

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

Dokumentace do předmětu ISA
Projekt Whois tazatel

Obsah

1	Úvod	2
2	Teoretický základ	2
2.1	Služba WHOIS	2
2.2	Služba DNS	2
3	Implementace	3
3.1	Kontrola argumentů	3
3.2	DNS dotazování	3
3.2.1	DNS dotazování při zadaném hostname	3
3.2.2	Reverzní lookup	4
3.3	Práce s dotazováním WHOIS serveru	4
3.4	Bonusové části zadání	4
4	Testování projektu	4
4.1	Testování přímého DNS lookupu a WHOIS serveru s daným záznamem	5
4.2	Testování reverzního IPv4 DNS lookupu a WHOIS serveru s odkazem na jiný server	5
4.3	Testování reverzního IPv6 DNS lookupu	6
4.4	Testování WHOIS serveru whois.arin.net	6
4.5	Testování WHOIS serveru whois.nic.cz	7
5	Testování obdobných WHOIS vyhledávačů	8
5.1	Testování dotazu na adresu IPv6	8
5.2	Testování dotazu na adresu IPv4	9
6	Testování obdobných DNS vyhledávačů	10
7	Závěr	11
8	Přílohy	12
8.1	Příloha A	12

1 Úvod

Program Whois tazatel je cílem projektu, který se zaměřuje na implementaci síťové služby v předmětu Síťové aplikace a správa sítí. Program má být implementován v jazyce C/C++. Spouští se s následujícími argumenty:

```
./isa-tazatel -q DOTAZOVANA_ADRESA [-d DNS_SERVER] -w WHOIS_SERVER
```

DOTAZOVANA_ADRESA může být hostname, IPv4 nebo IPv6, jedná se o adresu, na kterou se budeme dotazovat a hledat o ní potřebné informace.

DNS_SERVER může být ve formátu IPv4 nebo IPv6. Pokud je tento argument nezadaný, DNS adresou bude zvolena adresa DNS serveru nakonfigurovaná v operačního systému.

WHOIS_SERVER může být hostname, IPv4 nebo IPv6. Jedná se o adresu WHOIS serveru.

Pořadí argumentů je libovolné.

Pro spuštění nápovědy je potřeba zadat příkaz ve tvaru: `./isa-tazatel --help`

V dokumentaci jsou zahrnuty potřebné informace o základních krocích v problematice tvorby projektu, příklady testování a popis bonusových částí zadání.

2 Teoretický základ

2.1 Služba WHOIS

WHOIS je skupina databází rozmístěných po celém světě a fungujících na určité hierarchii. Základem WHOIS registru je záznam informací o hledané IPv4 nebo IPv6 adrese. Na základě tohoto vyhledávání se můžeme dozvědět informace o vlastníkovi, rozsah přidělených IP adres, adresu, telefonní číslo, jméno organizace a další údaje. Pro projekt jsou důležité tyto údaje z WHOIS serveru:

- `inetnum` - rozsah přidělených IP adres
- `netname` - název pro rozsah přidělených IP adres
- `descr` - název hledané organizace
- `country` - země
- `address` - adresa
- `phone` - telefon
- `admin-c` - označení administrátora

Hierarchie WHOIS registrů je zobrazena v příloze. Je nutno podotknout, že některé registry nemusí informace o naší adrese obsahovat, nebo mohou obsahovat např. pouze záznam `inetnum`, ale mohou odkazovat ve svém záznamu na adresu jiného WHOIS serveru, kde můžeme informace o naší zvolené adrese dále hledat.

2.2 Služba DNS

DNS je globálním adresářem identifikátorů síťových služeb a zařízení. Systém DNS je hierarchicky členěn do invertovaného stromu. Do datového prostoru jsou ukládány informace o adresách ve formě DNS záznamů. Pro náš projekt byly důležité tyto typy záznamů[1]:

- `A` - přímé mapování doménové adresy na IP adresu ve formátu IPv4
- `AAAA` - přímé mapování doménové adresy na IP adresu ve formátu IPv6

- MX - informace o poštovním serveru, který pro danou doménu přijímá poštu
- CNAME - oznamuje, že daná doména je aliasem domény jiné v tom smyslu, že mají společné DNS záznamy
- NS - autoritativní server nebo servery pro danou doménu
- SOA - obsahuje informace týkající se uložení autoritativních dat pro danou zónu, v programu jsou vypisované tyto údaje:
 - MNAME - jméno primárního serveru DNS pro danou doménu
 - RNAME - kontakt na správce domény (e-mailová adresa), zavináč je v tomto záznamu nahrazen za tečku, proto musí být při vypsání `admin email` provedena úprava
- PTR - provádí zpětné (rekurzivní) mapování - převod číselných IP adres na doménové jméno

3 Implementace

Svůj návrh jsem si rozdělila na několik postatných částí, nejprve jsem začala jednodušší částí ohledně WHOIS dotazování a pak jsem se začala věnovat práci s DNS. Ke konci implementace jsem vyřešila i problém reverzního DNS lookupu.

3.1 Kontrola argumentů

Parsování argumentů bylo provedeno pomocí funkce `getopt()` s využitím konstrukce `switch`, ve které kontrolyuji správnost přepínačů. Dále se musím ujistit, že počet argumentů je správný a také, že je povinně zadán argument dotazované adresy a WHOIS serveru. Případně zpracuji nápovědu `--help`. S využitím funkce `inet_pton()` ověřím, zda je na vstupu IPv4 adresa, IPv6 adresa nebo hostname. Pokud uživatel zadá hostname, musím si jeho existenci ověřit ještě zpětným převedením na IP adresu. Při zadaném argumentu DNS serveru dále kontrolyuji, zda uživatel zadal pouze validní IPv4 nebo IPv6 adresu. V případě zadání IPv6 adresy si dále musím ošetřit IPv6 adresy i u dalších parametrů. Všechny vstupní parametry ukládám do struktury `input_data`.

3.2 DNS dotazování

Při této části projektu jsem využívala funkcí pro práci s DNS v knihovně `resolv.h`. Na samotném začátku dotazování na server DNS je potřeba nainicializovat strukturu `res_init()`, která při zjišťování nakonfigurovaného DNS serveru využívá souboru v operačním systému `resolv.conf` uloženým v adresáři `/etc`. Tato funkce sama zajistí naplnění struktury `dat_res`, se kterou se bude pracovat při dotazování. Pokud uživatel však zadá ve vstupu programu adresu DNS serveru, je potřeba pole v struktuře přepsat zvolenou IP adresou v decimálním formátu. Také je třeba nastavit počet dotazovaných adres pouze na tuto jednu jedinou, protože kdybychom zadali nevalidní adresu DNS serveru, tak by si systém po chybě vzal další adresu v pořadí a odpověď na dotaz by byla provedena absolutně jiným DNS serverem, než jakým jsme vyžadovali.

3.2.1 DNS dotazování při zadaném hostname

Při DNS dotazu, kdy víme hostname adresy, o které chceme zjistit informace, si inicializujeme buffer pro odpověď od DNS serveru a pomocí funkce `res_query()` vyšleme dotaz na námi zvolený záznam (pokud se např. ptáme na záznam A, tak využijeme parametru `ns_t_a`, při záznamu SOA pak parametr `ns_t_soa` atd.). Po důležitém zavolání funkce `ns_initparse()` začneme zpracovávat daný záznam nebo záznamy, procházíme je, kopírujeme je do zadaných struktur. U záznamů A pak využívám funkci `inet_ntoa()` pro IPv4 adresy, u záznamů IPv6 u AAAA pak funkci `inet_ntop()`. Pro kopírování do zadaných struktur pak využívám `memcpy()` a struktur `in_addr` nebo `in6_addr`, u dalších záznamů pak využívám `ns_sprintrr()`. Poté výslednou strukturu zkonvertuji na řetězec, upravím ho a vypíši potřebné informace.

3.2.2 Reverzní lookup

Když nám uživatel zadá dotazovanou IP adresu místo hostname, je potřeba ve správném formátu poslat dotaz na záznam PTR. Pokud se např. dotazujeme na IPv4 adresu 147.229.176.14, pak musíme zvolenému DNS serveru poslat dotaz ve formě 14.176.229.147.in-addr.arpa. Z PTR záznamu jsem si pak zjistila hostname, který je přiřazen dané IP adrese a použila ho v další práci při hledání ostatních záznamů. Práce s IPv6 je složitější a je komplikovanější vytvořit reverzní IPv6 adresu. Při tomto úkonu jsem však našla potřebný zdroj na internetu, který jsem s odkazem na autora a dostupnost umístila do svého kódu. Výstup však bylo nutné ještě upravit a následně zformátovat. Pokud máme např. IPv6 adresu 2001:db8::567:89ab, výsledný dotaz pro DNS server bude b.a.9.8.7.6.5.0.0.0.0.0.0.0.0.0.0.0.0.0.0.0.8.b.d.0.1.0.0.2.ip6.arpa.[3] Znovu jsem si z záznamu PTR zjistila dané hostname a dále jsem pokračovala při hledání dalších záznamů.

3.3 Práce s dotazováním WHOIS serveru

Při části projektu zaměřující se na WHOIS jsem využila klasického TCP socketu, kterému jsem zadala port, na kterém naslouchají všechny WHOIS servery (43) a dále jeho IPv4 nebo IPv6 adresu. Po připojení na server jsem odeslala IP hledané adresy ve zprávě formy "%s\r\n". Odpověď jsem si postupně ukládala do bufferu a nakonec jsem si celou odpověď serveru vložila do konstrukce `istringstream`. S touto odpovědí jsem dále pracovala a hledala, zda odpověď neobsahuje náhodou nějaké jiné WHOIS servery, kde by se záznam o dané adrese mohl nacházet, protože ne každý server má všechny informace. Po nalezení relevantních záznamů (`inetnum`, `netname`, `descr`, atd.) jsem zadané informace upravila a vypsala. Některé WHOIS servery (např. `whois.arin.net`) však příliš nedodržují strukturu jednotlivých výpisů (např. místo `inetnum` záznamu mají totožný záznam `NetRange`), proto bylo potřeba upravit i tuto skutečnost. Pokud ani přes dotazování dalších možných serverů záznam nenajdu, vypíšu uživateli informační hlášení.

Vzhledem k tomu, že nějaké WHOIS databáze berou dotaz na hostname adresy a ne na IP (např. česká databáze `whois.nic.cz`), bylo potřeba ošetřit i tuto zkušenost, ačkoliv regionální registry typu `whois.ripe.net`, `whois.arin.net` nebo `whois.lacnic.net` mají klasickou zprávu ve formě IP adresy. Ve svém projektu se proto kvůli těmto rozdílům dotazuji na WHOIS jak hledaného hostname, tak hledané IP. Je to zapříčiněné hlavně touto nejednoznačností, kdy není pravidlem, že bude každá databáze přijímat dotaz ve formátu buď IP adresy, nebo hostname.

3.4 Bonusové části zadání

Jelikož jsem projekt začala řešit již od prvního týdne, kdy jsem ho měla zapsaný, brala jsem některé části v zadání jako spíše samozřejmost, ačkoliv se poté třeba už na základě fóra nakonec v základu nemuseli implementovat. Při práci s WHOIS jsem implicitně uvažovala podporování rekurzivního dotazování na položky záznamu, pokud nás odpověď serveru odkázala na nějaký jiný, kde by se mohla nacházet daná odpověď s co nejvíce dostupnými informacemi. V záznamu tedy hledám vždy před vypsáním při nedostatku informací některé zmínky o referenčním serveru a pokud ho najdu, připojím se k němu. V první verzi zadání dále bylo napsáno, že implicitním DNS serverem při zadání argumentu není DNS server operačního systému, ale adresa 1.1.1.1. Jelikož jsem byla při uvedení jiné DNS adresy než adresy operačního systému v pokročilé části řešení projektu, již jsem věděla, jak přepsat strukturu `_res`. Proto podporuji i zadání argumentu `-d` uživatelem.

4 Testování projektu

Projekt byl otestován na mém systému Ubuntu 18.04.3 LTS, dále na referenční virtuálce a na serveru `merlin.fit.vutbr.cz`. Testování probíhalo průběžně, pomáhalo při hledání chyb a nedostatků v kódu.

4.1 Testování přímého DNS lookupu a WHOIS serveru s daným záznamem

```
$ ./isa-tazatel -q www.fit.vutbr.cz -w whois.ripe.net
```

```
=== DNS ===
```

```
A:                147.229.9.23
AAAA:             2001:67c:1220:809::93e5:917
MX:               tereza.fit.vutbr.cz
```

```
=== WHOIS query for IP ===
```

```
inetnum:          147.229.0.0 - 147.229.254.255
netname:          VUTBRNET
descr:            Brno University of Technology
country:          CZ
admin-c:          CA6319-RIPE
address:           Brno University of Technology
address:           Antoninska 1
address:           601 90 Brno
address:           The Czech Republic
phone:            +420 541145453
phone:            +420 723047787
descr:            VUTBR-NET1
```

```
=== WHOIS query for hostname ===
```

```
Parent WHOIS server didn't find any relevant information about searched hostname!
```

4.2 Testování reverzního IPv4 DNS lookupu a WHOIS serveru s odkazem na jiný server

```
$ ./isa-tazatel -q 77.75.75.176 -w whois.iana.org -d 8.8.8.8
```

```
=== DNS ===
```

```
A:                77.75.74.176
A:                77.75.75.172
A:                77.75.75.176
A:                77.75.74.172
AAAA:             2a02:598:4444:1::1
AAAA:             2a02:598:4444:1::2
AAAA:             2a02:598:3333:1::1
AAAA:             2a02:598:3333:1::2
PTR:              www.seznam.cz
```

```
=== WHOIS query for IP ===
```

```
inetnum:          77.75.75.0 - 77.75.75.255
netname:          SEZNAM-CZ
descr:            Seznam.cz
country:          CZ
admin-c:          SZN5-RIPE
address:           Radlicka 3294/10 150 00 Prague 5 Czech Republic
phone:            +420 602 126 570
admin-c:          PZ172-RIPE
descr:            SEZNAM - II
```

=== WHOIS query for hostname ===

Parent WHOIS server didn't find any relevant information about searched hostname!

4.3 Testování reverzního IPv6 DNS lookupu

```
$ ./isa-tazatel -q 2001:67c:1220:9847::93e5:471c -w whois.iana.org
```

=== DNS ===

```
A:                147.229.71.28
AAAA:             2001:67c:1220:9847::93e5:471c
PTR:              www.fekt.vutbr.cz
```

=== WHOIS query for IP ===

```
inet6num:         2001:67c:1220::/46
netname:          VUTBR-TCZ
country:          CZ
descr:            ORG-BUOT1-RIPE
admin-c:          MS6207-RIPE
address:          Antoninska 548/1
address:          60190
address:          Brno
address:          CZECH REPUBLIC
phone:            +420541145453
address:          Brno University of Technology
address:          Antoninska 1
address:          Brno
address:          601 90
address:          The Czech Republic
phone:            +420 541 145 441
address:          Brno University of Technology
address:          Center of Computing and Information Services
address:          Antoninska 1
address:          Brno
address:          601 90
address:          The Czech Republic
phone:            +420 541145630
descr:            VUTBR6-NET
```

=== WHOIS query for hostname ===

Parent WHOIS server didn't find any relevant information about searched hostname!

4.4 Testování WHOIS serveru whois.arin.net

```
$ ./isa-tazatel -q facebook.com -w whois.arin.net
```

=== DNS ===

```
A:                157.240.30.35
AAAA:             2a03:2880:f13d:83:face:b00c:0:25de
MX:               smtpin.vvv.facebook.com
NS:               b.ns.facebook.com
NS:               a.ns.facebook.com
```

```
SOA:                a.ns.facebook.com
admin email:        dns@facebook.com
```

```
=== WHOIS query for IP ===
inetnum:            157.240.0.0 - 157.240.255.255
netname:             THEFA-3
descr:              Facebook, Inc. (THEFA-3)
address:            1601 Willow Rd.
country:            US
admin-c:            OPERA82-ARIN
phone:              +1-650-543-4800
```

```
=== WHOIS query for hostname ===
Parent WHOIS server didn't find any relevant information about searched hostname!
```

4.5 Testování WHOIS serveru whois.nic.cz

```
$ ./isa-tazatel -q centrum.cz -w whois.nic.cz
=== DNS ===
```

```
A:                  46.255.231.106
AAAA:               2a00:da80:f::106
MX:                 cax.virusfree.cz
MX:                 cbx.virusfree.cz
MX:                 cdx.spamfree.cz
MX:                 ccx.spamfree.cz
NS:                 max.ns.cloudflare.com
NS:                 pam.ns.cloudflare.com
SOA:                max.ns.cloudflare.com
admin email:        dns@cloudflare.com
```

```
=== WHOIS query for IP ===
Parent WHOIS server didn't find any relevant information about searched IP!
```

```
=== WHOIS query for hostname ===
admin-c:            CENTRUM-CZ
descr:              Economia, a.s.
address:            Pernerova 47
address:            Praha 8
address:            18600
address:            CZ
descr:              CZ.NIC, z.s.p.o.
address:            Milesovska 1136/5
address:            Praha 3
address:            130 00
address:            CZ
```

```
$ ./isa-tazatel -q seznam.cz -w 2001:1488::2:0:0:0:2
=== DNS ===
```

```
A:                  77.75.75.172
A:                  77.75.75.176
```



```
AAAA:          2a02:598:4444:1::2
AAAA:          2a02:598:4444:1::1
MX:            mx1.seznam.cz
MX:            mx2.seznam.cz
NS:            ans.seznam.cz
NS:            ams.seznam.cz
SOA:           ans.seznam.cz
admin email:   hostmaster@seznam.cz
```

=== WHOIS query for IP ===

Parent WHOIS server didn't find any relevant information about searched IP!

=== WHOIS query for hostname ===

```
admin-c:       SEZNAM-CZ-AS-TECH
descr:         Seznam.cz , a.s.
address:       Radlicka 3294/10
address:       Praha 5
address:       15000
address:       CZ
descr:         Seznam.cz , a.s.
address:       Radlicka 3294/10
address:       150 00
address:       CZ
```

5 Testování obdobných WHOIS vyhledávačů

Při porovnání jsem využila příkazu `whois`. Ve výpisu jsou obsažena pouze pole záznamů, které jsme měli v projektu vyhledat.

5.1 Testování dotazu na adresu IPv6

Testování dotazu na adresu `2001:67c:1220:9847::93e5:471c`:

```
$ whois 2001:67c:1220:9847::93e5:471c
% This is the RIPE Database query service.
% The objects are in RPSL format.
%
% The RIPE Database is subject to Terms and Conditions.
% See http://www.ripe.net/db/support/db-terms-conditions.pdf

% Note: this output has been filtered.
%       To receive output for a database update, use the "-B" flag.

% Information related to '2001:67c:1220::/46'

% Abuse contact for '2001:67c:1220::/46' is 'abuse@vutbr.cz'

inet6num:      2001:67c:1220::/46
netname:       VUTBR-TCZ
country:       CZ
```

org: ORG-BUOT1-RIPE
admin-c: MS6207-RIPE

address: Antoninska 548/1
address: 60190
address: Brno
address: CZECH REPUBLIC
phone: +420541145453

address: Brno University of Technology
address: Antoninska 1
address: Brno
address: 601 90
address: The Czech Republic
phone: +420 541 145 441

address: Brno University of Technology
address: Center of Computing and Information Services
address: Antoninska 1
address: Brno
address: 601 90
address: The Czech Republic
phone: +420 541145630

% Information related to '2001:67c:1220::/46 AS197451'

descr: VUTBR6-NET

% This query was served by the RIPE Database Query Service version 1.95.1 (WAGYU)

5.2 Testování dotazu na adresu IPv4

Testování dotazu na adresu 77.75.75.176:

\$ whois 77.75.75.176

% This is the RIPE Database query service.

% The objects are in RPSL format.

%

% The RIPE Database is subject to Terms and Conditions.

% See <http://www.ripe.net/db/support/db-terms-conditions.pdf>

% Note: this output has been filtered.

% To receive output for a database update, use the "-B" flag.

% Information related to '77.75.75.0 - 77.75.75.255'

% Abuse contact for '77.75.75.0 - 77.75.75.255' is 'abuse@seznam.cz'

inetnum: 77.75.75.0 - 77.75.75.255
netname: SEZNAM-CZ
descr: Seznam.cz

```
country :      CZ
admin-c :     SZN5-RIPE
tech-c :      SZN5-RIPE

address :     Radlicka 3294/10 150 00 Prague 5 Czech Republic
phone :       +420 602 126 570
admin-c :     PZ172-RIPE
```

```
% Information related to '77.75.75.0/24 AS43037'
```

```
descr :       SEZNAM - II
```

```
% This query was served by the RIPE Database Query Service version 1.95.1 (WAGYU)
```

6 Testování obdobných DNS vyhledávačů

Porovnání DNS výstupu mého projektu jsem porovnála s příkazem `dig`. Opět jsem vybrala pouze záznamy, co byly pro projekt klíčové.

```
$ dig www.fit.vutbr.cz any
```

```
; <<>> DiG 9.11.3-1ubuntu1.9-Ubuntu <<>> www.fit.vutbr.cz any
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 52823
;; flags: qr rd ra; QUERY: 1, ANSWER: 8, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;www.fit.vutbr.cz.                IN      ANY

;; ANSWER SECTION:
www.fit.vutbr.cz.                5      IN      MX      0 tereza.fit.vutbr.cz.
www.fit.vutbr.cz.                5      IN      AAAA    2001:67c:1220:809::93e5:917
www.fit.vutbr.cz.                5      IN      A       147.229.9.23

;; Query time: 4 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Mon Oct 28 20:10:04 CET 2019
;; MSG SIZE rcvd: 839
```

```
$ dig facebook.com any
```

```
; <<>> DiG 9.11.3-1ubuntu1.9-Ubuntu <<>> facebook.com any
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 39808
;; flags: qr rd ra; QUERY: 1, ANSWER: 9, AUTHORITY: 0, ADDITIONAL: 1
```

```
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags::; udp: 65494
;; QUESTION SECTION:
;facebook.com.                IN      ANY

;; ANSWER SECTION:
facebook.com.  5      IN      SOA      a.ns.facebook.com. dns.facebook.com.
facebook.com.  5      IN      MX       10 smtpin.vvv.facebook.com.
facebook.com.  5      IN      AAAA     2a03:2880:f13d:83:face:b00c:0:25de
facebook.com.  5      IN      A        157.240.30.35
facebook.com.  5      IN      NS       b.ns.facebook.com.
facebook.com.  5      IN      NS       a.ns.facebook.com.

;; Query time: 8 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Mon Oct 28 20:22:35 CET 2019
;; MSG SIZE rcvd: 395
```

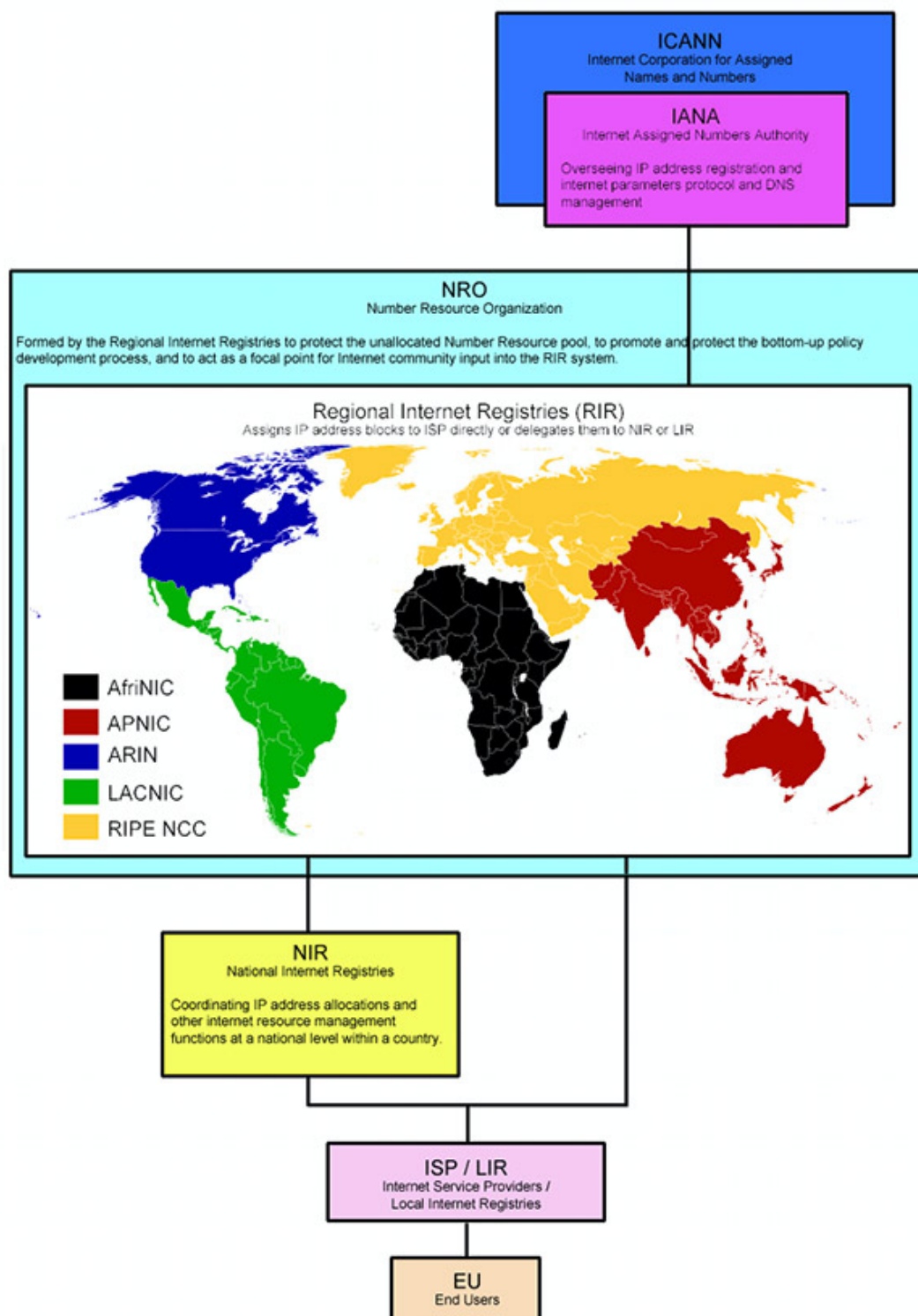
7 Závěr

Výsledky testování mého projektu s podobným WHOIS vyhledávačem dopadly dle mého uvážení dobře, hledané údaje sobě odpovídají. Při testování u DNS, při příkazu `dig`, jsem narazila na skutečnost, že při zadání zprávy při reverzním lookupu se nám dostane pouze záznamu PTR, tudíž v mém projektu spatřuji výhodu, protože přímo dokáže z tohoto záznamu zjistit hostname a hned uživateli poté vypsát všechny hledané informace. Můj projekt dokázal vyhledat stejné údaje o hledaných adresách. K projektu jsem přistupovala již od chvíle zadání poměrně zodpovědně a snažila se ho nenechat na poslední chvíli. Zpočátku jsem se zadání a vypracování projektu obávala, ale poté jsem zjistila, že mě práce na projektu opravdu baví, také i kvůli jeho samotnému zajímavému zadání.

8 Přílohy

8.1 Příloha A

Obrázek znázorňující řazení světových WHOIS registrů[2]:



Reference

- [1] MATOUŠEK, P. *Síťové aplikace a jejich architektura*. 1. vyd. Brno: VUTIUM, 2014. ISBN 978-80-214-3766-1.
- [2] MOBILEFISH.COM. *Whois servers list* [online]. Poslední změna 26. října 2019 [cit. 26. října 2019]. Dostupné na: <<https://www.mobilefish.com/images/tutorials/whois-icann-iana-rir.jpg>>.
- [3] WIKIPEDIA. *Reverse DNS lookup* [online]. Poslední změna 25. října 2019 [cit. 25. října 2019]. Dostupné na: <https://en.wikipedia.org/wiki/Reverse_DNS_lookup>.