

VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ

BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH TECHNOLOGIÍ
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

GEOGRAFICKÁ POZICE SERVERŮ SÍTĚ PLANETLAB

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

PAVOL ILKO

BRNO 2014



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY



**FAKULTA ELEKTROTECHNIKY A KOMUNIKAČNÍCH
TECHNOLOGIÍ**
ÚSTAV TELEKOMUNIKACÍ

FACULTY OF ELECTRICAL ENGINEERING AND COMMUNICATION
DEPARTMENT OF TELECOMMUNICATIONS

GEOGRAFICKÁ POZICE SERVERŮ SÍTĚ PLANETLAB

GEOGRAPHIC LOCATION OF PLANETLAB SERVERS

BAKALÁŘSKÁ PRÁCE
BACHELOR'S THESIS

AUTOR PRÁCE
AUTHOR

PAVOL IL'KO

VEDOUCÍ PRÁCE
SUPERVISOR

doc. Ing. DAN KOMOSNÝ, Ph.D.

BRNO 2014



VYSOKÉ UČENÍ
TECHNICKÉ V BRNĚ

Fakulta elektrotechniky
a komunikačních technologií

Ústav telekomunikací

Bakalářská práce

bakalářský studijní obor
Teleinformatika

Student: Pavol Il'ko

ID: 134505

Ročník: 3

Akademický rok: 2013/2014

NÁZEV TÉMATU:

Geografická pozice serverů sítě PlanetLab

POKYNY PRO VYPRACOVÁNÍ:

Seznamte se s experimentální sítí PlanetLab – <http://www.planet-lab.org/>. Vytvořte seznam stanic sítě PlanetLab, které se nacházejí v Evropě. Pro tyto stanice zjistěte jejich polohu pomocí informací ze sítě PlanetLab. Sestavte program, který ověří správnost zjištěné polohy pomocí zvolených mapových podkladů.

DOPORUČENÁ LITERATURA:

- [1] PlanetLab Consortium. PlanetLab: An open platform for developing, deploying, and accessing planetary-scale services. URL: <<http://www.planet-lab.org>> [cit. 10. 10. 2011].
- [2] Cooper, M. Advanced Bash-Scripting Guide. Lulu.com, 2010. ISBN: 978-1435752191.
- [3] Linux Dokumentační projekt. Computer Press, 2008. ISBN: 978-80-251-1525-1.

Termín zadání: 10.2.2014

Termín odevzdání: 4.6.2014

Vedoucí práce: doc. Ing. Dan Komosný, Ph.D.

Konzultanti bakalářské práce:

doc. Ing. Jiří Mišurec, CSc.

Předseda oborové rady

UPOZORNĚNÍ:

Autor bakalářské práce nesmí při vytváření bakalářské práce porušit autorská práva třetích osob, zejména nesmí zasahovat nedovoleným způsobem do cizích autorských práv osobnostních a musí si být plně vědom následků porušení ustanovení § 11 a následujících autorského zákona č. 121/2000 Sb., včetně možných trestněprávních důsledků vyplývajících z ustanovení části druhé, hlavy VI. díl 4 Trestního zákoníku č.40/2009 Sb.

ABSTRAKT

Táto práca sa zaoberá presnosťou polohy staníc v experimentálnej sieti PlanetLab. Práca je rozdelená na teoretickú a praktickú časť. V teoretickej časti je opísaná sieť PlanetLab, jej stručná história a prebiehajúce projekty. Taktiež je v nej opísané vývojové prostredie Google Maps, ktoré využívam v praktickej časti. Praktická časť opisuje funkčnosť programu na overenie presnosti pozície, ale tiež program na meranie odozvy stanice na SSH, či mapu, na ktorej sú zobrazené stanice PlanetLab v Európe. Programy a namerané dáta sú priložené na CD.

KLÚČOVÉ SLOVÁ

PlanetLab, JavaScript, ssh, Google Maps API, skript, bash, geokódovanie

ABSTRACT

This thesis deals with accuracy of location of nodes in experimental network PlanetLab. Thesis is divided into a theoretical and a practical part. Theoretical part consists of the description of network PlanetLab, its brief history and current projects. At the same time, Google Maps framework, which is later employed in the practical part, is described. Practical part describes the functionality of the application aimed at validation of location accuracy. Next, the application measuring latency of node to SSH is also described, as well as the map which contains PlanetLab nodes located in Europe. Applications and obtained data are attached on CD disk.

KEYWORDS

PlanetLab, JavaScript, ssh, Google Maps API, script, bash, geocoding

PREHLÁSENIE

Prehlasujem, že som svoju bakalársku prácu na tému „Geografická pozice serverů sítě PlanetLab“ vypracoval samostatne pod vedením vedúceho bakalárskej práce, využitím odbornej literatúry a ďalších informačných zdrojov, ktoré sú všetky citované v práci a uvedené v zozname literatúry na konci práce.

Ako autor uvedenej bakalárskej práce ďalej prehlasujem, že v súvislosti s vytvorením tejto bakalárskej práce som neporušil autorské práva tretích osôb, najmä som nezasiahol nedovoleným spôsobom do cudzích autorských práv osobnostných a/nebo majetkových a som si plne vedomý následkov porušenia ustanovenia § 11 a nasledujúcich autorského zákona č. 121/2000 Sb., o právu autorskom, o právach súvisiacich s právom autorským a o zmene niektorých zákonov (autorský zákon), vo znení neskorších predpisov, vrátane možných trestnoprávných dôsledkov vyplývajúcich z ustanovenia časti druhej, hlavy VI. diel 4 Trestného zákoníka č. 40/2009 Sb.

Brno

.....

(podpis autora)

POĎAKOVANIE

Rád by som poďakoval vedúcemu bakalárskej práce pánovi doc. Ing. Dan Komosnému, PhD. za odborné vedenie, konzultácie, trpezlivosť a podnetné návrhy k práci. Taktiež by som rád poďakoval slečne Bc. Lýdii Rezničákovej za trpezlivosť a korekciu textu.

Brno

.....

(podpis autora)

POĎAKOVANIE

Výskum popísaný v tejto bakalárskej práci bol realizovaný v laboratóriách podporených z projektu SIX; registračné číslo CZ.1.05/2.1.00/03.0072, operačný program Výzkum a vývoj pro inovace.

Brno

.....
(podpis autora)

OBSAH

| | |
|--|-----------|
| Úvod | 10 |
| 1 PlanetLab | 11 |
| 1.1 História | 11 |
| 1.2 Projekty v rámci siete PlanetLab | 12 |
| 1.3 Zoznam staníc siete PlanetLab | 12 |
| 2 Google Maps API | 15 |
| 2.1 Static Maps API | 15 |
| 2.2 JavaScript API | 16 |
| 2.3 Geocoding API | 17 |
| 2.3.1 Formát požiadavky na Geocoding API | 17 |
| 2.3.2 Odpoveď Geocoding API | 18 |
| 3 Overenie presnosti pozície | 21 |
| 4 Odozva na ssh | 25 |
| 5 Zobrazenie polohy staníc na mape | 28 |
| 6 Analýza nameraných dát | 30 |
| 6.1 Overenie presnosti pozície | 30 |
| 6.2 Odozva na ssh | 30 |
| 7 Záver | 33 |
| Literatúra | 34 |
| Zoznam symbolov, veličín a skratiek | 35 |
| Zoznam príloh | 36 |
| A Obsah priloženého CD | 37 |

ZOZNAM OBRÁZKOV

| | | |
|-----|--|----|
| 1.1 | Detaily o organizácií. | 13 |
| 1.2 | Detaily o uzle. | 13 |
| 1.3 | Ukážka časti zoznamu staníc. | 14 |
| 2.1 | Obrázok vygenerovaný pomocou Static Maps API. | 15 |
| 3.1 | Vývojový diagram programu na určenie presnosti pozície. | 23 |
| 3.2 | Ukážka výpočtu presnosti pozície. | 24 |
| 4.1 | Vývojový diagram programu na meranie dostupnosti serveru. | 27 |
| 5.1 | Zobrazenie polohy staníc na mape. | 29 |
| 6.1 | Pravdepodobnosť počtu uzlov proti vzdialenosti medzi súradnicami a adresou organizácie. | 31 |
| 6.2 | Najkratší nameraný čas, za ktorý sa bolo možné pripojiť ku stanici. . | 31 |
| 6.3 | Priemerný čas potrebný na pripojenie ku stanici. | 32 |
| 6.4 | Najdlhší nameraný čas, za ktorý sa bolo možné pripojiť ku stanici. . . | 32 |

ZOZNAM TABULIEK

| | | |
|-----|-------------------------|----|
| 4.1 | Argumenty ssh | 26 |
|-----|-------------------------|----|

ÚVOD

V rámci bakalárskej práce som sa zoznámil s experimentálnou sieťou PlanetLab a opísal som ju v kapitole 1. V rámci tejto kapitoly som predstavil stručnú históriu siete PlanetLab a zaujímavé projekty, ktoré sa v sieti realizujú. Zadaním tejto práce bolo vytvoriť zoznam staníc siete, ktoré sa nachádzajú v Európe. Pre tieto stanice som zistil ich polohu pomocou informácií získaných zo siete PlanetLab. Opis získavania týchto údajov je možné nájsť v podkapitole 1.3.

Nasledujúca kapitola 2 sa venuje teoretickému úvodu k vývojovému prostrediu Google Maps. V podkapitole 2.1 som znázornil použitie Static Maps API a následne v podkapitole 2.2 som rozobral použitie dynamických máp s použitím JavaScriptu. Podkapitola 2.3 hovorí o využití Geocoding API, ktoré slúži na preklad adresy na zemepisné súradnice.

Geocoding API využívam v programe na overenie presnosti zistenej polohy, ktorý opisuje kapitola 3. Kapitola 3 približuje štruktúru aplikácie, ktorá porovnáva pozíciu určenú zemepisnými súradnicami zo siete PlanetLab s pozíciou určenou adresou inštitúcie.

V rámci bakalárskej práce som vytvoril skript na meranie odozvy serveru na ssh. Meral som čas, za ktorý bolo možné sa pripojiť a odpojiť zo serveru. Meranie odozvy detailne popisuje kapitola 4.

Kvôli potrebe pripájať sa na jednotlivé servery pod rôznymi účtami s rozdielnymi súkromnými kľúčmi som vytvoril interaktívnu mapu, na ktorej sú zobrazené stanice v sieti PlanetLab v Európe. Na mape je možné si zvoliť ktorý účet budeme používať, a ktorý súkromný kľúč. Po kliknutí na bod, kde je stanica umiestnená sa zobrazia informácie o stanici ako aj príkaz na vytvorenie ssh spojenia. Detaily sú zachytené v kapitole 5.

Na záver popisujem výsledky meraní naprogramovaných programov. Najskôr prezentujem výsledky programu na overenie presnosti pozície, a následne programu na meranie odozvy na ssh. Tieto výsledky sú dostupné v kapitole 6.

1 PLANETLAB

PlanetLab je celosvetová distribuovaná sieť serverov, z ktorých najväčšie množstvo serverov zabezpečujú výskumné organizácie, či univerzity. Zároveň je niekoľko z nich umiestnených v routovacích centách (napr. na projekte Internet2 v Abilene sieti¹). Všetky servery sú pripojené k internetu. Úlohou týchto serverov je poskytnúť testovacie prostredie na testovanie sieťových aplikácií a protokolov v reálnych podmienkach a sprístupnenie vyvíjaných služieb koncovým užívateľom [8].

Terminológia

- Organizácia – *Site* je fyzické miesto, kde sú umiestnené servery siete PlanetLab označované ako *nodes*.
- Uzol – *Node* je server, na ktorom prebiehajú súčasti služieb siete PlanetLab. *Node* musí mať aspoň jednu nezdieľanú IP adresu.
- Projekt – *Slice* je sada virtuálnych serverov distribuovaných cez sieť PlanetLab. Pre väčšinu užívateľov *Slice* znamená meno účtu unixového terminálu s prístupom na množstvo PlanetLab uzlov (*node*).
- Silver – je sada alokovaných zdrojov na jednom PlanetLab uzle.

1.1 História

V marci 2002 Larry Paterson z Princeton University a David Culler z Intel Research a University of California Berkeley zorganizovali stretnutie výskumných pracovníkov, ktorí sa zaujímali o globálne počítačové siete a navrhli PlanetLab ako komunitné testovacie prostredie. Na tomto stretnutí sa zúčastnilo 30 výskumníkov z MIT, Washington University, Rice, Berkeley, Princeton, Columbia, Duke, CMU a z Utahu. David Tennenhouse z Intel Research súhlasil so začiatočným financovaním projektu so 100 servermi [9]. V septembri 2003 už PlanetLab dosiahol na hranicu 200 uzlov, a už v decembri presiahol počet uzlov 300. V januári 2004 bolo formálne vytvorené PlanetLab konzorcium, teda skupina akademických, priemyselných a vládnych organizácií spolupracujúcich na podpore a raste siete PlanetLab. Odvtedy sa sieť PlanetLab masívne rozrástla. Aktuálne (máj 2014) pozostáva sieť z 1187 uzlov (*nodes*) v 585 organizáciách (*sites*) [9].

¹Bližšie informácie o projekte Internet2 sú dostupné na <http://www.internet2.edu/products-services/advanced-networking/>.

1.2 Projekty v rámci siete PlanetLab

Keďže jedným z cieľov siete PlanetLab je slúžiť ako testovacie prostredie, kde vývojárske skupiny môžu experimentovať s pestrou skupinou služieb ako je zdieľanie dát, vrstvenie QoS, detekcia anomálií v sieti a mnoho ďalších služieb. Výhodou siete PlanetLab je experimentovanie v reálnych podmienkach. V rámci siete PlanetLab beží aktuálne okolo 700 projektov [9].

Príklad aktuálnych projektov:

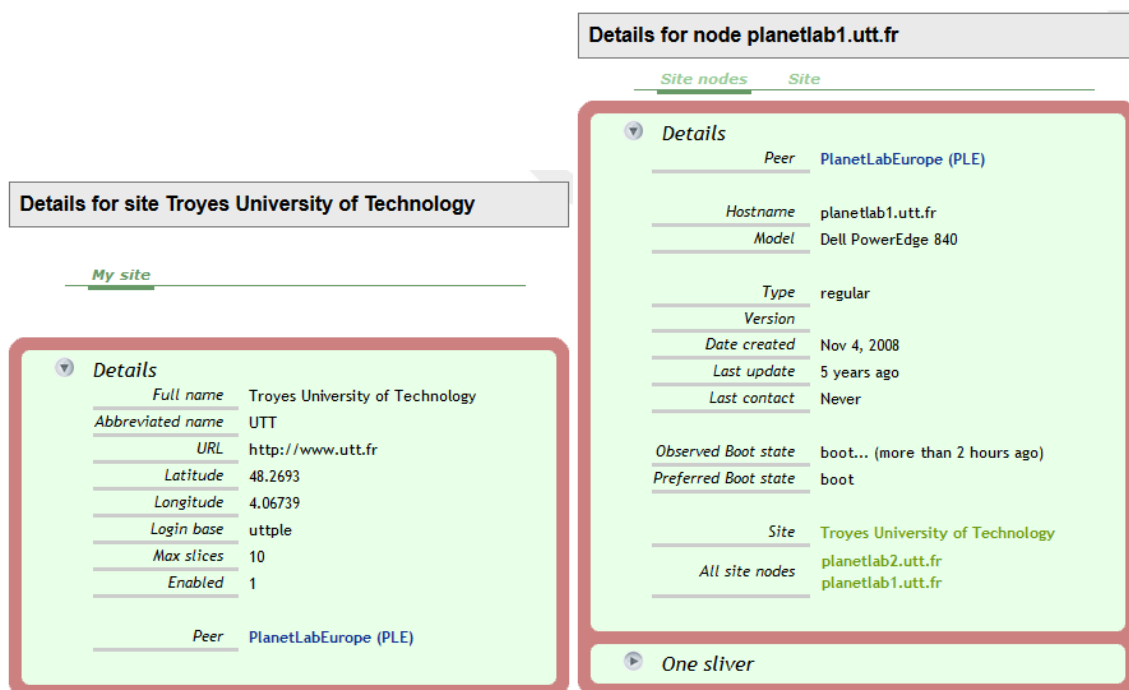
- A Moving Target DDoS Defense Mechanism – experimenty na potvrdenie obrany „pohyblivým objektom“ proti útokom na odoprenie služby (DOS útokom).
- Anonymous Publish Subscription – projekt, ktorý sa zameriava na vytvorenie systému, ktorý poskytne funkciu anonymného komunikačného systému na základe publish–subscribe modelu.
- Crypto-Book – experimentálny framework na budovanie kryptografických identít, ktoré zaručujú súkromie nad rámec už existujúcich nezabezpečených identít na sociálnych sieťach.
- OneSwarm – súkromie zabezpečujúca P2P sieť. V sieti PlanetLab prebiehajú výkonové testy zdieľania náhodne generovaných dát.

1.3 Zoznam staníc siete PlanetLab

Praktickou časťou tejto práce bolo vytvoriť zoznam staníc siete PlanetLab, ktoré sa nachádzajú v Európe. Spoločne s Jakubom Polášekom a Bc. Jánom Pružinským sme si túto úlohu rozdelili a každý získal potrebné informácie približne o tretine staníc. Celkovo sme vytvorili zoznam s 367 stanicami. Pre každú stanicu sme získali údaje so siete manuálne z webu <<https://www.planet-lab.org>>. Detaily o zemepisných súradniciach a o webovej stránke inštitúcie sme čerpali z detailov danej organizácie (*site*), ako je zobrazené na Obr. 1.1 a informácie o uzloch sme získali z detailov o uzle (*nodes*), ktoré možno vidieť na Obr. 1.2. Následne sme vyhľadávali fyzickú adresu inštitúcie na jej webových stránkach. Vyhľadané informácie sme pridávali do spoločnej tabuľky. Do tejto tabuľky som prispel ešte hodnotami v stĺpcoch Difference_Accuracy, SSH_Latency_Maximum, SSH_Latency_Average, SSH_Latency_Minimum. Tieto hodnoty sú výsledkom meraní overenia presnosti pozície a odozvy na SSH. Časť tabuľky je zobrazená na Obr. 1.3.

Problémy so spracovaním časti adries

Po prvom teste programu na určenie presnosti pozície, ktorý opisujem v kapitole 3, som zistil, že niektoré z adries neboli správne formátované a geokódovací systém



Obr. 1.1: Detaily o organizácií.

Obr. 1.2: Detaily o uzle.

ich nerozoznal správne. Ručne som preto prešiel všetky adresy, aby som skontroloval, či ich systém správne rozpoznáva. Keď bol geokódovací systém neúspešný, tak som vyhľadal na webe inštitúcie adresu a zmenil som jej formát tak, aby ju systém našiel. Častou chybou bolo, že v adrese sa vyskytovali informácie navyše viz Výpis kódu 1.1, hoci sa adresa zdá presnejšia, geokódovací systém nevrátil správnu odpoveď. Po odobratí nadbytočných informácií viz Výpis kódu 1.2 geokóder vráti správnu odpoveď.

Výpis kódu 1.1: Adresa s informáciami navyše

```
Dept._of_Information_Engineering,_Via_Vignolese,_905,_Modena,_Italy
,_41125,_Italy
```

Odpoveď geokodéra:

```
1 {
2   "results" : [],
3   "status" : "ZERO_RESULTS"
4 }
```

Výpis kódu 1.2: Správna adresa

```
Via_Vignolese,_905,_Modena,_Italy,_41125,_Italy
```

Odpoveď geokodéra:

```

1  ...
2  "formatted_address" : "Via Vignolese, 905, 41125 Modena, Italy",
3  "geometry" : {
4    "location" : {
5      "lat" : 44.6276204,
6      "lng" : 10.9480486
7    },
8    "location_type" : "ROOFTOP",
9    "viewport" : {
10     "northeast" : {
11       "lat" : 44.6289693802915,
12       "lng" : 10.9493975802915
13     },
14     "southwest" : {
15       "lat" : 44.6262714197085,
16       "lng" : 10.9466996197085
17     }
18   }
19 }
20 ...

```

| A | B | C | D | E | K | P | V | W | X |
|--------|-----------------|-------------------------------|--------------|---------------|--|--------------------------|-------------------------------|-------------------------------|-------------------------------|
| ID (1) | IP (2) | DNS (3) | Latitude (4) | Longitude (5) | Adresa (11) | Difference_Accuracy (16) | SSH_Latency_Maximum [ms] (22) | SSH_Latency_Average [ms] (23) | SSH_Latency_Minimum [ms] (24) |
| 10631 | 152.81.47.5 | host4-plb.loria.fr | 48.6661 | 6.15775 | 615_Rue_du_Jardin_Botanique_54600_Villers-lès-Nancy | 0.0936674 | 14926 | 4527 | 823 |
| 10632 | 152.81.47.4 | host3-plb.loria.fr | 48.6661 | 6.15775 | 615_Rue_du_Jardin_Botanique_54600_Villers-lès-Nancy | 0.0936674 | 28184 | 5060 | 906 |
| 10633 | 130.79.48.55 | planetlab1.u-strasbg.fr | 48.5237 | 7.73833 | Boulevard_Sébastien_Brant_300_Illkirch_67412_France | 0.0157344 | TimeOut | 4395 | 676 |
| 10634 | 130.79.48.57 | planetlab2.u-strasbg.fr | 48.5237 | 7.73833 | Boulevard_Sébastien_Brant_300_Illkirch_67412_France | 0.0157344 | 7749 | 4035 | 639 |
| 10635 | 193.191.148.227 | planck227ple.test.ibbt.be | 51.04 | 3.74 | 9050_Ghent-Ledeberg_Belgium | 0.5284928 | 17706 | 4691 | 849 |
| 10636 | 193.191.148.228 | planck228ple.test.ibbt.be | 51.04 | 3.74 | 9050_Ghent-Ledeberg_Belgium | 0.5284928 | TimeOut | TimeOut | TimeOut |
| 10637 | 193.190.168.51 | planetlab2.extern.kuleuven.be | 50.8625 | 4.68599 | ESAT_Kasteelpark_Arenberg_10_bus_2446_Heverlee_Heverlee_3001_Belgium | 0.0339429 | 7898 | 4199 | 603 |
| 10638 | 193.190.168.49 | planetlab1.extern.kuleuven.be | 50.8625 | 4.68599 | ESAT_Kasteelpark_Arenberg_10_bus_2446_Heverlee_Heverlee_3001_Belgium | 0.0339429 | 6026 | 4146 | 631 |
| 10639 | 130.104.72.200 | onelab1.info.ucl.ac.be | 50.6833 | 4.61667 | Place_Sainte-Barbe_2_Louvain-la-Neuve_1348_Belgium | 1.6853749 | TimeOut | 10684 | 5887 |
| 10640 | 130.104.72.201 | onelab2.info.ucl.ac.be | 50.6833 | 4.61667 | Place_Sainte-Barbe_2_Louvain-la-Neuve_1348_Belgium | 1.6853749 | TimeOut | 12096 | 5722 |
| 10641 | 130.104.72.213 | onelab3.info.ucl.ac.be | 50.6833 | 4.61667 | Place_Sainte-Barbe_2_Louvain-la-Neuve_1348_Belgium | 1.6853749 | TimeOut | 3966 | 519 |
| 10642 | 138.48.3.201 | chimay.infonet.fundp.ac.be | 50.467 | 4.867 | 21_rue_Grandgagnage_Namur_B-5000_Belgium | 0.6749787 | TimeOut | TimeOut | TimeOut |
| 10643 | 138.48.3.203 | rochefort.infonet.fundp.ac.be | 50.467 | 4.867 | 21_rue_Grandgagnage_Namur_B-5000_Belgium | 0.6749787 | 14746 | 4233 | 553 |
| 10644 | 138.48.3.202 | orval.infonet.fundp.ac.be | 50.467 | 4.867 | 21_rue_Grandgagnage_Namur_B-5000_Belgium | 0.6749787 | 11972 | 4149 | 567 |

Obr. 1.3: Ukážka časti zoznamu staníc.

2 GOOGLE MAPS API

Spoločnosť Google ponúka vývojárom jednoduchú cestu ako vložiť Google Mapy do webových stránok, a tak využívať rozsiahle množstvo funkcií. Využiť nato môžu Static Maps API, či JavaScript API. Na špecifické úkony je možné využiť mnoho ďalších API ako je napr. Geocoding API, Time Zone API, Elevation API ...

2.1 Static Maps API

Použitie Static Maps API dovoľuje umiestniť mapový obrázok na webovú stránku bez nutnosti využitia akýchkoľvek skriptov. Služba vytvorí mapový obrázok založený na parametroch umiestnených v URL adrese, ktorá je poslaná pomocou štandardného HTTP požiadavku a vráti mapu ako obrázok späť viz Obr. 2.1. Ten môžeme jednoducho umiestniť na webovú stránku. Komplexnú dokumentáciu k Static Maps API je možné nájsť na vývojárskom webe Google [4].

Výpis kódu 2.1: Príklad HTTP požiadavky na vygenerovanie statickej mapy.

```
http://maps.googleapis.com/maps/api/staticmap?center=Brno  
+Technicka+12,Czech+Republic&zoom=16&size=600x300&  
maptype=roadmap&markers=color:blue|label:A  
|49.226803,16.57483&markers=color:green|label:B  
|49.225453,16.57583&markers=color:red|label:C  
|49.225553,16.57383&sensor=false
```



Obr. 2.1: Obrázok vygenerovaný pomocou Static Maps API.

2.2 JavaScript API

JavaScript API v3 je aktuálna verzia dynamického API. Na jeho využitie je potreba aspoň základná znalosť JavaScriptu a princípov objektovo orientovaného programovania. Dokumentácia a popis syntaxe je umiestnená na vývojárskom webe [3].

Príklad využitia JavaScript API:

1. Deklarujeme aplikáciu ako HTML5 použitím `<!DOCTYPE html>`.
2. Vložíme Google Maps API JavaScript.
3. Vytvoríme div element, ktorý pomenujeme „map-canvas“. V tomto elemente bude umiestnená mapa.
4. Vytvoríme JavaScript konštantu, do ktorej umiestníme vlastnosti mapy.
5. Vytvoríme JavaScript objekt „map“ založený na div elemente a vlastnostiach mapy.
6. Použijeme event listener na načítanie mapy po načítaní stránky.

Výpis kódu 2.2: Príklad kódu JavaScript Maps API

```
1 <!DOCTYPE html>
2 <html><head><title>Simple Map</title>
3   <meta name="viewport" content="initial-scale=1.0,
4     user-scalable=no">
5   <meta charset="utf-8">
6   <style>
7     html, body, #map-canvas { height: 100%; margin: 0px;
8       padding: 0px }
9   </style>
10  <script src="https://maps.googleapis.com/maps/api/js?v=3.exp
11    &sensor=false"></script>
12  <script>
13    var map;
14    function initialize() {
15      var mapOptions = {
16        zoom: 8,
17        center: new google.maps.LatLng(49.226803, 16.57483)
18      };
19      map = new google.maps.Map(document.getElementById('map-canvas'
20        ),
21        mapOptions);
22    }
23    google.maps.event.addDomListener(window, 'load', initialize);
24  </script></head>
25  <body>
26    <div id="map-canvas"></div>
27  </body>
28 </html>
```

2.3 Geocoding API

Geokódovanie je proces, pri ktorom sa adresa konvertuje na zemepisné súradnice, ktoré môžeme použiť na umiestnenie ukazovateľa na mape. Spätné geokódovanie je presne opačný proces. Konvertujú sa zemepisné súradnice na ľahšie čitateľnú adresu. Google Geocoding API zabezpečuje priamy prístup k tejto službe cez HTTP požiadavku [5].

2.3.1 Formát požiadavky na Geocoding API

V prípade že užívateľ chce získať od služby správnu odpoveď, musí byť na službu poslaná HTTP požiadavka v správnom tvare:

```
http://maps.googleapis.com/maps/api/geocode/output?parameters
```

Hodnota output môže nadobudnúť jednu z hodnôt:

- json – výstupom bude súbor JavaScript Object Notation (JSON),
- xml – výstupom bude XML súbor.

Položka **parameters** môže nadobudnúť viacero hodnôt, pričom jednotlivé parametre sú oddelené použitím znaku ampersand (&). Niektoré parametre sú vyžadované, iné výberové:

Vyžadované parametre v požiadavke na geokódovanie:

- address – adresa, na ktorú chceme použiť geokódovaciu službu,
- sensor – určuje, či požiadavka prichádza zo zariadenia, ktoré má polohový senzor. Táto hodnota musí byť **true** alebo **false**.

Výberové parametre v požiadavke na geokódovanie:

- bounds – preferované budú miesta umiestnené v zadanom rozmedzí,
- key – API key identifikuje kľúč aplikácie,
- language – jazyk, v ktorom sa majú vrátiť výsledky,
- region – kód krajiny špecifikovaný podľa ccTLD (doména najvyššej úrovne) ¹,
- components – filter komponentov je oddelený rúrou (|).

¹Domény najvyššej úrovne pre krajiny je možné nájsť na <<http://www.wipo.int/amc/en/domains/cctld>>.

2.3.2 Odpoveď Geocoding API

Odpoveď geokódovacej služby sa vráti vo formáte nastavenom v output z HTTP požiadavky.

Odpoveď vo formáte JSON

Ukážka prípadu, kde je v požiadavke nastavená hodnota output na json a address na Andorra:

```
https://maps.googleapis.com/maps/api/geocode/json?address=Andorra&
sensor=false
```

Odpoveď služby Google API:

```
1 {
2   "results" : [
3     {
4       "address_components" : [
5         {
6           "long_name" : "Andorra",
7           "short_name" : "AD",
8           "types" : [ "country", "political" ]
9         }
10      ],
11      "formatted_address" : "Andorra",
12      "geometry" : {
13        "bounds" : {
14          "northeast" : {
15            "lat" : 42.655791,
16            "lng" : 1.786639
17          },
18          "southwest" : {
19            "lat" : 42.4287488,
20            "lng" : 1.4087052
21          }
22        },
23        "location" : {
24          "lat" : 42.5062854,
25          "lng" : 1.5218018
26        },
27        "location_type" : "APPROXIMATE",
28        "viewport" : {
29          "northeast" : {
30            "lat" : 42.655791,
31            "lng" : 1.786639
32          },
33          "southwest" : {
```

```

34         "lat" : 42.4287488,
35         "lng" : 1.4087052
36     }
37 }
38 },
39     "types" : [ "country", "political" ]
40 }
41 ],
42     "status" : "OK"
43 }

```

Odpoveď vo formáte json obsahuje dva koreňové elementy:

- „status“ – obsahuje metadata požiadavky,
- „results“ – obsahuje polia z geokódovanou adresou a zemepisné súradnice.

Status

V odpovedi sa nachádza položka „status“, ktorá obsahuje informácie, ktoré môžu poslúžiť v prípade problémov s geokódovaním.

Položka „status“ môže obsahovať tieto hodnoty:

- „OK“ – žiaden problém sa neobjavil, adresa bola úspešne preložená,
- „ZERO_RESULTS“ – geokódovanie prebehlo bez problému, ale výsledok sa vrátil prázdny. Táto hodnota sa môže vyskytnúť, keď posielaná adresa neexistuje,
- „OVER_QUERY_LIMIT“ – bola presiahnutá denná kvóta požiadaviek,
- „REQUEST_DENIED“ – požiadavka bola zamietnutá. Obvykle sa objaví, keď chýba parameter „senzor“,
- „INVALID_REQUEST“ – môže sa objaviť, keď v požiadavke chýba parameter „address“,
- „UNKNOWN_ERROR“ – požiadavka nemohla byť spracovaná kvôli chybe serveru.

Results

V odpovedi vracia geokodér vo výsledku vždy aj pole „results“. V prípade, že pole neobsahuje žiadne informácie (napr. kvôli neexistujúcej adrese), odpoveď zahŕňa prázdne pole **"results" : []**.

Typický výsledok obsahuje tieto polia:

- types – pole hovorí o type návratovej hodnoty. V príklade sme geokodovali adresu „Andorra“, a návratová hodnota v riadku 39 je „country“, čo znamená, že „Andorra“ je krajina a taktiež „political“ čo hovorí o politickom charaktere,

- `formatted_address` – je reťazec obsahujúci ľuďom dobre čitateľnú adresu,
- `address_components` – je pole obsahujúce adresu rozdelenú na časti. Každá časť adresy je určená tromi hodnotami:
 - `types` – popisuje typ adresovej časti (napr. poštové smerovacie číslo, krajina, kraj, mesto, mestská časť ...),
 - `long_name` – dlhý názov (napr. „`long_name`“ : „Czech Republic“),
 - `short_name` – krátky názov (napr. „`short_name`“ : „CZ“),
- `geometry` – obsahuje nasledujúce informácie:
 - `location` – obsahuje geokódované hodnoty zemepisnej dĺžky a šírky,
 - `location_type` – ukladá špecifické informácie o danej lokalite. Aktuálne podporované hodnoty sú:
 - * „`ROOFTOP`“ informácia o polohe je veľmi presná, až na ulicu a číslo domu,
 - * „`RANGE_INTERPOLATED`“ približná poloha je určená interpolovaním dvoch presných bodov,
 - * „`GEOMETRIC_CENTER`“ výsledná poloha je geometrickým stredom krivky (napr. pri ulici) alebo polygónu (pri väčšom rozsahu),
 - * „`APPROXIMATE`“ informácia o polohe je približná,
 - `viewport` – obsahuje doporučený výrez pre zobrazenie návratových hodnôt, ktorý je špecifikovaný dvoma bodmi: juhozápadným a severovýchodným rohom,
 - `bounds` – obsahuje celkové hranice oblasti,
- `partial_match` – naznačuje, že geokódovací server nevrátil úplne presnú adresu, ktorú sme požadovali, ale bol schopný vyhľadať odpoveď z časti adresy. V tomto prípade je treba skontrolovať, či je zadaná správna adresa.

3 OVERENIE PRESNOSTI POZÍCIE

Na overenie presnosti pozície získanej zo siete PlanetLab som vytvoril program, ktorý som napísal v skriptovacom jazyku Bash [7]. Grafický priebeh programu je zobrazený vo vývojovom diagrame na Obr. 3.1. Program počíta vzdialenosť medzi zemepisnými súradnicami získanými zo siete PlanetLab a reálnou adresou inštitúcie.

Program si po spustení načíta zoznam staníc opísaný v kapitole 1.3 a vyčíta z neho počet serverov, pre ktoré sa bude overovať presnosť pozície. Následne zo zoznamu staníc vyparsuje ku prvému serveru **Latitude** (zemepisnú šírku), **Longitude** (zemepisnú dĺžku) a **Address** (adresu). Program s pomocou Google Geocoding API, ktorý som opísal v kapitole 2.3, zašle požiadavku s adresou na server. Server v odpovedi zašle súbor vo formáte **json**. Skript ho pomocou programu **wget** s prepínačom **-O** uloží ako dočasný súbor z názvom **gps**.

Výpis kódu 3.1: Uloženie json súboru z Google API

```
wget -q -O gps "http://maps.googleapis.com/maps/api/
geocode/json?address=Leibniz_Universitat_Hannover,
_Forschungszentrum_L3S,_Appelstr._9a,_Hannover,-,
_30167,_Germany&sensor=false"
```

V súbore **gps** program vyhledá zemepisné súradnice odpovedajúce zadanej adrese. Tieto súradnice a súradnice získané zo zoznamu staníc následne vloží do JavaScript kódu mapy. Premenné **Latitude** a **Longitude** označujú súradnice získané zo súboru, zatiaľ čo premenné **LatGoogle** a **LonGoogle** označujú súradnice získané zo súboru **gps**.

Výpis kódu 3.2: Súradnice definované v JavaScripte

```
var Coordinates = [
    new google.maps.LatLng($Latitude,$Longitude),
    new google.maps.LatLng($LatGoogle,$LonGoogle)
];
```

Mapa so súradnicami sa uloží do súboru **Map.html**. Skript spustí webový prehliadač Firefox a v ňom otvorí súbor **Map.html**. Pomocou Google Geometry Library [2]. JavaScript vypočíta vzdialenosť medzi dvoma bodmi, ktoré sú určené zemepisnými súradnicami. V našom prípade medzi bodom určeným súradnicami **Latitude** a **Longitude** a druhým bodom určeným súradnicami **LatGoogle** a **LonGoogle**.

Výpis kódu 3.3: Vzdialenosť vypočítaná pomocou Google Geometry Library

```
google.maps.geometry.spherical.computeDistanceBetween(Coordinates
[0], Coordinates[1]);
```

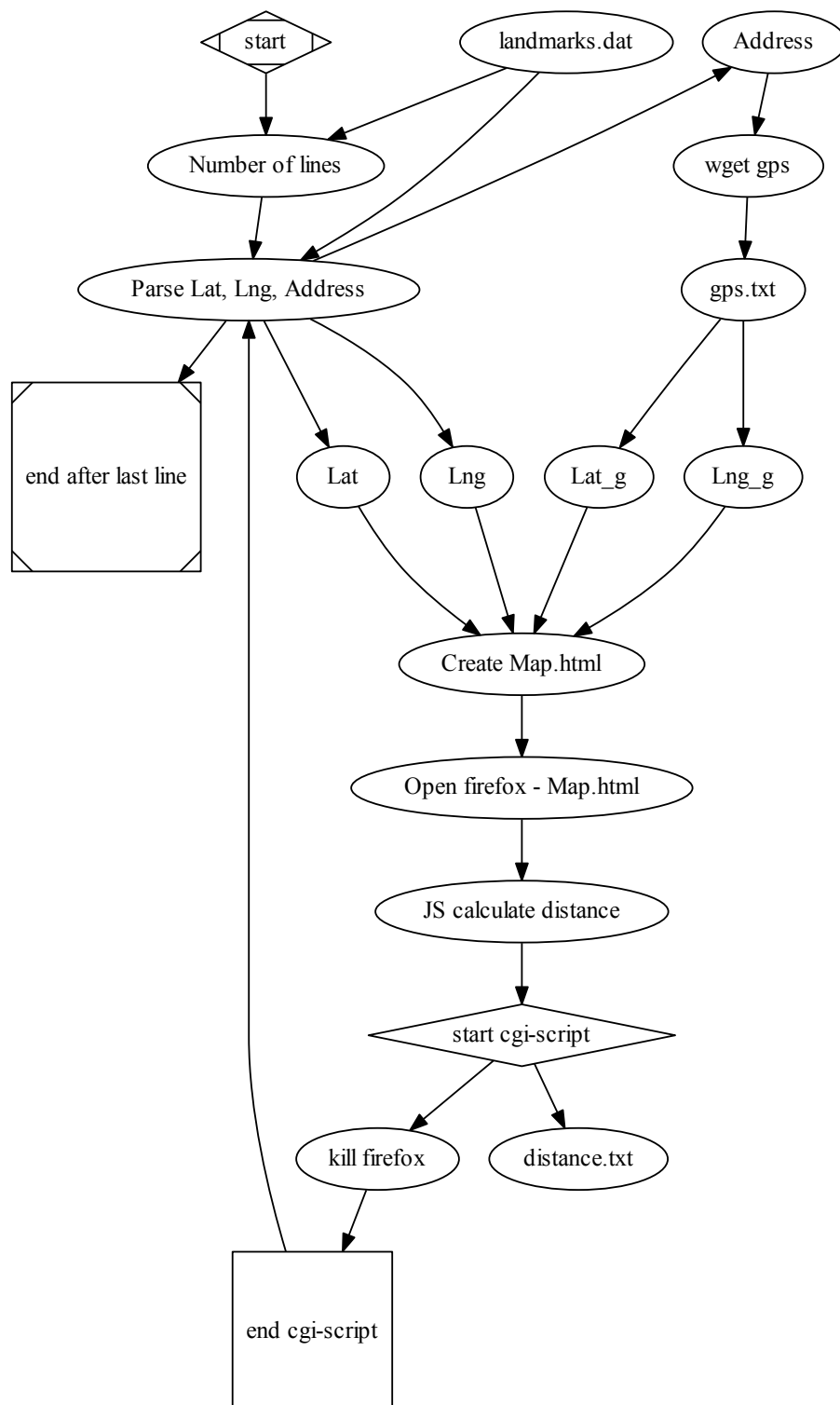
Na overenie presnosti vybranej metódy výpočtu som vypočítal vzdialenosť medzi dvoma bodmi pomocou Haversinovho algoritmu [11] a porovnal som výsledok s výpočtom pomocou Google Geometry Library. Výpočet algoritmu prebieha v JavaScripte. Rozdiel medzi výsledkami výpočtov pomocou Google Geometry Library a Haversinovým algoritmom bol menší ako 10^{-12} km, čo potvrdzuje presnosť výpočtu pomocou Googlu Geometry Library.

Výpis kódu 3.4: Haversinov algoritmus napísaný v jazyku JavaScript

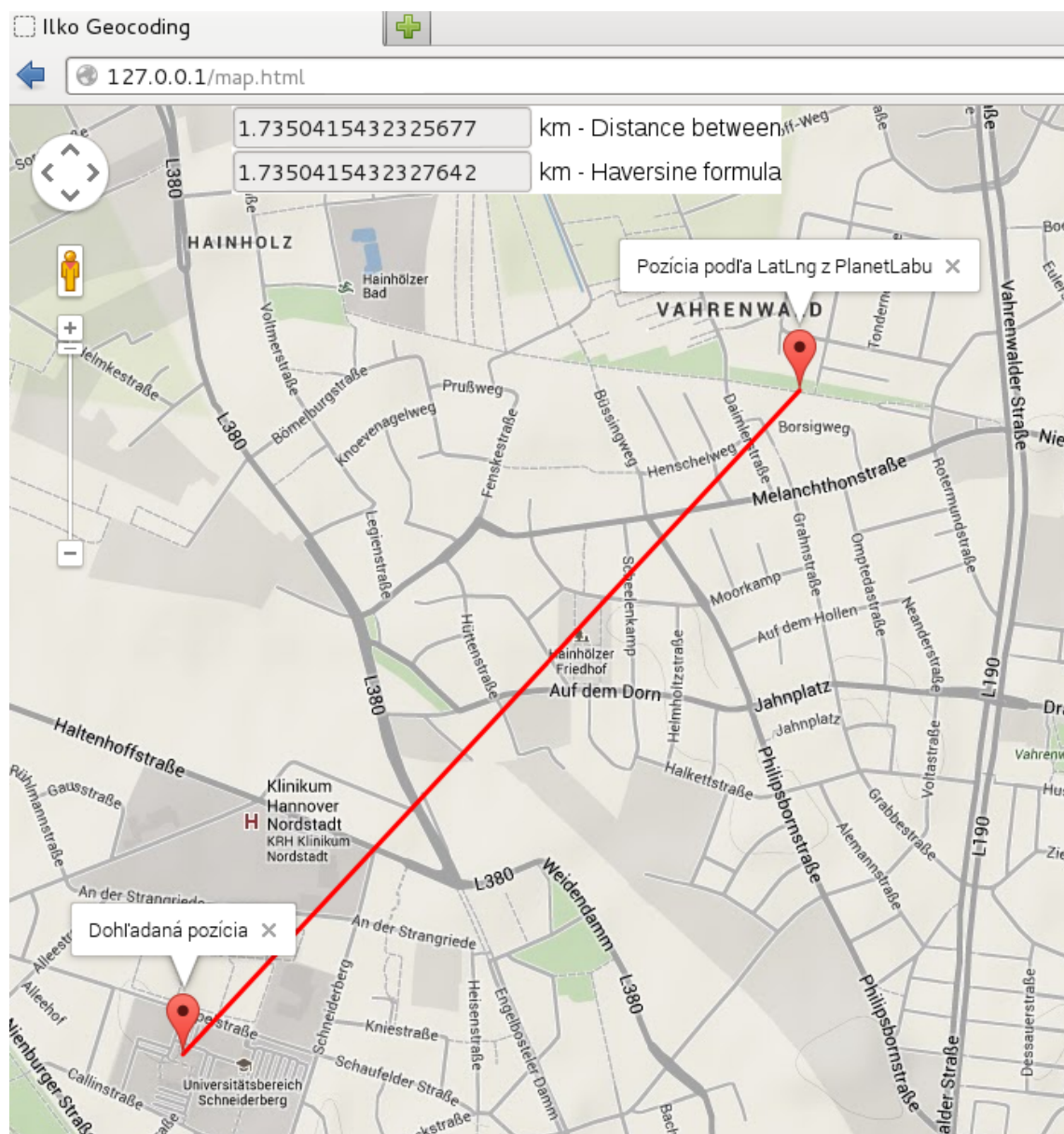
```
1 var R = 6378137; // Polomer zeme v metroch
2 var dLat = deg2rad(co2lat-co1lat);
3 var dLon = deg2rad(co2lng-co1lng);
4 var a =
5     Math.sin(dLat/2) * Math.sin(dLat/2) +
6     Math.cos(deg2rad(co1lat)) * Math.cos(deg2rad(co2lat)) *
7     Math.sin(dLon/2) * Math.sin(dLon/2);
8 var c = 2 * Math.atan2(Math.sqrt(a), Math.sqrt(1-a));
9 var d = R * c; // Vzdialenosť v metroch
10 havers = d/1000; // Do poľa havers pošli vzdialenosť v km
11 document.getElementById('havers').value = (havers);
12 }
13 function deg2rad(deg) {
14     return deg * (Math.PI/180) ;
15 }
```

Po úspešnom výpočte vzdialenosti medzi dvoma bodmi sa spustí cgi skript, ktorý uloží vypočítanú vzdialenosť do súboru `distance.txt` a následne ukončí činnosť webového prehliadača Firefox. Aby mohol cgi skript ukončiť webový prehliadač Firefox, bolo potrebné pridať práva užívateľovi `www-data`, pod ktorým sa cgi skript spúšťa. Kvôli bezpečnosti tento užívateľ tieto práva za normálnych okolností nemá. Po ukončení webového prehliadača sa skript vráti na začiatok a získa údaje o nasledujúcom serveri. Keďže sa webový prehliadač po každom výpočte ukončí, je nutné zablokovať takzvaný `safe mode`, ktorý sa bežne zapína po násilnom vypnutí prehliadača. To je možné nastaviť v pokročilých nastaveniach prehliadača. Namiesto URL adresy sa zadá `about:config`, čím sa sprístupnia tieto nastavenia. Je potrebné zmeniť hodnotu kľúča `toolkit.startup.max_resumed_crashes` na hodnotu `-1`¹. Po spracovaní údajov zo všetkých serverov skript ukončí svoju činnosť.

¹Zdrojový kód Mozilla je prístupný na <<http://mxr.mozilla.org/mozilla-release/source/toolkit/components/startup/nsAppStartup.cpp#975>>.



Obr. 3.1: Vývojový diagram programu na určenie presnosti pozície.



Obr. 3.2: Ukážka výpočtu presnosti pozície.

4 ODOZVA NA SSH

Meraním času nutného na pripojenie sa k serveru pomocou ssh získame údaj, ktorý môžeme použiť na hodnotenie kvality dostupnosti daného serveru. Na meranie času nutného na pripojenie a odpojenie zo servera som vytvoril program v skriptovacom jazyku Bash [7]. Vývojový diagram programu je zobrazený na Obr. 4.1.

Priebeh programu

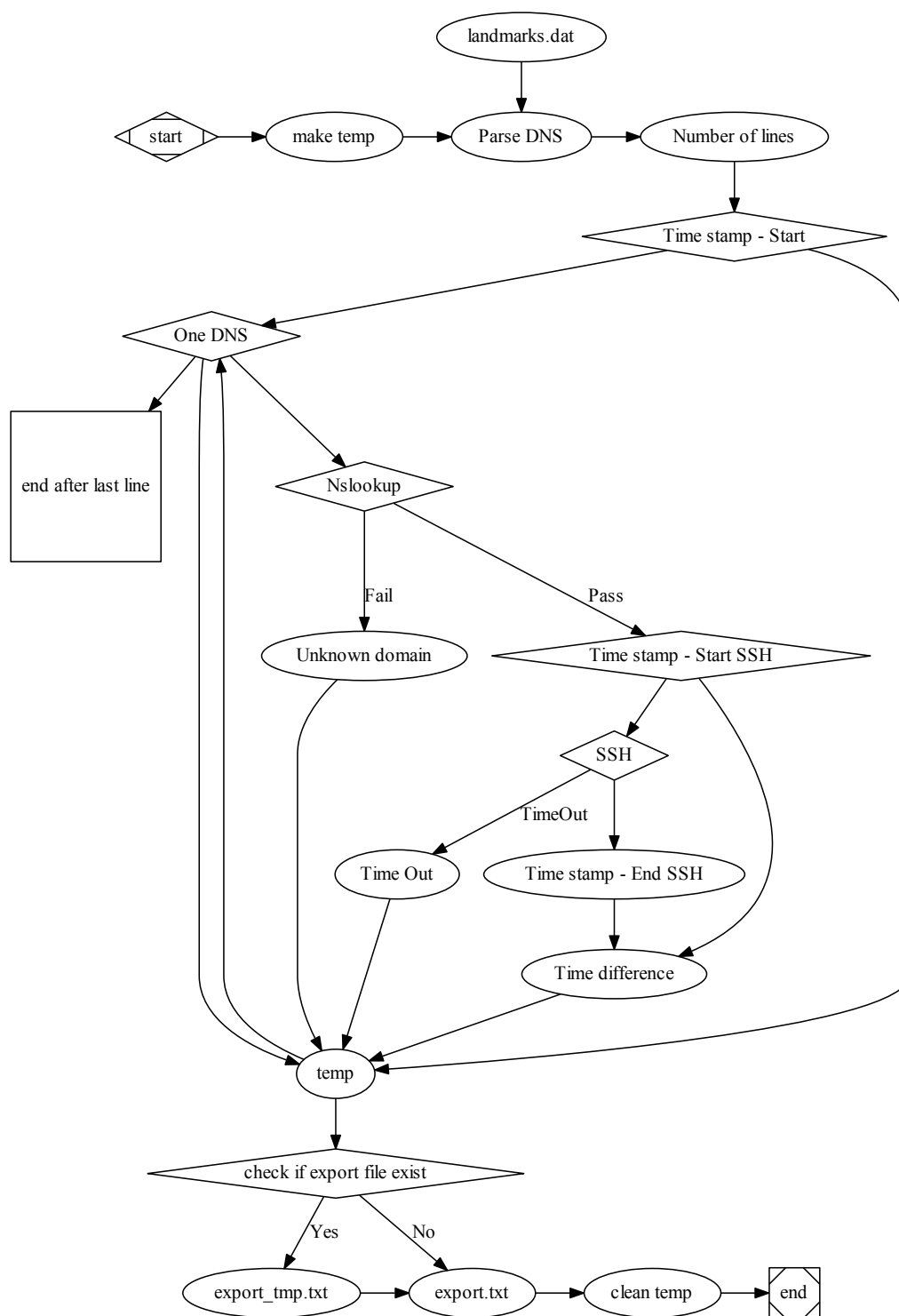
Po spustení programu sa vytvorí dočasný súbor `temp`, ktorý bude program neskôr využívať na ukladanie priebežných hodnôt. Zo súboru `landmarks.dat` program vyberie DNS názvy serverov a spočíta ich celkový počet. Program následne uloží aktuálny čas, teda čas začiatku merania do dočasného súboru `temp`. Program vyberie doménové meno prvého serveru a overí pomocou nástroja `nslookup` [7], či pre dané doménové meno existuje IP adresa. Ak nástroj vyhodnotí toto doménové meno ako neznáme, skript preskočí meranie pre tento server a do dočasného súboru `temp` zapíše doménové meno serveru a k nemu namiesto času zapíše „UnknownDomainName“. Nasleduje meranie podľa ďalšieho doménového mena. V prípade, keď nástroj `nslookup` vráti odpoveď, že doménové meno je možné preložiť na IP adresu, môže začať samotné meranie. Na úvod merania sa uloží aktuálny čas v nanosekundách do premennej `T`, a následne prebehne pripojenie na server pomocou ssh, kde sa vykoná príkaz `exit`, ktorý spojenie ukončí. Po ukončení spojenia sa načíta aktuálny čas a odpočíta sa od neho čas uložený v premennej `T`, čím sa vypočíta čas, ktorý bol potrebný na pripojenie a odpojenie so serveru. Doménové meno spolu s vypočítaným časom sa uloží do dočasného súboru `temp`. Ak by pripojenie nebolo úspešné, sa premenná `timeout` nastaví na 1, čo symbolizuje, že do výstupného súboru sa zapíše ku doménovému menu namiesto času `TimeOut`. Po meraní času nutného na pripojenie na posledný server zo zoznamu program skontroluje, či existuje výstupný súbor `ssh_latency.txt`. V prípade, že súbor neexistuje, vytvorí ho a zapíše do neho výsledky uložené v dočasnom súbore `temp`. V prípade, že súbor `ssh_latency.txt` existuje, vytvorí sa dočasný súbor `ssh_latency_tmp.txt`, do ktorého sa prepíše obsah dočasného súboru `temp`, a následne sa obsah tohto súboru zapíše do nových stĺpcov súboru `ssh_latency.txt`. Každé meranie je zaznamenané v súbore `ssh_latency.txt` v dvoch stĺkoch. V prvom je doménové meno serveru a v druhom zaznamenaný čas. Programový kód je zobrazený v rámci 4.1. Parametre programu ssh som nastavil podľa [10]. Význam jednotlivých argumentov je zobrazený v tabuľke 4.1. Za premennú `$foo` sa nahradí doménový názov serveru.

Výpis kódu 4.1: Meranie času nutného na pripojenie k ssh serveru

```
# time stamp -> ssh connect
T="$(date +%s%N)"
ssh -o PreferredAuthentications=publickey -o
    PasswordAuthentication=no -o ConnectTimeout=30 -o
    StrictHostKeyChecking=no -o UserKnownHostsFile=/dev/
    null -l cesnet_feec -i ~/.ssh/xilkop00_planetlab $foo
    exit || timeout=1
# Nanoseconds
T="$((($(date +%s%N)-T))"
# Milliseconds
M="$((T/1000000))"
```

Tab. 4.1: Argumenty ssh

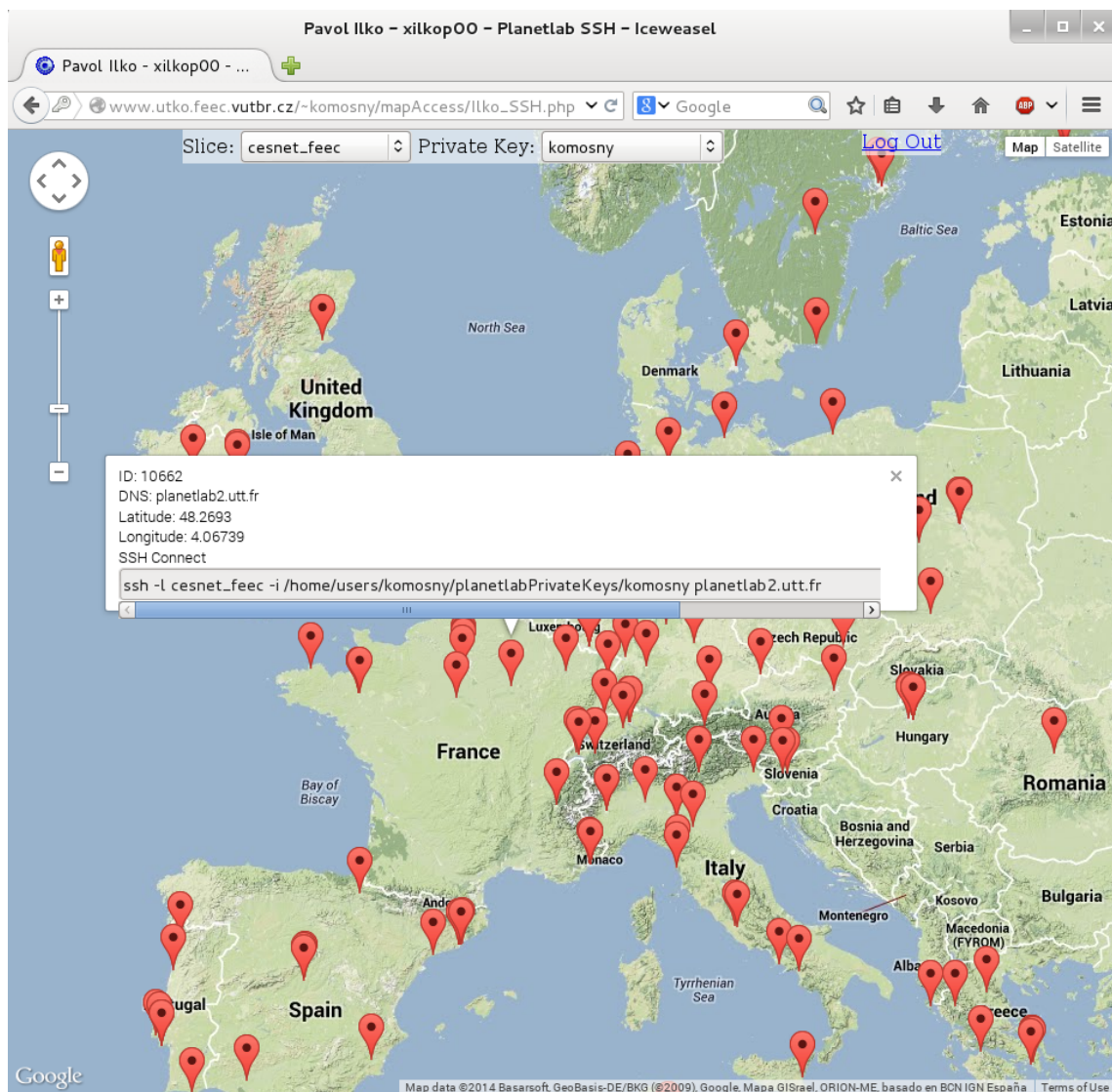
| Použité argumenty SSH | Význam argumentov |
|------------------------------------|--|
| [-i identity_file] | Cesta k súkromnému kľúču |
| [-l login_name] | Prihlasovacie meno |
| [-o option] | Doplnkové nastavenia |
| PreferredAuthentications=publickey | Preferované overenie súkromným kľúčom |
| PasswordAuthentication=no | Zakázané overenie heslom |
| ConnectTimeout=30 | Dostupnosť serveru sa testuje 30 sekúnd |
| StrictHostKeyChecking=no | Automatické prijatie hostiteľského kľúča |
| UserKnownHostsFile=/dev/null | Prijatý kľúč bude uložený do /dev/null |



Obr. 4.1: Vývojový diagram programu na meranie dostupnosti serveru.

5 ZOBRAZENIE POLOHY STANÍC NA MAPE

Kvôli zjednodušeniu pripojenia na jednotlivé stanice som vytvoril mapu z využitím mapových podkladov `maps.google.com`. Na mape sú stanice zobrazené podľa ich zemepisných súradníc zo súboru `landmarks.dat`, teda zo zoznamu staníc. Mapu som vytvoril pomocou JavaScript Maps API. Po kliknutí na stanicu sa zobrazí informačné okno, v ktorom sú uvedené údaje o stanici: `ID`, `DNS`, `Latitude`, `Longitude` a `SSH` príkaz, pomocou ktorého sa je možné pripojiť na vybraný server. Vytvorenú webovú stránku, na ktorej je zobrazená poloha staníc som umiestnil na webový server Ústavu telekomunikácií VUTBR (`teko.utko.feec.vutbr.cz`). Mapa je dostupná na osobných stránkach vedúceho tejto bakalárskej práce doc. Ing. Dana Komosného Ph.D. <<http://www.utko.feec.vutbr.cz/~komosny>> pod odkazom `PlaneLab access`. Na webovom serveri sú umiestnené aj dva súkromné kľúče `komosny` a `xilkop00_planetlab`. Na stránku som preto umiestnil výberové pole `Private Key`. Podľa vybraného kľúča sa do príkazu na pripojenie pomocou `ssh` vygeneruje názov vybraného kľúča. VUTBR má k dispozícii dva projekty (`slice`) zameriavajúce sa na geolokáciu: `cesnet_feec` a `cesnetple_vutbr1` v rámci akademického združenia CESNET [1]. Taktiež som na webovú stránku umiestnil výberové pole, kde je možné medzi týmito projektami prepínať. Po prepnutí projektu sa vo vygenerovanom `ssh` príkaze zmení prihlasovacie meno. Ukážku webovej stránky je možné vidieť na Obr. 5.1.



Obr. 5.1: Zobrazenie polohy staníc na mape.

6 ANALÝZA NAMERANÝCH DÁT

Na základe meraní presnosti pozície a odozvy na ssh som získal údaje, ktoré som spracoval a vložil ich do súboru `landmarks.dat`, na ktorom sme pracovali spoločne s Jakubom Poláškom a Bc. Jánom Pružinským. Tento súbor spolu s kompletnými nameranými dátami je priložený v elektronickej forme na CD.

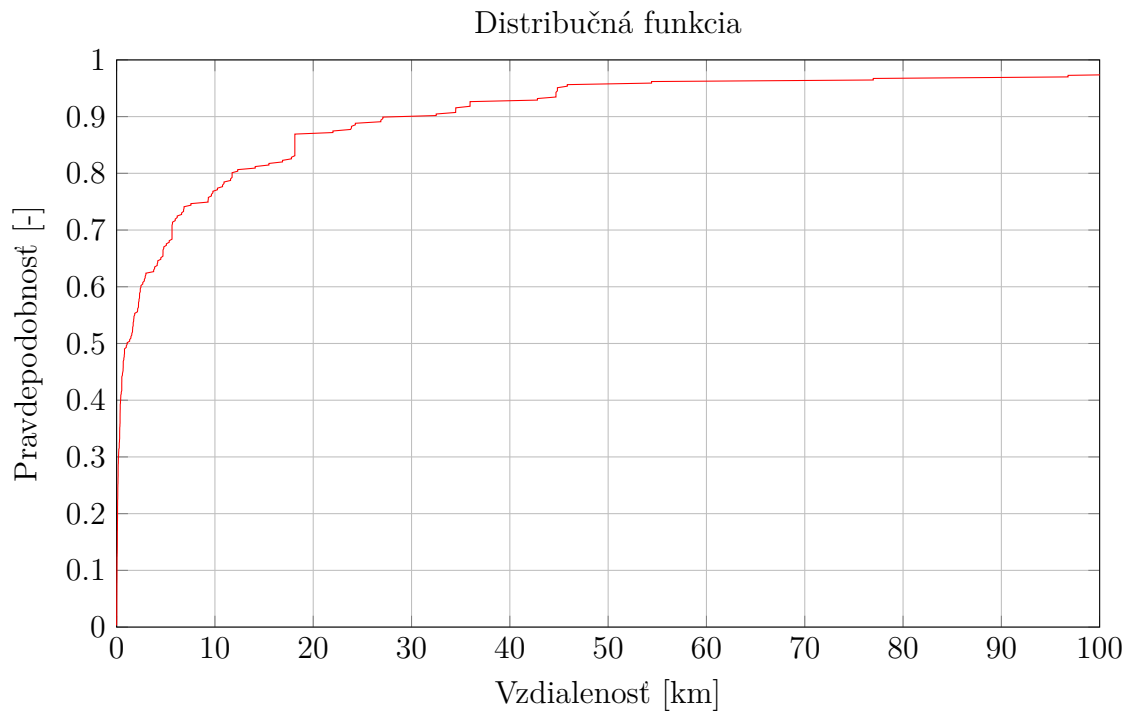
6.1 Overenie presnosti pozície

Výpočet presnosti prebehol pre 367 staníc siete PlanetLab, ktoré sa nachádzajú v Európe. Program počítal vzdialenosť medzi zemepisnými súradnicami získanými zo siete PlanetLab a reálnou adresou inštitúcie. Z vypočítaných dát môžeme odvodiť priemernú vzdialenosť medzi týmito bodmi, ktorá je 17,61 km. Medián týchto hodnôt je 1,07 km. Najmenšia vzdialenosť, teda najpresnejšia pozícia je 0,00088 km (Reykjavik University) a najväčšia vzdialenosť je 716,68 km (Telecom Italia). Najväšia vzdialenosť môže byť zapríčinená tým, že stanice Telecom Italia sa nemusia nachádzať na kontaktnej adrese inštitúcie.

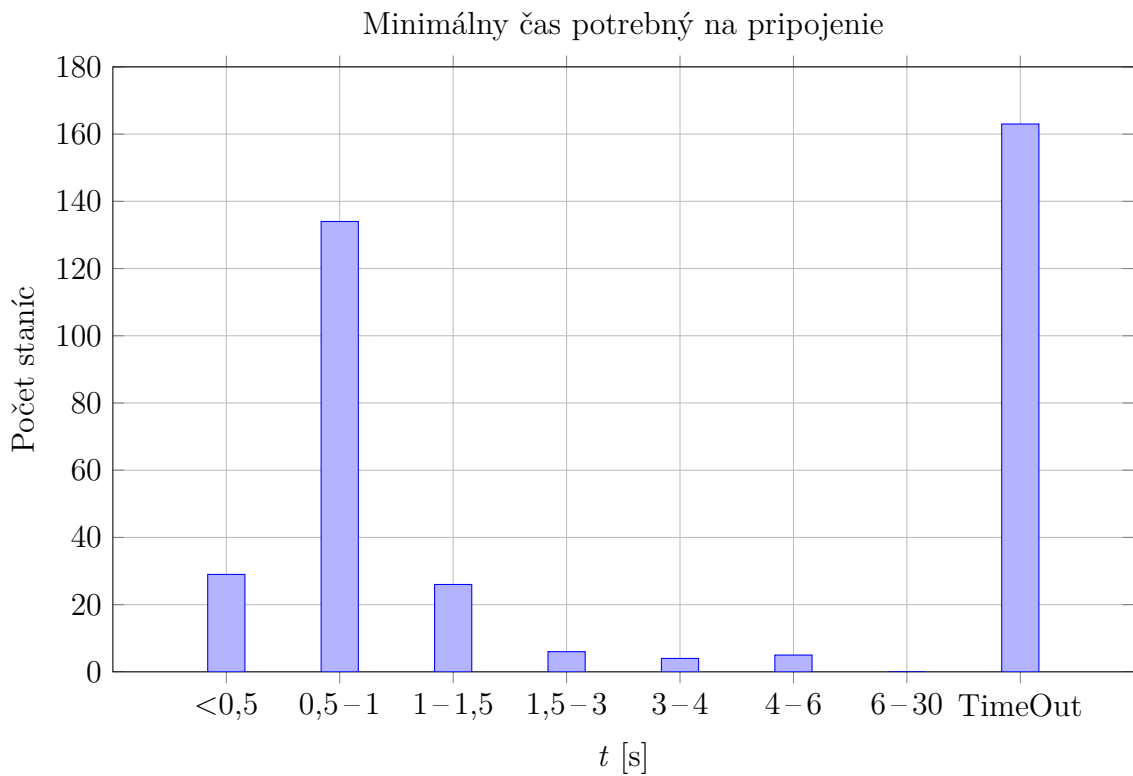
Na zobrazenie presnosti pozície som zostrojil distribučnú funkciu pre vypočítané vzdialenosti, ktorá zobrazuje pravdepodobnosť, že chyba bude menšia, ako daná vzdialenosť. Distribučná funkcia je zobrazená na Obr. 6.1. Z distribučnej funkcie môžeme vyčítať, že až okolo 65 % staníc je vo vzdialenosti do 5 km od adresy ich organizácie [6]. Ja taktiež vidieť, že menej ako 5 % staníc je vo väčšej vzdialenosti ako 50 km.

6.2 Odozva na ssh

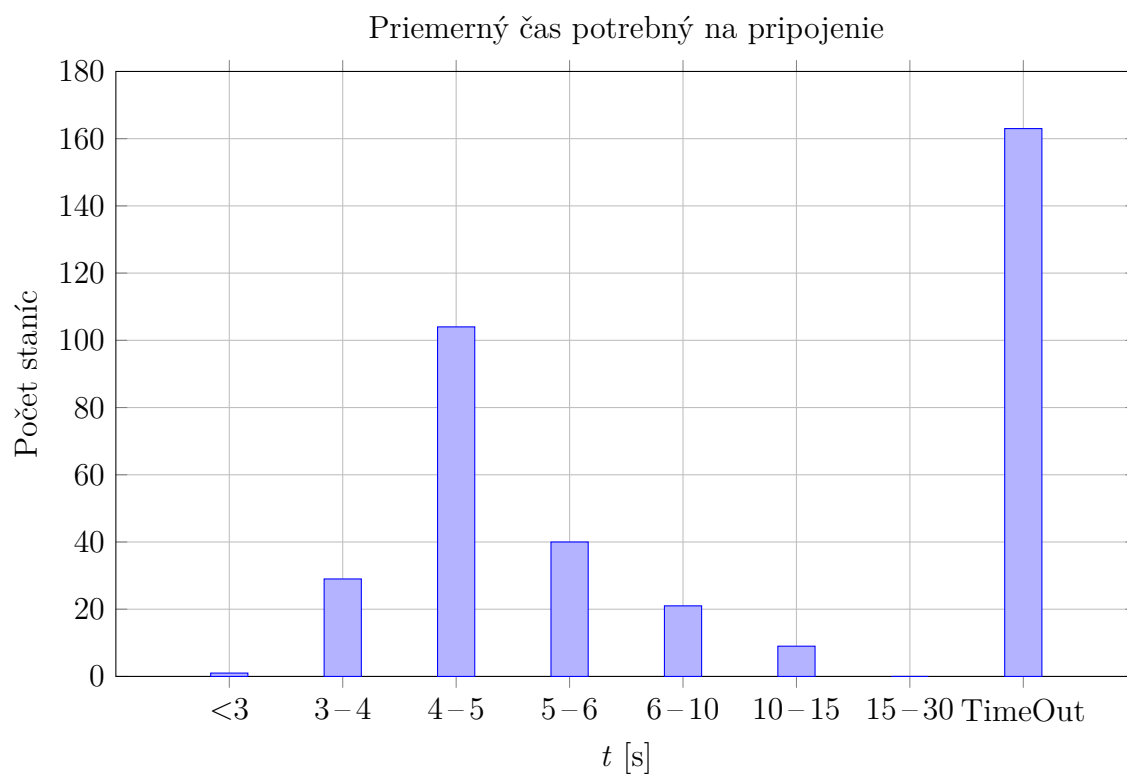
Meranie času nutného na pripojenie a odpojenie so serveru bolo realizované od 12.04.2014 do 05.05.2014. V tomto období prebehlo celkovo 371 meraní na každú stanicu. Jedno celé kolo meraní trvalo v priemere 90 minút. Z nameraných dát som AWK skriptom našiel ku každej stanici záznam s najdlhším časom, najkratším časom, a taktiež vypočítal priemerný čas. AWK skript prikladám v elektronickej podobe na CD. Z najkratších, najdlhších a priemerných časov som zostavil grafy 6.2, 6.3, 6.4, z ktorých je jasne čitateľné, že okolo 160 staníc bolo úplne nedostupných. Z najkratších časov je vidieť, že na väčšinu staníc bolo možné sa pripojiť do 1 sekundy. V priemere to ale vychádza v rozsahu 4 - 5 sekúnd. Pri pohľade na graf Maximálny čas potrebný na pripojenie je vidieť, že počet serverov, kde pripájanie trvalo dlhšie ako 30 sekúnd narástol na 239 staníc. Je ale tiež vidieť, že na 128 staníc sa bolo možné pripojiť pri každom meraní.



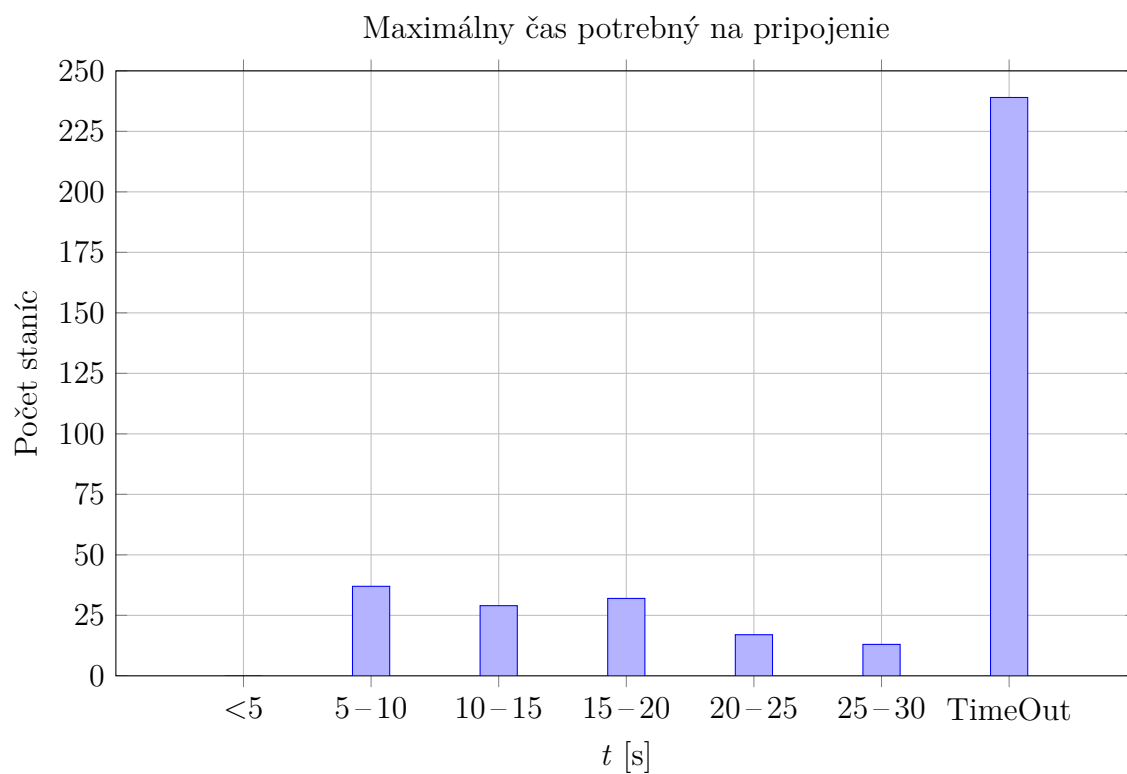
Obr. 6.1: Pravdepodobnosť počtu uzlov proti vzdialenosti medzi súradnicami a adresou organizácie.



Obr. 6.2: Najkratší nameraný čas, za ktorý sa bolo možné pripojiť ku stanici.



Obr. 6.3: Priemerný čas potrebný na pripojenie ku stanici.



Obr. 6.4: Najdlhší nameraný čas, za ktorý sa bolo možné pripojiť ku stanici.

7 ZÁVER

Na začiatku tejto práce som sa zoznámil s experimentálnu sieťou PlanetLab. Opísal som stručný vývoj tejto experimentálnej siete a predstavil pár projektov, ktoré sú v rámci siete realizované. V zadaní práce bolo zostaviť zoznam staníc siete PlanetLab nachádzajúcich sa v Európe. Na konečnom zozname som pracoval spoločne s Jakubom Polášekom a Bc. Jánom Pružinským. Každý do zoznamu pridal približne tretinu staníc a doplnil ho o svoje merania, či výpočty. Výsledný zoznam je priložený v elektronickej podobe na CD pod názvom `landmarks.dat`. Pre každú stanicu sme získali zo siete PlanetLab jej fyzické umiestnenie, teda zemepisné súradnice. Následne sme na webových stránkach organizácií spravujúcich tieto stanice získali ich reálnu adresu. Mojou úlohou bolo zostaviť program, ktorý overí správnosť zistenej polohy pomocou zvolených mapových podkladov. Mojim výberom boli mapové podklady spoločnosti Google, pretože ich vývojové prostredie mi poskytlo množstvo užitočných nástrojov, napríklad nástroj na počítanie vzdialenosti medzi dvoma bodmi. Program na overenie správnosti zistenej polohy som naprogramoval v skriptovacom jazyku `bash`. Program vyčíta zo zoznamu staníc ich zemepisné súradnice a adresu, ktorú následne pomocou Google Geocoding API premení na zemepisné súradnice. Oba body určené ich zemepisnými súradnicami sa vykreslia na mape a s využitím Google Geometric Library sa vypočíta rozdiel medzi týmito bodmi. Rozdiel sa zapíše do súboru `distance.txt`. Program túto operáciu opakuje pre všetky stanice. Priemerná vzdialenosť medzi zemepisnými súradnicami zo siete PlanetLab a adresou inštitúcie bola určená na 17,61 km. Z týchto hodnôt som vypočítal aj medián, ktorého hodnota je 1,07 km.

V rámci práce som vytvoril program na meranie času nutného na pripojenie sa na jednotlivé stanice pomocou ssh. Analýza dát ukázala, že na približne 160 staníc sa nebolo možné pripojiť vôbec, ale zároveň, že na 128 staníc sa bolo možné pripojiť pri každom meraní. V priemere sa čas nutný na pripojenie pohybuje v okolo 4-5 sekúnd.

Na základe tejto práce bola vytvorená publikácia na konferenciu *International Conference on Telecommunications and Signal Processing* ¹.

¹Bližšie informácie o konferencií môžete nájsť na <http://tsp.vutbr.cz>

LITERATÚRA

- [1] Czech Education and Scientific NETwork: Prostředí pro vývoj a testování (PlanetLab). [online], 2013, [cit. 14. 5. 2014]. URL: <<http://www.cesnet.cz/sluzby/planetlab/>>
- [2] Google: Geometry Library. [online], naposledy upravené 11.03.2014 [cit. 8. 5. 2014]. URL: <<https://developers.google.com/maps/documentation/javascript/geometry/>>
- [3] Google: Google Maps JavaScript API v3. [online], naposledy upravené 13.03.2014 [cit. 8. 5. 2014]. URL: <<https://developers.google.com/maps/documentation/javascript/tutorial>>
- [4] Google: Static Maps API V2 Developer Guide. [online], naposledy upravené 11.03.2014 [cit. 8. 5. 2014]. URL: <<https://developers.google.com/maps/documentation/staticmaps/>>
- [5] Google: The Google Geocoding API. [online], naposledy upravené 11.03.2014 [cit. 8. 5. 2014]. URL: <<https://developers.google.com/maps/documentation/geocoding/>>
- [6] KOMOSNY, D., PRUZINSKY, J., ILKO, P., POLASEK, J., OGUZHAN, K.: On Geographic Coordinates of PlanetLab Europe. In *37th International Conference on Telecommunications and Signal Processing (TSP)*, 2014, str. 7.
- [7] Mendel, C.: *Advanced bash scripting guide 5.3: an in-depth exploration of the art of shell scripting*. Lexington: Lulu, 2008, ISBN 978-143-5752-191.
- [8] Peterson, L.; Anderson, T.; Culler, D.; aj.: A Blueprint for Introducing Disruptive Technology into the Internet. In *Proceedings of HotNets-I*, Princeton, New Jersey, October 2002, str. 7.
- [9] PlanetLab: An open platform for developing, deploying, and accessing planetary-scale services. [online], 2007, [cit. 11. 5. 2014]. URL: <<https://www.planet-lab.org/>>
- [10] Provos, N.; Friedl, M.: OpenSSH Manual pages. [online], July 2008, [cit. 3. 4. 2014]. URL: <<http://www.openssh.com/manual.html>>
- [11] Veness, C.: Calculate distance, bearing and more between Latitude/Longitude points. [online], January 2010, [cit. 2. 4. 2014]. URL: <<http://www.movable-type.co.uk/scripts/latlong.html>>

ZOZNAM SYMBOLOV, VELIČÍN A SKRATIEK

API Application Programming Interface

ccTLD country code Top-Level Domain

DOS Denial of Service

DNS Domain Name System

HTML HyperText Markup Language

HTTP Hypertext Transfer Protocol

IP Internet Protocol

JSON JavaScript Object Notation

MIT Massachusetts Institute of Technology

P2P Peer to Peer

PLC The PlanetLab Consortium

SSH Secure Shell

QoS Quality of Service

URL Uniform Resource Locator

XML Extensible Markup Language

ZOZNAM PRÍLOH

A Obsah priloženého CD

37

A OBSAH PRILOŽENÉHO CD

- elektronická verzia práce
- zoznam serverov siete PlanetLab v Európe
- skript na overenie presnosti pozície
- skript na meranie odozvy ssh
- AWK skript vyhľadávajúci minimálnu, maximálnu a priemernú hodnotu
- mapa, zobrazujúca polohu serverov