praci s sitovym rozhrani pomocí knihovny `libpcap`. Obecne tyto 2 funkce vytvareji a nastavuji filtry na protokol `UDP` a port 53.

3.2.3 Zachycení paketu

Dále probíhá kontrola, jestli se jedná o soubor PCAP nebo o interface. Na základě toho bude volána funkce `handle_pcap_file()` nebo `handle_interface()`. Tyto funkce slouží pro zpracování PCAP souboru nebo pro práci s síťovým rozhraním pomocí knihovny `libpcap`. Obecně tyto 2 funkce vytvářejí a nastavují filtry na protokol `UDP` a port 53.

3.2.4 Pruchod sekci

Po zjištění důležitých informací je program připraven pro zjišťování a vypisování informací o DNS komunikacích. Do struktury `dns_header` se ukládají počty jednotlivých sekcí a postupně je program prochází v funkci `section_handle()`. Pro zjištění doménových jmen se používá pomocná funkce `receive_domain()`.

V tomto bodě to my to potřebujeme pro zjištění rodičovské domény, respektive pro jakou doménu vypisujeme informaci. Dále program prochází celou DNS hlavičku a vypisuje typ, třídu a pro typy `Answer`, `Authority` a `Additional` hodnotu `TTL (Time To Live)` a navíc délku dat, ve které lze zjistit IP adresu nebo `Name Server`.

3.2.5 Výpis hodnot záznamu

Po výpisu základních informací musí program také vypsat data, která se nacházejí v `RDATA`. `RDATA` obsahuje data, která přímo odpovídají na dotaz. Pro zjištění doménových jmen nebo jmen serverů se znovu používá funkce `receive_domain()`. Poté program zjištěná data vypisuje na `stdout` a vrátí se zpátky k bodu 3.2.4.

3.2.6 Zkracený výpis

V případě absence parametru `-v` program vypíše zjednodušený výpis, který ukáže uživateli datum, čas, zdrojovou a cílovou IP adresu, typ zprávy (`Query` nebo `Response`) a počty záznamů v sekcích `Question`, `Answer`, `Authority` a `Additional`.

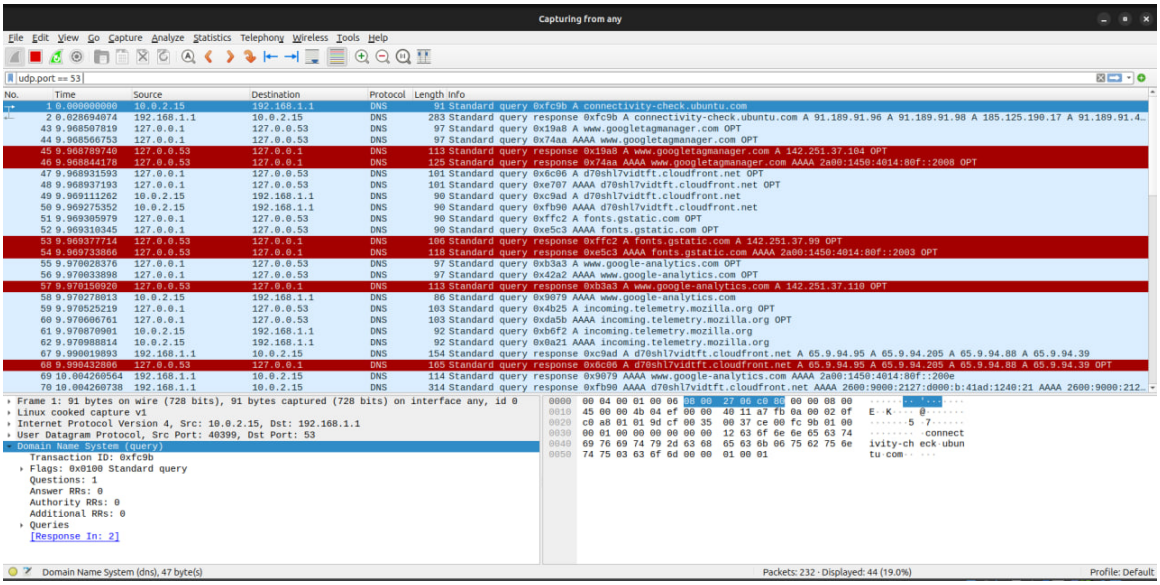
4 Testování

4.1 Na virtuálním stroji

Pro testování programu na virtuálním stroji byl použit referenční virtuální stroj [6] v předmětu IPK [15]. Tento virtuální stroj představuje linuxovou distribuci s názvem NixOS a testovalo se v něm pouze rozhraní. Byl spuštěn příkaz *./dns-monitor -i any -v*, otevřen Wireshark s filtrem `udp.port == 53` a otevřen webový prohlížeč Firefox, ve kterém jsem otevíral různé stránky.

```
(nix:nix-shell-env) ipk@ipk24:~/Documents/isa/ISA$ sudo ./dns-monitor -i any -v
Timestamp: 2024-11-16 18:09:01
SrcIP: 10.0.2.15
DstIP: 192.168.1.1
SrcPort: UDP/40399
DstPort: UDP/53
Identifier: 0xFC9B
Flags: QR=0, OPCODE=0, AA=0, TC=0, RD=1, RA=0, AD=0, CD=0, RCODE=0

[Question Section]
connectivity-check.ubuntu.com. IN A
====================

Timestamp: 2024-11-16 18:09:01
SrcIP: 192.168.1.1
DstIP: 10.0.2.15
SrcPort: UDP/53
DstPort: UDP/40399
Identifier: 0xFC9B
Flags: QR=1, OPCODE=0, AA=0, TC=0, RD=1, RA=1, AD=0, CD=0, RCODE=0

[Question Section]
connectivity-check.ubuntu.com. IN A

[Answer Section]
connectivity-check.ubuntu.com. 30 IN A 91.189.91.96
connectivity-check.ubuntu.com. 30 IN A 91.189.91.98
connectivity-check.ubuntu.com. 30 IN A 185.125.190.17
connectivity-check.ubuntu.com. 30 IN A 91.189.91.49
connectivity-check.ubuntu.com. 30 IN A 185.125.190.18
connectivity-check.ubuntu.com. 30 IN A 185.125.190.48
connectivity-check.ubuntu.com. 30 IN A 185.125.190.98
connectivity-check.ubuntu.com. 30 IN A 185.125.190.97
connectivity-check.ubuntu.com. 30 IN A 185.125.190.96
connectivity-check.ubuntu.com. 30 IN A 185.125.190.49
connectivity-check.ubuntu.com. 30 IN A 91.189.91.97
connectivity-check.ubuntu.com. 30 IN A 91.189.91.48
====================

Timestamp: 2024-11-16 18:09:11
SrcIP: 127.0.0.1
```

Obrázek 5: Terminal ve virtuálním stroji



Obrázek 6: Wireshark ve virtuálním stroji

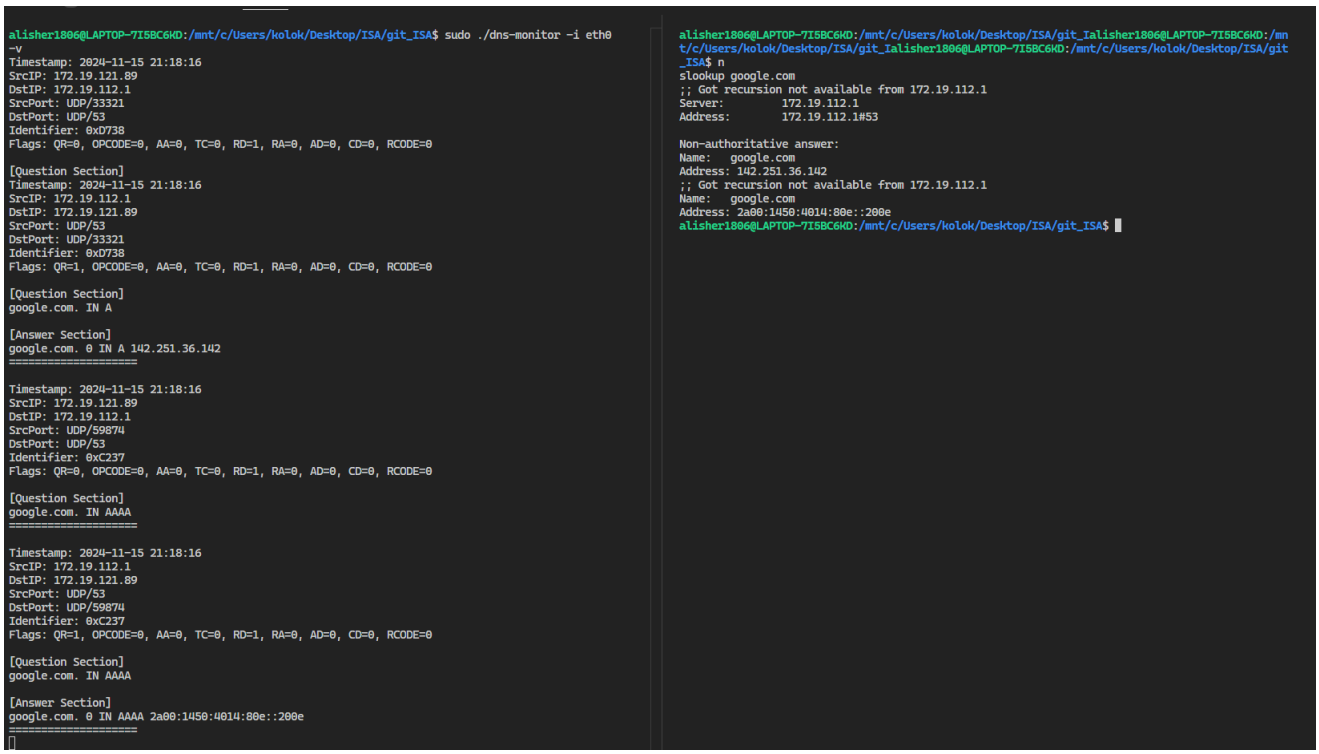
4.2 Lokálně

4.2.1 Rozhraní

Pro testování rozhraní byly použity příkazy `nslookup` a `dig`. Zároveň také byl spuštěn Wireshark s filtrem `udp.port == 53`. Pro spuštění programu jsem zadával do terminálu příslušné příkazy a porovával výstup programu s výstupem v programu Wireshark.

4.2.2 Příklady výpisu

Níže je uveden příklad výpisu komunikace z rozhraní.



Obrázek 7: NsLookup a kompletní výpis

Obrázek 8: NsLookup a zkracený výpis

4.2.3 Soubory PCAP

Pro testování pomocí souboru PCAP jsem použil předpřipravené soubory pro různé typy záznamů. Pro záznam komunikace kromě použití příkazů `nslookup` a `dig` jsem také zkusil otevřít v prohlížeči různé webové stránky, aby komunikaci mohl zachytit Wireshark.

4.2.4 Příklady výpisu

Níže je uveden příklad výpisu komunikace z souboru PCAP.

Source	Destination	Protocol	Length	Info
172.19.121.89	172.19.112.1	DNS	72	Standard query 0xa2be A google.com
172.19.112.1	172.19.121.89	DNS	98	Standard query response 0xa2be A google.com A 142.251.36.142
172.19.121.89	172.19.112.1	DNS	72	Standard query 0x9249 AAAA google.com
172.19.112.1	172.19.121.89	DNS	110	Standard query response 0x9249 AAAA google.com AAAA 2a00:1450:4014:80e::200e

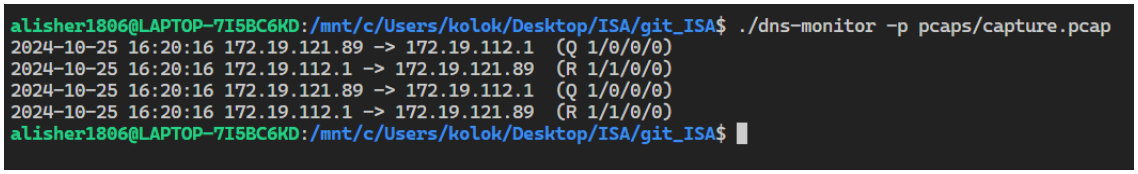
Obrázek 9: PCAP

```
alisher1806@LAPTOP-7I5BC6KD:/mnt/c/Users/kolok/Desktop/ISA/git_ISA$ ./dns-monitor -p pcaps/capture.pcap -v
Timestamp: 2024-10-25 16:20:16
SrcIP: 172.19.121.89
DstIP: 172.19.112.1
SrcPort: UDP/38592
DstPort: UDP/53
Identifier: 0xA2BE
Flags: QR=0, OPCODE=0, AA=0, TC=0, RD=1, RA=0, AD=0, CD=0, RCODE=0

[Question Section]
google.com. IN A
==================

Timestamp: 2024-10-25 16:20:16
SrcIP: 172.19.112.1
DstIP: 172.19.121.89
SrcPort: UDP/53
DstPort: UDP/38592
Identifier: 0xA2BE
Flags: QR=1, OPCODE=0, AA=0, TC=0, RD=1, RA=0, AD=0, CD=0, RCODE=0

[Question Section]
google.com. IN A

[Answer Section]
google.com. 0 IN A 142.251.36.142
==================

Timestamp: 2024-10-25 16:20:16
SrcIP: 172.19.121.89
DstIP: 172.19.112.1
SrcPort: UDP/50773
DstPort: UDP/53
Identifier: 0x9249
Flags: QR=0, OPCODE=0, AA=0, TC=0, RD=1, RA=0, AD=0, CD=0, RCODE=0

[Question Section]
google.com. IN AAAA
==================

Timestamp: 2024-10-25 16:20:16
SrcIP: 172.19.112.1
DstIP: 172.19.121.89
SrcPort: UDP/53
DstPort: UDP/50773
Identifier: 0x9249
Flags: QR=1, OPCODE=0, AA=0, TC=0, RD=1, RA=0, AD=0, CD=0, RCODE=0

[Question Section]
google.com. IN AAAA

[Answer Section]
google.com. 0 IN AAAA 2a00:1450:4014:80e::200e
==================
```

Obrázek 10: Kompletní výpis

```
alisher1806@LAPTOP-7I5BC6KD:/mnt/c/Users/kolok/Desktop/ISA/git_ISA$ cat domains.txt
google.com
alisher1806@LAPTOP-7I5BC6KD:/mnt/c/Users/kolok/Desktop/ISA/git_ISA$ cat translations.txt
google.com 142.251.36.142
google.com 2a00:1450:4014:80e::200e
alisher1806@LAPTOP-7I5BC6KD:/mnt/c/Users/kolok/Desktop/ISA/git_ISA$
```

Obrázek 11: Výpis pro -d a -t parametry



Obrázek 12: Zkracený výpis

Literatura

[1] Domain names - concepts and facilities. RFC 1034, Listopad 1987, doi:10.17487/RFC1034. Dostupné z: https://www.rfc-editor.org/info/rfc1034

[2] Domain names - implementation and specification. RFC 1035, Listopad 1987, doi:10.17487/RFC1035. Dostupné z: https://www.rfc-editor.org/info/rfc1035

[3] Cloudflare: What is a DNS A record? 2024. Dostupné z: https://www.cloudflare.com/ru-ru/learning/dns/dns-records/dns-a-record/

[4] Cloudflare: What is a DNS AAAA record? 2024. Dostupné z: https://www.cloudflare.com/ru-ru/learning/dns/dns-records/dns-aaaa-record/

[5] dnsimple: SRV Records. 2024. Dostupné z: https://support.dnsimple.com/articles/srv-record/

[6] Dolejska, D.: Počítačové komunikace a sítě. 2024. Dostupné z: https://git.fit.vutbr.cz/NESFIT/dev-envs

[7] Google: About CNAME records. 2024. Dostupné z: https://support.google.com/a/answer/112037?hl=en#zippy=%2Cset-up-cname-records-now

[8] Gulbrandsen, A.; Esibov, D. L.: A DNS RR for specifying the location of services (DNS SRV). RFC 2782, Únor 2000, doi:10.17487/RFC2782. Dostupné z: https://www.rfc-editor.org/info/rfc2782

[9] Ksinant, V.; Huitema, C.; Thomson, D. S.; aj.: DNS Extensions to Support IP Version 6. RFC 3596, Říjen 2003, doi:10.17487/RFC3596. Dostupné z: https://www.rfc-editor.org/info/rfc3596

[10] mailtrap: DNS MX Records Explained. 202š. Dostupné z: https://mailtrap.io/blog/dns-mx-records/

[11] Matoušek, P.: *Síťové služby a jejich architektura*. Publishing house of Brno University of Technology VUTIUM, 2014, ISBN 978-80-214-3766-1, 396 s. Dostupné z: https://www.fit.vut.cz/research/publication/10567

[12] Matoušek, P.: *Síťové služby a jejich architektura*. Publishing house of Brno University of Technology VUTIUM, 2014, ISBN 978-80-214-3766-1, 111-112 s. Dostupné z: https://www.fit.vut.cz/research/publication/10567

[13] Matoušek, P.: *Síťové služby a jejich architektura*. Publishing house of Brno University of Technology VUTIUM, 2014, ISBN 978-80-214-3766-1, 114-115 s. Dostupné z: https://www.fit.vut.cz/research/publication/10567

[14] MenandMice: What is a NS Record? 2024. Dostupné z: https://www.menandmice.com/glossary/dns-ns-record

[15] of Technology, B. U.: Počítačové komunikace a sítě. 2024. Dostupné z: https://www.fit.vut.cz/study/course/IPK/.cs