

# VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



Projekt ISA 2023/2024

## **DNS resolver**

# Obsah

<b>1</b>	<b>Úvod do problematiky</b>	<b>2</b>
<b>2</b>	<b>Návrh aplikace</b>	<b>2</b>
2.1	Zpracování Argumentů . . . . .	2
2.2	Odeslání DNS Dotazu . . . . .	2
2.3	Příjem a Zpracování Odpovědi . . . . .	2
<b>3</b>	<b>Popis implementace</b>	<b>3</b>
3.1	Použité Knihovny . . . . .	3
3.2	Struktura projektu . . . . .	3
<b>4</b>	<b>Návod na použití</b>	<b>4</b>
4.1	Parametry . . . . .	4
4.2	Příklady použití . . . . .	4

# 1 Úvod do problematiky

Domain Name System (DNS) je klíčovým prvkem internetové infrastruktury, který překládá srozumitelné lidské názvy domén (např. `www.example.com`) na IP adresy, které jsou potřebné pro směrování síťových paketů. DNS je zásadní pro fungování moderního internetu, umožňuje nám používat srozumitelné doménové názvy místo zapamatování IP adres.

V tomto kontextu vzniká potřeba pro implementaci DNS klienta, který je schopen zasílat dotazy na DNS servery a zpracovávat přijaté odpovědi. Implementovaný DNS klient bude schopen pracovat s protokolem UDP, který je běžně používán pro komunikaci mezi DNS klienty a servery.

Cílem tohoto projektu je vytvořit nástroj, který umožní uživatelům zasílat DNS dotazy a získávat odpovědi v čitelné podobě. DNS klient nabídne několik klíčových funkcí, včetně možnosti specifikovat rekurzivní nebo non-rekurzivní dotazy, provádět reverzní dotazy a vybírat mezi dotazy typu A (IPv4) a AAAA (IPv6).

Implementace DNS klienta vyžaduje schopnost sestavovat DNS dotazy v souladu s DNS protokolem, odesílat je na specifikovaný DNS server pomocí UDP a následně analyzovat a interpretovat přijaté DNS odpovědi. Kromě toho musí klient umožňovat specifikaci IP adresy nebo doménového jména cílového serveru a dotazované adresy.

## 2 Návrh aplikace

### 2.1 Zpracování Argumentů

Zpracování argumentů probíhá v funkcích `parseCommandLineArguments` a `checkArgument`. Funkce projde všechny argumenty příkazové řádky a identifikuje přepínače a jejich hodnoty. Pokud dojde k chybě, program vypíše chybové hlášení a ukončí se. Přepínače jsou následně nastaveny v globální proměnné `flags` a odpovídající proměnné pro uchování hodnot, jako jsou adresa serveru a číslo portu.

### 2.2 Odeslání DNS Dotazu

Funkce `performDNSServerCommunication` zajišťuje komunikaci se DNS serverem. Nejprve se získají informace o serveru a následně se vytvoří socket pro komunikaci. Procházejí se všechny dostupné adresy serveru a připojuje se k první dostupné adrese. Poté se sestaví a odešle DNS dotaz na server pomocí funkce `sendDNSQuery`.

### 2.3 Příjem a Zpracování Odpovědi

Po odeslání dotazu se volá funkce `receiveAndProcessDNSAnswer`, která přijímá odpověď od serveru pomocí funkce `receiveDNSPacket`. Přijatá odpověď se následně analyzuje a zpracuje ve funkci `processAndPrintDNSAnswer`. Tato funkce vypisuje informace o DNS odpovědi, včetně údajů o autoritativnosti, rekurzi, případně dalších sekcích jako otázky, odpovědi, autority a doplňkové informace.

## 3 Popis implementace

Zdrojový kód této implementace je napsán v programovacím jazyce C++. Překlad lze provést pomocí makefile, který je součástí projektu.

### 3.1 Použité Knihovny

- `<vector>`: Knihovna pro práci s vektory, což jsou dynamická pole, která jsou v tomto případě využívána pro uchovávání DNS odpovědí a jiných datových struktur.
- `<netdb.h>`: Poskytuje funkce pro práci s DNS (Domain Name System), zejména pro získání informací o DNS serveru.
- `<unistd.h>`: Obsahuje funkce pro systémová volání, například pro získání identifikátoru procesu.
- `<arpa/inet.h>`: Poskytuje funkce pro práci s adresami a protokoly, například pro převod IP adres na řetězce a zpět.

### 3.2 Struktura projektu

- `dns.h`: Hlavičkový soubor obsahující deklarace funkcí a struktur použitých v kódu. Definuje také struktury pro zpracování DNS hlavičky, otázky a odpovědi.
- `dns.cpp`: Implementační soubor obsahující hlavní logiku aplikace. Zahrnuje zpracování příkazové řádky, komunikaci se DNS serverem a zpracování odpovědí.
- `Makefile`: Soubor pro automatizaci procesu překladu a sestavení aplikace.
- `README`: Základní informace o projektu.
- `manual.pdf`: Dokumentace.
- `test_dns_resolver.py`: Testy.

## 4 Návod na použití

Použití: `dns [-r] [-x] [-6] -s server [-p port] adresa`

### 4.1 Parametry

Pořadí parametrů je libovolné.

- `-r`: Požadována rekurze (Recursion Desired = 1), jinak bez rekurze.
- `-x`: Reverzní dotaz místo přímého.
- `-6`: Dotaz typu AAAA místo výchozího A.
- `-s`: IP adresa nebo doménové jméno serveru, kam se má zaslat dotaz.
- `-p port`: Číslo portu, na který se má poslat dotaz, výchozí 53.
- `adresa`: Dotazovaná adresa.

### 4.2 Příklady použití

**`./dns -r -s kazi.fit.vutbr.cz example.com`**

Authoritative: No, Recursive: Yes, Truncated: No

Question section (1)

example.com., A, IN

Answer section (1)

example.com., A, IN, 83319, 93.184.216.34

Authority section (0)

Additional section (0)

**`./dns -r -6 -s kazi.fit.vutbr.cz www.fit.vut.cz`**

Authoritative: Yes, Recursive: Yes, Truncated: No

Question section (1)

www.fit.vut.cz., AAAA, IN

Answer section (1)

www.fit.vut.cz., AAAA, IN, 14400, 2001:67c:1220:809::93e5:91a

Authority section (0)

Additional section (0)