Středoškolská technika 2017

Setkání a prezentace prací středoškolských studentů na ČVUT

NetAnalyzer

David Podzimek, Jakub Levý

Střední škola a vyšší odborná škola aplikované kybernetiky s.r.o.
Hradecká 1151, 500 03 Hradec Králové

Anotace

Cílem našeho úsilí bylo vytvořit komplexní řešení pro monitorování a analýzu provozu na lokální počítačové síti. Naše zařízení je "plug and go." Je tedy nutné pouze zapojit měřicí zařízení a ostatní se řeší automaticky, spojí se s cloudem a spustí měření. Na webové aplikaci pak najdete informace a statistky o dění ve vaší počítačové síti, seznam a vlastnosti všech zařízení, která se během měření na síti objevila. Samozřejmostí jsou i nástroje pro řešení problémů nebo upozornění na podezřelé chování či chyby.

Klíčová slova: počítačová síť, síťová bezpečnost, analýza dat

Goal of our effort was to create complex solution for monitoring and analysis of traffic on local computer network. Our device is conceived as "plug and go". It is only necessary to connect measuring device and everything else is done automatically. It connects to cloud and starts measurement. In the web application you can find information and statistics about your computer network, list and properties of devices that communicated during measuring of computer network. Tools for solving problems or warnings on computer network for suspicious behavior or errors are matter of course.

Keywords: computer network, networking security, data analysis

Obsah

Anotace 1

Obsah 1

1 Úvod 3
1.1 Situace 3
1.2 Hypotéza 3
1.3 Slovníček 3

2 Počítačová síť 5
2.1 Referenční model ISO/OSI 5
2.2 Internet 6
Obrázek 2: Znázornění internetu 6
2.3 Zařízení síťové infrastruktury 7
2.3.1 Směrovač 7
2.3.2 Přepínač 7
2.4 Vybrané služby ulehčující používání počítačových sítí 8
2.4.1 DHCP 8
2.4.2 DNS 8

3 Hardware 9
3.1 Volba hardwaru 9
3.2 Parametry APU.2C4 10
3.3 Zapojení zařízení 11

4 Webová aplikace 12
4.1 Uživatelské rozhraní 12
4.2 Technologie 13
4.3 Design 14
4.4 Funkce 15
4.5 Databáze 16

5 Měřicí aplikace 17
5.1 Využité technologie 18
5.1.1 C# 18
5.1.1.1 SignalR 18
5.1.2 Python 19
5.1.2.1 Zeromq 19
5.1.2.1 Scapy 19

6 Využití 20

7 Závěr 21

8 Zdroje 22

9 Seznam obrázků 23

1 Úvod

1.1 Situace

V dnešní době si už život bez připojení k internetu nedokážeme téměř ani představit, k internetu je připojeno obrovské množství zařízení a s rozvojem IoT bude jejich počet stále narůstat. Tato zařízení už dávno neslouží pouze ke komunikaci mezi lidmi, stále více zasahují do všech aspektů našeho života. Poskytují nám obrovské množství informací a ulehčují nám život a držet náš svět v chodu a ovládat jej.

Proto nastává čím dál větší potřeba zajistit bezchybný a bezpečný provoz síťové infrastruktury. Aktuálně na trhu není cenově dostupné a komplexní řešení, které by dokázalo v reálném čase vypracovat analýzu sítě s detekcí chyb a podezřelého chování na síti, pomoct s řešením nastalých problémů a vše zobrazilo přehledně v uživatelském rozhraní.

1.2 Hypotéza

Komunikace na síti je rozdělaná na určité vrstvy, tyto vrstvy a jejich funkce jsou detailně popsané v takzvaném ISO/OSI modelu, který by mohl posloužit jako předloha pro navržení databáze, ve které bychom ukládali naměřené informace z počítačové sítě. Tyto informace se budou nadále zpracovávat. Jelikož se jedná o obrovské množství dat, nelze tato data uchovávat v surovém stavu, bude tedy nutné uložit jen takovou část informací, které nám vypoví co nejvíce o stavu sítě či zařízení, které po této počítačové síti komunikuje. Nejvhodnější umístění našeho zařízení, které bude získávat informace o dané počítačové síti, by mělo být mezi směrovačem a hlavním přepínačem, zde by se nám mělo podařit zaznamenat důležité informace k dalšímu zpracování.

1.3 Slovníček

Počítačová síť - propojuje koncové body a ostatní počítačové sítě

Koncový bod - zařízení připojené k počítačové síti (počítač, telefon, tiskárna…)

IP adresa - unikátní adresa koncového bodu v rámci počítačové sítě

Cloud - vzdálené zařízení, které disponuje velkým výpočetním výkonem

Měřicí zařízení - naše zařízení, které má za úkol shromažďovat data z počítačové sítě a odesílat je do cloudu

Framework - softwarová struktura, která slouží jako podpora při programování

Websocket - komunikační protokol, umožňující komunikovat asynchronně přes TCP spojení

IoT - moderní pojem, který se používá k o označení všech koncových bodů k internetu

2 Počítačová síť

2.1 Referenční model ISO/OSI

Obrázek 1: Referenční model ISO/OSI

Fyzická vrstva definuje prostředky pro komunikaci s přenosovým médiem a s technickými prostředky rozhraní. Jedná se jedinou vrstvu, která je zcela hardwarová.

Linková vrstva zajišťuje integritu toku dat z jednoho uzlu počítačové sítě na druhý.

Síťová vrstva definuje protokoly pro směrování dat, jejichž prostřednictvím je zajištěn přenos informací do požadovaného cílového uzlu. V lokální počítačové síti vůbec nemusí být, pokud se nepoužívá směrování.

Transportní vrstva definuje protokoly pro zprávy a zabezpečuje bezchybnost přenosu.

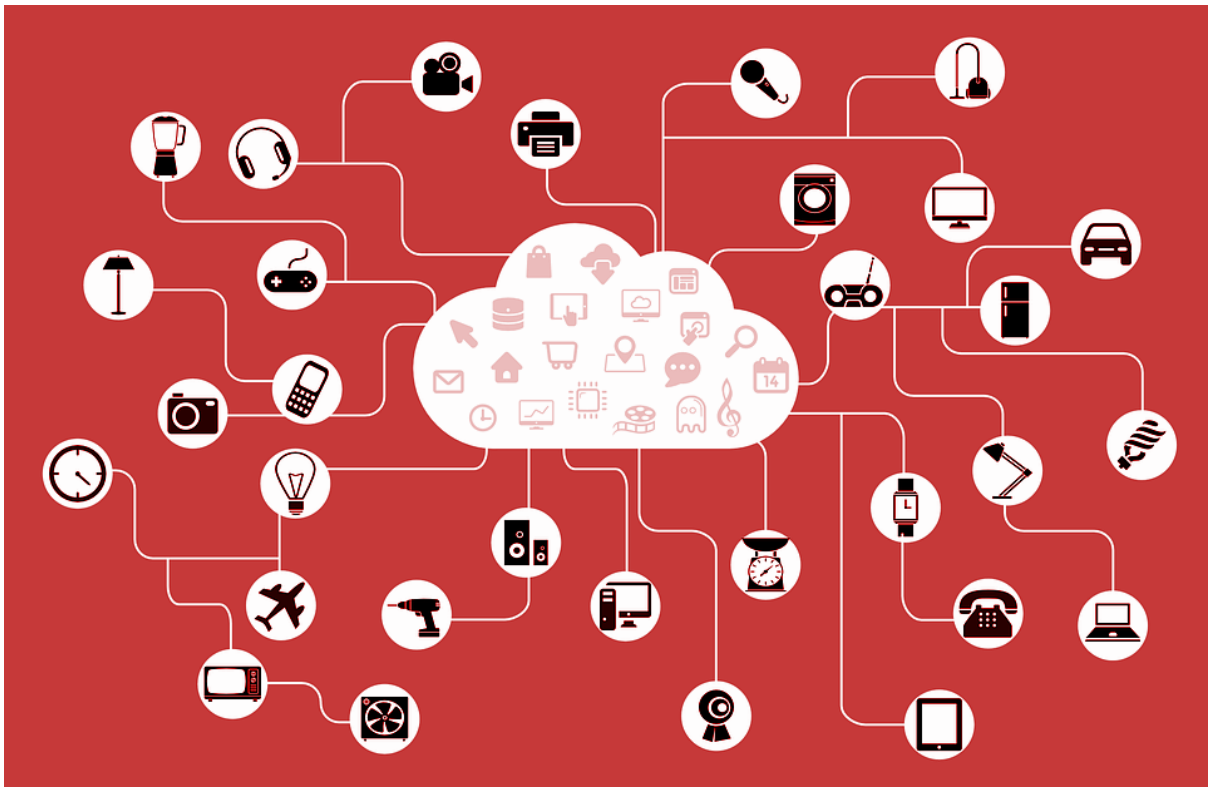
Relační vrstva koordinuje komunikace a udržuje relaci tak dlouho, dokud je potřebná.

Prezentační vrstva udává způsob, jakým jsou data formátována, prezentována a kódována.

Aplikační vrstva je v modelu vrstva nejvyšší. Definuje způsob, jakým komunikují s počítačovou sítí aplikace. Používá služby nižších vrstev a díky tomu je izolována od problémů síťových technických prostředků.

2.2 Internet

Internet je celosvětový systém propojených počítačových sítí, ve kterých mezi sebou počítače komunikují nejčastěji prostřednictvím rodiny protokolů TCP/IP.



Obrázek 2: Znázornění internetu

2.3 Zařízení síťové infrastruktury

2.3.1 Směrovač

Směrovač je síťové zařízení, které procesem zvaným směrování přeposílá datagramy směrem k jejich cíli.

2.3.2 Přepínač

Přepínač je prvek, který připojuje jednotlivé prvky do počítačové sítě. Přepínač obvykle obsahuje desítky síťových portů, na něž se připojují síťová zařízení nebo části sítě. Přepínač přeposílá síťový provoz jenom do těch směrů, do kterých je to potřeba.

2.4 Vybrané služby ulehčující používání počítačových sítí

2.4.1 DHCP

DHCP je služba umožňující automatickou konfiguraci počítačů připojených do počítačové sítě. Tento server přiděluje počítačům pomocí DHCP protokolu zejména IP adresu, masku sítě, výchozí bránu a adresu DNS serveru.

2.4.2 DNS

Jedná se o hierarchický decentralizovaný jmenný systém pro počítače připojené do internetu nebo lokální počítačové sítě. Asociuje mnoho informací s doménovým jménem, zejména zajišťuje překlad doménových jmen na IP adresy a obráceně.

3 Hardware

3.1 Volba hardwaru

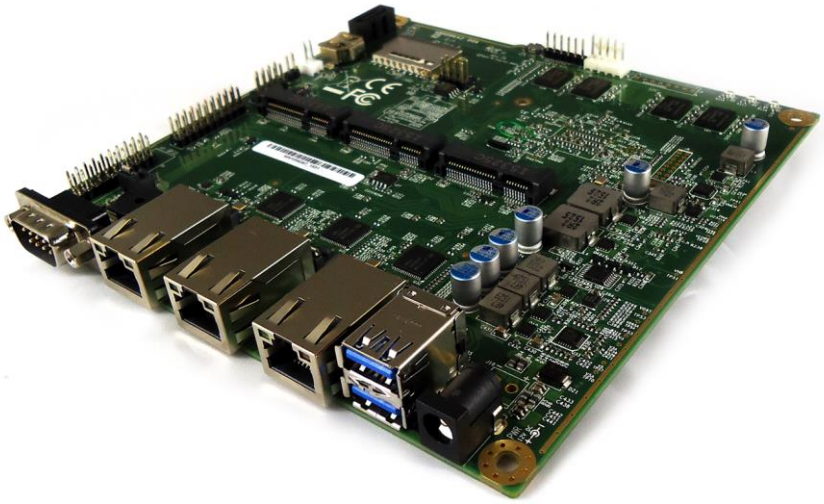
Při volbě hardwaru, který bude umístěný v dané počítačové síti a bude mít za úkol sbírat a odesílat data za pomocí našeho softwaru do cloudu, jsme hleděli především na výkon a periferie. V první řadě jsme chtěli zařízení, které disponuje alespoň dvěma síťovými porty, abychom jej mohli zapojit mezi směrovač sítě a hlavní přepínač. Také bylo nutné, aby zařízení mělo dostatečně velkou operační paměť, ve které budou dočasně uložena data před odesláním.

3.2 Parametry APU.2C4

Procesor: AMD GX-412TC-quad core, 1 GHz

Operační paměť: 4 GB DRAM

Periferie: 3x GLAN
2x miniPCIe
1x mSATA
4x USB
GPIO



Obrázek 3: APU.2C4

3.3 Zapojení zařízení

Instalace samotného měřicího zařízení do počítačové sítě je “plug and go.” Je tedy nutné pouze zapojit zařízení a vše ostatní se už řeší automaticky, měřící zařízení se spojí s cloudem a spustí měření. Nejvhodnější umístění našeho zařízení je mezi směrovač a hlavní přepínač.

4 Webová aplikace

4.1 Uživatelské rozhraní

Webovou aplikaci jsme se snažili za pomocí nejnovějších technologií navrhnout co nejpřehlednější. Uživatelské rozhraní je rozděleno na základních šesti sekcí.

Dashboard - přehled - obsahuje základní informace o provozu na počítačové síti a varování o podezřelém nebo nežádoucí chování

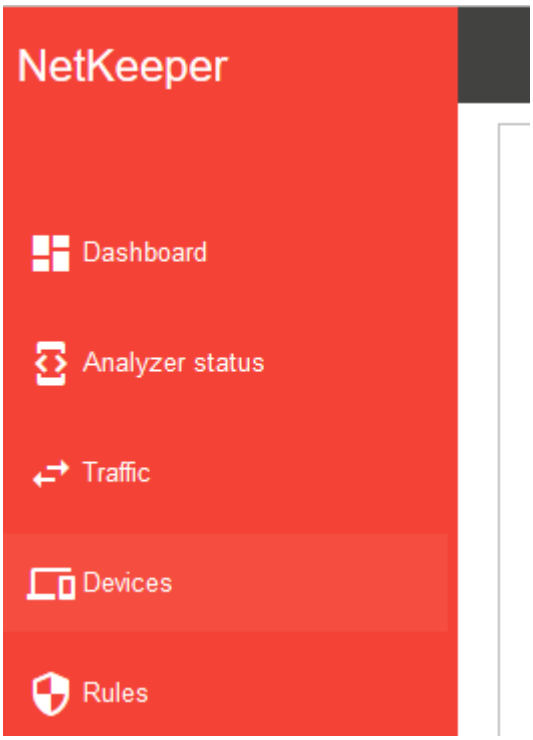
Analyzer status - zde najdeme informace o vytížení měřicího zařízení

Traffic - podrobné statistiky a přehled informující o dění v počítačové síti

Devices - seznam zařízení, která se během měření v síti objevila

Device detail - obsahuje podrobné informace o koncových bodech připojených k počítačové síti a nastroje pro jejich diagnostiku

Rules - seznam pravidel použitých v daném měření



Obrázek 4: Menu

Tyto sekce má uživatel vždy po ruce, nalezne je v levé části obrazovky. Postupem času chceme přidávat další sekce, případně uživatelům umožnit pomocí zpětné vazby požádat o přidání sekcí dle jejich požadavků.

4.2 Technologie

Základem webové aplikace je framework ASP .Net s architekturou MVC. Tato architektura se dělí na tři části:

- **M - Model** - zde jsou uložena data;
- **V - View** - uživatelské rozhraní;
- **C - Controller** - řeší získávání dat z databáze a logiku kódu.

Samozřejmostí je JavaScript, s knihovnou JQuery, který nám pomáhá s vykreslováním grafů a s ostatními dynamickými prvky ve View.

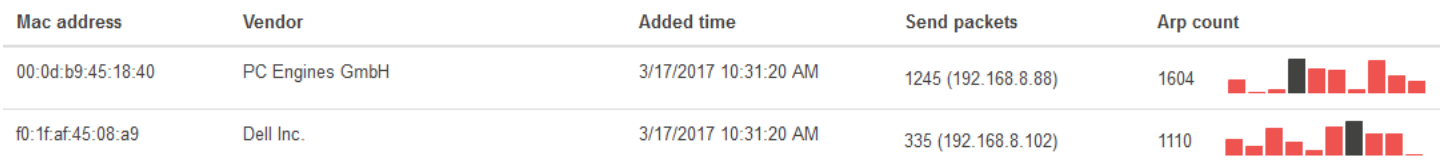
Real time asynchronní komunikaci řeší SignalR, který je detailně popsaný v sekci 5.1.1.1 SignalR.

4.3 Design

Snažili jsme se držet pravidel Material design. Tato pravidla vymyslela společnost Google, která je používá při návrhu mobilní aplikací i webových stránek. Jedná se o pravidla, která definují standardy pro uživatelské rozhraní tak, aby prvky jako ikonky, odsazení ale i chybové hlášky byli podobné napříč aplikacemi a srozumitelné pro uživatele. Dále definují, jak by se měly kombinovat barvy, které k sobě ladí. Více o Material designu najdete [12].

4.4 Funkce

Průběh měření může správce počítačové sítě sledovat v reálném čase na naší webové aplikaci, jež má celou řadu funkcí. Tou hlavní je přehledné znázornění naměřených informací v grafech a ostatních statistikách i upozornění na podezřelé chování některého z koncových bodů, který se v dané počítačové síti nachází. Dále umožňuje okamžitě na takovéto podezřelé chování reagovat. Kromě statistik popisujících jak je využívána síť, zaznamenáváme typ, výrobce a počet koncových bodů v počítačové síti a jejich historii IP adres. Další podrobnosti o ztomto bodu zobrazí funkce "Detail zařízení." Nechybí ani kompletní informace o počítačové síti jako takové, sledujeme např. tyto parametry: virtuální sítě, DNS a DHCP servery a další.

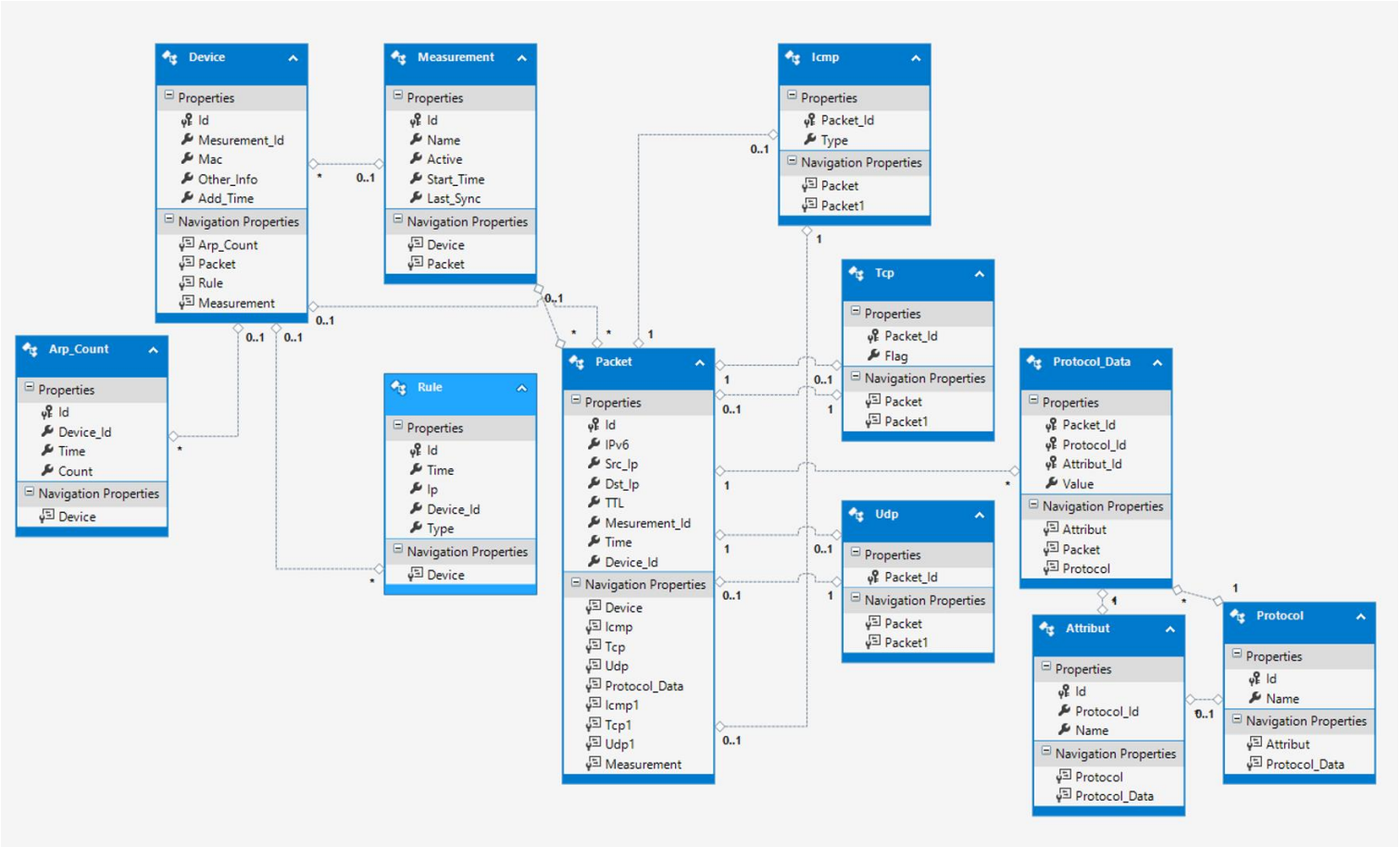


Mac address	Vendor	Added time	Send packets	Arp count
00:0d:b9:45:18:40	PC Engines GmbH	3/17/2017 10:31:20 AM	1245 (192.168.8.88)	1604
f0:1f:af:45:08:a9	Dell Inc.	3/17/2017 10:31:20 AM	335 (192.168.8.102)	1110

Obrázek 5: Sekce Devices

4.5 Databáze

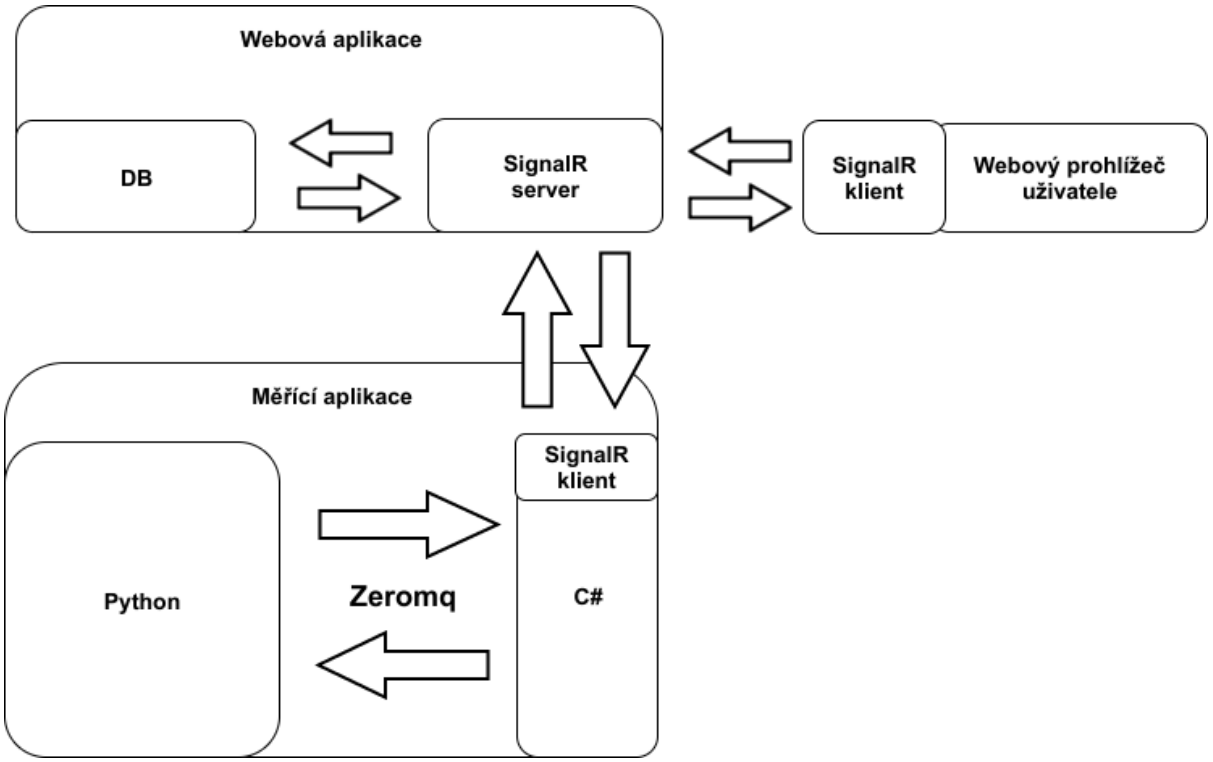
Důležitou součástí našeho řešení je databáze, ve které jsou naměřená data uložena. OSI model, tedy model síťové komunikace, nás inspiroval k vytvoření naší vlastní databázové struktury, do které jsou data ukládána. Začínáme na druhé vrstvě OSI modelu (Linková vrstva) a pokračujeme až k sedmé, té nejvyšší (Aplikační vrstva). Tímto způsobem jsme schopni efektivně uchovávat veškerá data přenášená na počítačové síti nebo zaznamenávat jen takové informace, které jsou v danou chvíli důležité. Dlouho jsme se rozhodovali jaký typ databáze zvolit; zda relační, nebo objektovou. Po dlouhé úvaze nám připadal nejvhodnější relační model, který umožňuje zaznamenávat například vztah mezi zařízením a jeho paketem bez duplikace informací.



Obrázek 6: Model databáze

5 Měřicí aplikace

Klientská část, tedy část běžící na APU.2C4 (viz kapitola 3) je naprogramována pomocí dvou programovacích jazyků - C# a Python. Model funkčnosti klientské aplikace zobrazuje níže uvedený diagram.



Obrázek 7: Diagram funkčnosti aplikace

5.1 Využité technologie

Pro vytvoření našeho softwaru jsme využili několik programovacích jazyků, frameworků a různých dalších technologií zobrazených ve výše uvedeném diagramu.

5.1.1 C#

C# je vysokoúrovňový objektově orientovaný programovací jazyk vyvinutý firmou Microsoft zároveň s platformou .NET Framework. C# lze využít k tvorbě databázových programů, webových aplikací a stránek, webových služeb, mobilních a desktopových aplikací.

5.1.1.1 SignalR

SignalR je knihovna, která může běžet jako součást webové aplikace (serverová část) nebo na druhé straně (u klienta). Mezi její hlavní výhodu patří možnost asychronního volání klienta, konkurenční technologie Ajax toto nepodporuje. SignalR interně ke své funkčnosti využívá websockety. Hlavní nevýhoda této knihovny je implementace Microsoftu pouze pro .NET, to je také důvod, proč jsme se rozhodli pro komunikaci mezi dvěma programovacími jazyky.

5.1.2 Python

Python je vysokoúrovňový skriptovací programovací jazyk, který nabízí dynamickou kontrolu datových typů a podporuje různá programovací paradigmata, včetně objektově orientovaného, imperativního, procedurálního nebo funkcionálního. Veškerá logika měřicího zařízení byla implementována v tomto programovacím jazyku.

5.1.2.1 Zeromq

Zeromq je asynchronní knihovna určená pro komunikaci mezi aplikacemi, cílená pro použití v paralelních aplikacích. Mezi její hlavní výhody patří implementace téměř pro jakýkoliv programovací jazyk. Využíváme ji pro komunikaci mezi Python a C#.

5.1.2.1 Scapy

Scapy je knihovna pro manipulaci s pakety, je napsaná v Pythonu. Vzhledem k její kvalitě jsme se právě rozhodli pro Python jako hlavní programovací jazyk pro měřicí aplikaci.

6 Využití

Naše řešení je aktuálně koncipováno do počítačové sítě malých a středních firem, které si nemohou dovolit investovat statisíce do bezpečnostních a kontrolních prvků ve své infrastruktuře, nebo pro specialisty, kteří se živí návrhem a správou síťové infrastruktury. Snažili jsme se pokrýt co největší oblast problémů, které v současné době na počítačové síti vznikají a vytvořit tak komplexní nástroj, implementovat zajímavé funkce, které konkurenční produkty na trhu neobsahují, ulehčit práci správcům těchto počítačových sítí, zabezpečit chod a bezpečnost dané počítačové sítě.

7 Závěr

Cílem práce bylo vytvořit komplexní řešení, které umožní diagnostiku chyb na počítačové síti, detekci podezřelého chování, monitorování síťového provozu a s ním spojené statistiky vypovídající o využití počítačové sítě. Zároveň jsme se snažili vytvořit nástroj, který pomůže nastalé potíže řešit. Myslíme si, že se nám podařilo se k našemu cíli přiblížit, v řádu měsíců bychom chtěli naše řešení zdokonalit a stát se konkurencí pro obdobné produkty na trhu.

8 Zdroje

[1] [online]. Dostupné z: http://site.the.cz/index.php?id=4

[2] C Sharp – Wikipedie. [online]. Dostupné z: https://cs.wikipedia.org/wiki/C_Sharp

[3] Protokolová datová jednotka – Wikipedie. [online]. Dostupné z: https://cs.wikipedia.org/wiki/Protokolov%C3%A1_datov%C3%A1_jednotka

[4] Domain Name System - Wikipedia. [online]. Dostupné z: https://en.wikipedia.org/wiki/Domain_Name_System

[5] Dynamic Host Configuration Protocol – Wikipedie. [online]. Dostupné z: https://cs.wikipedia.org/wiki/Dynamic_Host_Configuration_Protocol

[6] Internet – Wikipedie. [online]. Dostupné z: https://cs.wikipedia.org/wiki/Internet

[7] Router – Wikipedie. [online]. Dostupné z: https://cs.wikipedia.org/wiki/Router

[8] Switch - Wikipedia. [online]. Dostupné z: https://en.wikipedia.org/wiki/Switch

[9] [online]. Dostupné z: http://signalr.net

[10] ZeroMQ - Wikipedia. [online]. Dostupné z: https://en.wikipedia.org/wiki/ZeroMQ

[11] Scapy - Wikipedia. [online]. Dostupné z: https://en.wikipedia.org/wiki/Scapy

[12] Introduction - Material design - Material design guidelines. Material Design [online]. Copyright © [cit. 26.02.2017]. Dostupné z: https://material.io/guidelines/

[13] Srovnání operátorů - Recenze a hodnocení operátorů [online]. Copyright © [cit. 26.02.2017]. Dostupné z: http://srovnanioperatoru.cz/wp-content/uploads/2016/03/internet-v%C4%9Bc%C3%AD.png

Všechny zdroje jsou aktuální k datu 26. února 2017

9 Seznam obrázků

Obrázek 1: Referenční model ISO/OSI

Obrázek 2: Znázornění internetu

Obrázek 3: APU.2C4

Obrázek 4: Menu

Obrázek 5: Sekce Devices

Obrázek 6: Model databáze

Obrázek 7: Diagram funkčnosti aplikace