

**ŽILINSKÁ UNIVERZITA V ŽILINE**

**FAKULTA ELEKTROTECHNIKY  
A INFORMAČNÝCH TECHNOLOGIÍ**

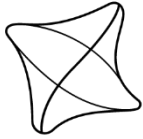
**Katedra multimédií a informačno-komunikačných technológií**

28260120241016

## **MONITOROVACIE SYSTÉMY**

**2024**

**Andrej Pastorák**



ŽILINSKÁ UNIVERZITA V ŽILINE  
Fakulta elektrotechniky  
a informačných technológií

**Katedra multimédií a informačno-komunikačných  
technológií**

## **Monitorovacie systémy**

BAKALÁRSKA PRÁCA

Študijný odbor:	informatika
Študijný program:	komunikačné a informačné technológie
Vedúci bakalárskej práce:	Ing. Slavomír Matúška, PhD.
Pracovisko vedúceho práce:	Žilinská univerzita v Žiline

**Žilina, 2024**

**Andrej Pastorák**



ŽILINSKÁ UNIVERZITA V ŽILINE  
Fakulta elektrotechniky  
a informačných technológií

Katedra multimédií  
a informačno-komunikačných technológií



Akademický rok 2023/2024

## ZADANIE BAKALÁRSKEJ PRÁCE

Meno, priezvisko: **Andrej Pastorák**  
Študijný odbor: **informatika**  
Študijný program: **komunikačné a informačné technológie**  
Téma bakalárskej práce: **Monitorovacie systémy**

Pokyny na vypracovanie bakalárskej práce:

1. Teoreticky spracujte problematiku monitorovania počítačov.
2. Spracujte teoretický prehľad existujúcich systémov určených pre monitorovanie počítačov.
3. Podrobne popíšte voľne dostupný monitorovací systém Zabbix.
4. Nainštalujte a nakonfigurujte monitorovací systém Zabbix. Minimálne požiadavky: monitorovanie aspoň 2 rôznych počítačov s OS Linux a OS Windows, posielanie upozornení pri splnení podmienky (nedostupnosť PC, vysoké zaťaženie PC a pod.), posielanie upozornení cez email a telegram, monitorovanie dostupnosti aspoň dvoch webových lokalít, monitorovanie dostupnosti aspoň 2 RestAPI.
5. Vytvorte podrobný manuál inštalácie a konfigurovania Zabbix-u.

Vedúci bakalárskej práce: Matúška Slavomír, Ing. PhD., Katedra multimédií a informačno-komunikačných technológií, FEIT, Žilinská univerzita v Žiline

Dátum odovzdania bakalárskej práce: 13. 05. 2024

ŽILINSKÁ UNIVERZITA V ŽILINE  
FAKULTA ELEKTROTECHNIKY  
A INFORMAČNÝCH TECHNOLOGIÍ  
KATEDRA MULTIMÉDIÍ A INFORMAČNO-  
KOMUNIKAČNÝCH TECHNOLOGIÍ  
Univerzitná 8215/1, 010 26 Žilina  
-1-

prof. Ing. Róbert Hudec, PhD.  
vedúci katedry

V Žiline dňa 31. 10. 2023

# Anotačný záznam

**Žilinská univerzita v Žiline**

**Fakulta elektrotechniky a informačných technológií**

**Katedra multimédií a informačno-komunikačných technológií**

<b>Typ práce:</b>	Bakalárska práca
<b>Meno a priezvisko:</b>	Andrej Pastorák
<b>Akademický rok:</b>	2023/2024
<b>Názov práce:</b>	Monitorovacie systémy
<b>Počet strán:</b>	37
<b>Počet obrázkov:</b>	23
<b>Počet tabuliek:</b>	0
<b>Počet grafov:</b>	0
<b>Počet príloh:</b>	1
<b>Počet použ. lit.:</b>	9

## **Anotácia v slovenskom jazyku:**

Zabbix je výkonný monitorovací softvér určený na sledovanie stavu a výkonu informačných systémov a sieťových zariadení. Táto práca poskytuje prehľad o jeho architektúre, funkcionalite a možnostiach konfigurácie, s dôrazom na efektívne monitorovanie a správu upozornení. Je užitočným nástrojom pre záujemcov o optimalizáciu IT infraštruktúry.

## **Annotation in foreign language:**

Zabbix is a powerful monitoring software designed to monitor the health and performance of information systems and network devices. This thesis provides an overview of its architecture, functionality, and configuration options, with an emphasis on effective monitoring and alert management. It is a useful tool for those interested in IT infrastructure optimization.

**Kľúčové slová:** Zabbix,  
monitorovacie systémy,  
sieťové zariadenie

**Vedúci bakalárskej práce:** Ing. Slavomír Matúška, PhD.

**Dátum odovzdania ZP:** 10. 05. 2024

## **PodĎakovanie**

Chcem úprimne poďakovať svojmu vedúcemu práce, Ing. Slavomírovi Matuškovi, PhD. za neoceniteľnú podporu, múdre usmernenia a trpezlivé vedenie počas celej mojej cesty vytvárania tejto bakalárskej práce. Odbornosť a široký prehľad mi poskytli potrebnú istotu a inšpiráciu pri prekonávaní všetkých výziev, s ktorými som sa stretol. Bez jeho vedenia by som nedokázal dosiahnuť tento míľnik. Srdečne ďakujem.

# **Abstrakt**

## **Abstrakt v slovenskom jazyku**

Táto práca sa zaoberá monitorovacími systémami, s dôrazom na ich implementáciu pomocou nástroja Zabbix. Monitorovacie systémy sú kľúčovými nástrojmi pre správu informačných technológií a infraštruktúry. Zabbix je voľne dostupný softvér poskytujúci robustnú platformu na sledovanie výkonnosti a dostupnosti rôznych komponentov IT prostredia. Jeho flexibilita a možnosti modulácie umožňujú rôznym organizáciám prispôbiť monitorovanie ich konkrétnym potrebám a prostrediu. V tejto práci sa zameriame na základné princípy Zabbixu, ako je zber a analýza dát, konfigurácia monitorovaných cieľov a upozornení. Tiež sa budeme venovať jeho výhodám, ako je možnosť automatizácie, histórie udalostí a škálovateľnosti. Zabbix je silným nástrojom nielen pre IT profesionálov, ale aj pre organizácie, ktoré potrebujú efektívne monitorovať svoje prostredie a zabezpečiť jeho spoľahlivosť a výkonnosť. Výsledkom práce bude plne funkčný monitorovací systém Zabbix, implementovaný v reálnej sieťovej infraštruktúre.

**Kľúčové slová:** Zabbix, monitorovacie systémy, server

## **Abstract in foreign language**

This thesis covers monitoring systems, with an emphasis on their implementation using the Zabbix software. Monitoring systems are key tools for information technology and infrastructure management. Zabbix is an open-source software providing a robust platform for monitoring performance and availability of various components of an IT environment. Its flexibility and modularization options allow different organizations to tailor monitoring to their specific needs and environments. In this thesis, we focus on the basic principles of Zabbix, such as data collection and analysis, configuration of monitoring targets and notifications. We will also discuss its benefits such as automation, event history and scalability. Zabbix is a powerful tool not only for IT professionals, but also for organizations that need to effectively monitor their environment and ensure its reliability and performance. The work will result in a fully functional Zabbix monitoring system, implemented in a real network infrastructure.

**Keywords:** Zabbix, monitoring systems, server

# Obsah

Anotačný záznam.....	i
PodĎakovanie .....	ii
Abstrakt.....	iii
Abstrakt v slovenskom jazyku .....	iii
Abstract in foreign language .....	iii
Obsah .....	iv
Zoznam obrázkov a tabuliek.....	vi
Zoznam skratiek.....	vii
Úvod.....	1
1    Monitorovanie zariadení .....	3
1.1    Služby monitorovania výkonu .....	3
2    Bežné funkcie monitorovacieho softvéru .....	4
2.1    Využitie CPU (Core Processing Unit – procesor) .....	4
2.2    Využitie disku .....	4
2.3    Monitorovanie a analýza siete .....	4
2.4    Zisťovanie a analýza chybovosti .....	5
2.5    Analýza šírky pásma .....	5
2.6    Prístrojové panely .....	6
2.7    Vzdialený prístup .....	6
2.8    Dostupnosť zariadenia .....	6
2.9    Sledovanie zmien konfigurácie .....	7
3    Prehľad existujúcich softvérov .....	7
3.1    Sematext monitoring.....	7
3.2    Microsoft System Center .....	8
3.3    New Relic .....	9
3.3.1    Monitorovanie aplikácií a služieb .....	9
3.3.2    Monitorovanie webových stránok .....	9
4    Zabbix .....	10
4.1    Úvod do softvéru Zabbix .....	10
4.2    Funkcie Zabbix .....	10
4.3    Prehľad Zabbix .....	12
4.3.1    Server .....	12
4.3.2    Databázové úložisko .....	12

4.3.3	Webové rozhranie .....	12
4.3.4	Proxy server .....	13
4.3.5	Agent.....	13
4.3.6	Tok dát .....	13
5	Implementácia systému Zabbix .....	14
5.1	Počiatkové spracovanie, potrebné nástroje .....	14
5.1.1	Úvaha a návrh potrebných nástrojov .....	14
5.1.2	Virtuálny server .....	16
5.1.3	Sieťové zariadenia .....	20
5.1.4	Upozornenia – e-mail.....	22
5.1.5	Upozornenia – Telegram .....	23
5.2	Webové rozhranie Zabbix.....	25
5.2.1	Prehľad webového rozhranie Zabbix .....	25
5.2.2	Pridávanie sieťových zariadení.....	26
5.2.3	Zozbierané dáta z monitorovacích cieľov.....	27
5.2.4	Monitorovanie webových lokalít .....	29
5.2.5	Monitorovanie REST API .....	31
5.2.6	Trigger – vlastné upozornenia .....	32
5.2.7	Telegram .....	34
5.3	Výsledky práce a diskusia.....	35
	Záver .....	36



## Zoznam obrázkov a tabuliek

Obr. 1: Úvodná stránka HTTP servera .....	16
Obr. 2: Úvodná informačná stránka PHP .....	17
Obr. 3: MariaDB zobrazenie tabuľky užívateľov .....	18
Obr. 4: Zobrazenia služby Zabbix-server .....	18
Obr. 5: Úvodná stránka Zabbix – prvé spustenie.....	19
Obr. 6: Konfiguračný súbor Zabbix agenta - Linux .....	20
Obr. 7: Zobrazenie služby Zabbix agent - Linux.....	20
Obr. 8: Konfiguračné okno Zabbix agenta - Windows.....	21
Obr. 9: Zobrazenie služby Zabbix agent - Windows .....	21
Obr. 10: Automatické e-mailové upozornenia Zabbix .....	22
Obr. 11: Ukážka doručených e-mailov .....	22
Obr. 12: Vytváranie Telegram bota .....	23
Obr. 13: Skupina pre automatické Zabbix upozornenia .....	24
Obr. 14: Úvodná stránka Zabbix webového rozhrania .....	25
Obr. 15: Pridávanie nového zariadenia.....	26
Obr. 16: Ukážka zozbieraných dát zo zariadenia vo forme zápisov.....	27
Obr. 17: Ukážka zozbieraných dát zo zariadenia vo forme grafov .....	28
Obr. 18: Pridávanie novej webovej lokality .....	29
Obr. 19: Parametre a podmienky monitorovania webovej lokality .....	30
Obr. 20: Pridávanie nového REST API.....	31
Obr. 21: Trigger na monitorovanie webovej lokality .....	32
Obr. 22: Trigger na monitorovanie REST API.....	33
Obr. 23: Konfigurácia automatických upozornení cez Telegram.....	34

## Zoznam skratiek

Skratka	Anglický význam	Slovenský význam
<b>IPv4</b>	Internet Protocol version 4	verzia 4 Internet Protokolu
<b>SMTP</b>	Simple Mail Transfer Protocol	Jednoduchý protokol na prenos pošty
<b>RUM</b>	Real User Monitoring	Monitorovanie skutočných užívateľov
<b>IPv6</b>	Internet Protocol version 6	verzia 6 Internet Prokolu
<b>LAN</b>	Local Area Network	Miestna sieť

# Úvod

Témou tejto práce sú monitorovacie systémy. Práca vystihuje všeobecné využitie monitorovacích systémov v praxi, popis a oboznámenie sa s rôznymi dostupnými softvérmi na monitorovanie a predovšetkým monitorovacím systémom Zabbix. Cieľom práce bolo teoretické spracovanie problematiky monitorovania sieťových zariadení. Následne sme teoreticky spracovali rôzne dostupné systémy určené pre monitorovanie počítačov. Poslednou časťou teoretickej časti bolo detailné spracovanie monitorovacieho systému Zabbix, ktorý následne bol aj časťou praktickej časti tejto práce. Hlavnou časťou praktickej časti bolo vytvoriť robustný a spoľahlivý monitorovací systém, ktorý umožní administrátorovi efektívne riadiť a udržiavať sieťové zariadenia a služby v prevádzke. V našom prípade pôjde o monitorovanie a správu štyroch rôznych sieťových zariadení, dvoch webových lokalít a dvoch REST API pomocou softvéru Zabbix. Tento monitorovací systém musí byť schopný poskytnúť administrátorovi detailné informácie o stave jednotlivých prvkov v reálnom čase a umožniť mu reagovať na vzniknuté problémy.

Práca bude zahŕňať nasadenie a konfiguráciu Zabbixu na virtuálnom serveri. Pre jednoduchú správu a organizáciu bude potrebné definovať parametre monitorovania pre každý typ zariadenia a služby.

Dôležitou súčasťou bude nastavenie upozornení pre administrátora v prípade výpadku zariadenia, nízkeho výkonu alebo iných kritických situácií. Tieto upozornenia budú doručené prostredníctvom služieb ako e-mail a Telegram, aby sa zabezpečila rýchla odozva na vzniknuté problémy.

Okrem monitorovania a upozorňovania bude táto práca zahŕňať aj dokumentáciu vytvoreného systému, ktorá bude obsahovať podrobný manuál popisujúci proces inštalácie, konfigurácie a správy Zabbixu.

Voľba témy bakalárskej práce bola nie len z dôvodu osobného rozvoja ale taktiež kariérneho rastu.

Prácu si môžeme rozdeliť na viaceré kapitoly. V kapitole č. 1 sme sa zaoberali všeobecným vysvetlením procesu monitorovania sieťových zariadení. Táto kapitola bola teoretickou časťou našej práce.

Kapitola č. 2 sa vzťahuje na kapitolu č. 1, slúži na lepšie oboznámenia sa s témou monitorovanie sieťových zariadení. Hovorí o bežných funkciách alebo základných metrikách, ktoré sa využívajú pri monitorovaní zariadení ako napr. využitie procesoru, využitie diskov, dostupnosť zariadenia a iné.

V kapitole č. 3 sa venujeme prehľadu existujúcich softvérov na monitorovanie, pričom sa detailnejšie zameriava na populárne nástroje ako Sematext monitoring, Microsoft System Center, New Relic a hlavne Zabbix. Podrobne rozoberáme funkcie a možnosti týchto nástrojov a ich vhodnosť pre rôzne typy prostredí a požiadaviek

Kapitola č. 4 sa zaoberá implementáciou systému Zabbix, počiatočným spracovaním a potrebnými nástrojmi, ako aj detailným popisom implementácie a konfigurácie webového rozhrania Zabbix a využitím v praxi.

V kapitole 5.4 sme zhodnotili dosiahnuté ciele a výsledky práce, ich prínos a problémy s ktorými sme sa stretli.

# 1 Monitorovanie zariadení

V tejto časti sa budeme zaoberať problematikou monitorovania počítačov od základov. Pôjde o spracovanie a rozdelenie komplexného procesu na jednotlivé časti, z ktorých sa môže monitorovanie počítačov skladať.

## 1.1 Služby monitorovania výkonu

Predstavuje softvérový program, ktorý poskytuje monitorovanie stavu systémov a poskytuje služby na zobrazenie stavu výkonu pre jednotlivé koncové zariadenia. Taktiež podporuje monitorovanie prahových, hodnôt pravidelným zhromažďovaním údajov zo systému. Môžu byť použité metódy zhromažďovania údajov o výkone, ktoré poskytuje monitorovanie operačného systému zariadenia alebo virtualizovanej infraštruktúry fyzického počítača, či už s operačným systémom Windows alebo Linux. Taktiež môžu byť použité virtuálne stroje bežiace na serveroch VMware, Hyper-V, Xen Server alebo KVM. Prípadne môže byť spustený akýkoľvek skript na serveri, určený na akumuláciu údajom o výkone. Zozbierané údaje o výkone môžeme efektívne uchovávať na dlhé obdobie. Uchovávanie zozbieraných údajov o výkone počas dlhého obdobia si vyžaduje veľký dátový priestor. Služby monitorovania výkonu agregujú dáta ako „súhrnné dáta“, ktoré sa následne zhromažďujú, čím sa zabráni rýchlemu zaplneniu dátového priestoru. Táto služba taktiež poskytuje grafické zobrazenie pre zozbierané dáta, ktoré následne uľahčuje sledovanie výkonu stavu systému. Problémy s výkonom, ako napríklad vysoké zaťaženie, možno rýchlo odhaliť pomocou grafického zobrazenia a monitorovania stavu systému v reálnom čase. Takisto je možné opätovne zobrazit' dáta o výkonoch, ktoré boli zhromaždené v minulosti. To umožňuje skontrolovať prevádzkový stav systému v minulosti. Funkcia monitorovania prahových hodnôt môže automaticky porovnávať výkon zo zozbieraných dát a ľubovoľnú prahovú hodnotu. V prípade prekročenia prahovej hodnoty môže zaznamenať neobvyklé zaťaženie monitorovaného zariadenia do záznamu alebo o tom informovať systém pomocou služieb tretích strán. Zariadenie, na ktorom beží služba monitorovania výkonu, sa nazýva "server pre správu" a zariadenie, ktorého výkon sa monitoruje, by mal nazývať "monitorovaný stroj". Služba monitorovania výkonu beží na pozadí a spúšťa sa automaticky pri spustení operačného systému [1].

## **2 Bežné funkcie monitorovacieho softvéru**

Niektoré základné metriky na monitorovanie zariadení, ktoré môžu používateľom pomôcť pri monitorovaní stavu ich zariadení, sú tieto:

### **2.1 Využitie CPU (Core Processing Unit – procesor)**

Toto je základná funkcia, ktorú musí byť softvér schopný efektívne vykonávať. Zaťaženie CPU by malo byť možné sledovať prostredníctvom softvéru na monitorovanie zariadení. Keďže procesor je jadrom zariadenia, akákoľvek porucha alebo problémy s jeho výkonom môžu spôsobiť spomalenie a prípadne aj pád celého zariadenia. Okrem toho nadmerné využívanie CPU bude mať za následok nízke využitie pamäte, čo ďalej zhorší stav zariadenia [2].

### **2.2 Využitie disku**

Používateľ môže pomocou tejto funkcie skontrolovať, koľko miesta na disku je k dispozícii na použitie. Ponúka riešenia, ktoré zabránia úplnému vyčerpaniu miesta na disku, a pomáha určiť, ktoré programy alebo procesy spotrebúvajú najviac miesta. Umožňuje používateľovi jednoducho a efektívne rozhodovať o plánovaní kapacity úložného priestoru. Softvér na monitorovanie serverov by mal byť schopný sledovať okrem miesta na disku aj využitie pamäte RAM. Pamäť RAM zaznamenáva len údaje, ktoré sa práve používajú, pretože sa v nej uchovávajú len krátky čas. Z tohto dôvodu môže softvér automaticky monitorovať aktualizácie systému a vyrovnávaciu pamäť a informovať používateľa, kedy má v prípade spomalenia systému vyrovnávaciu pamäť vymazať [2].

### **2.3 Monitorovanie a analýza siete**

Prostredie IT, ktoré je prepojené s rozsiahlou globálnou počítačovou sieťou, podporujú predovšetkým servery. Keďže nie je možné, aby ľudia ručne kontrolovali sieť v každom bode pripojenia, je v tejto situácii nevyhnutný softvér. S cieľom poskytnúť prehľad a analýzu táto funkcia testuje, zhromažďuje, spracováva a vytvára databázu sieťových štatistík. Do tejto oblasti by patrilo aj monitorovanie brány firewall [2].

## 2.4 Zisťovanie a analýza chybovosti

Počet problémov v porovnaní so všetkými požiadavkami na chyby sa nazýva chybovosť. Analýza chybovosti zariadenia dáva používateľovi možnosť včas zistiť potenciálne problémy a vyhnúť sa prípadným výpadkom. Ideálne by bolo zabezpečiť, aby sa vyskytovalo málo alebo žiadne chyby, aj keď prijateľná štandardná chybovosť je menej ako 1% [2].

## 2.5 Analýza šírky pásma

Množstvo prenesených údajov za určitý časový úsek sa nazýva šírka pásma. Pre server, ktorý spracováva údaje smerom dovnútra aj von, je veľmi dôležité vedieť, ktoré aplikácie využívajú najväčšiu šírku pásma. Využívanie väčšej šírky pásma by spôsobilo oneskorenie servera, čo by zhoršilo výkon aplikácie. Používateľ môže zaručiť bezproblémový chod servera minimalizovaním preťaženia a úzkych miest monitorovaním šírky pásma, ktorú využívajú jednotlivé aplikácie. Na monitorovanie šírky pásma sú k dispozícii tri metódy: netflow, SNMP (Simple Network Management Protocol) a packet sniffing. Pokiaľ ide o základné potreby monitorovania šírky pásma, SNMP je najlepšou možnosťou. Serverový monitorovací systém založený na SNMP by zhromažďoval údaje alebo odosielať a prijímať hodnoty prostredníctvom rozhraní sieťových zariadení. Pri rozhodovaní o plánovaní kapacity je SNMP fantastickým nástrojom. Používatelia si môžu vybrať aj rozhranie Windows Management Interface (WMI), protokol spoločnosti Microsoft vytvorený špeciálne pre pracovné stanice, Azure stacky, servery (ako sú servery SQL), virtualizované prostredia a správu siete založenej na technológii Microsoft. Zaujímavou dodatočnou funkciou, ktorú umožňuje analýza šírky pásma, je monitorovanie narušení dát v IT systéme, pri ktorých by hackeri mohli zneužiť sieť spotrebovaním šírky pásma. Správcovia systému môžu v prípade náhleho nárastu alebo zvýšenia využívania šírky pásma rýchlo prijať potrebné opatrenia, ktoré môžu pomôcť pri identifikácii páchatel'ov [2].

## 2.6 Prístrojové panely

V súčasnosti je nevyhnutné, aby riešenie na monitorovanie zariadení obsahovalo prispôsobiteľný ovládací panel alebo šablónu. Pre servery Windows, Linux a Unix existuje viacero šablón pre platformy na monitorovanie serverových aplikácií. Prístrojový panel ponúka robustnú vizualizáciu údajov v rôznych prispôbelených formátoch. Poskytuje používateľovi možnosť vyhodnocovať údaje a ponúkať návrhy a odporúčania na základe údajov. Medzi ďalšie inklúzie patrí kolekcia analytických nástrojov a rozhraní API, ktoré umožňujú bezpečnú integráciu s inými aplikáciami tretích strán. Okrem toho má tento softvér intuitívne webové rozhranie, ktoré umožňuje správcovi prispôbiť a spravovať informačný panel podľa svojich potrieb [2].

## 2.7 Vzdialený prístup

Keďže práca z domu je čoraz bežnejšia, schopnosť softvéru na monitorovanie zariadenia slúžiť ako spojovací článok medzi používateľom a serverom v situáciách, keď používateľ nemôže fyzicky navštíviť serverovňu, sa stala nevyhnutnou. Poskytnutím vzdialeného prístupu k sieťovým zariadeniam by používatelia mohli prevziať kontrolu nad nimi z pohodlia domova a vyriešiť množstvo problémov. Okrem toho má množstvo podnikov tisíce serverov umiestnených vo vzdialených dátových centrách, čo fyzicky znemožňuje manuálnu kontrolu každého jedného servera či zariadenia. Vďaka systémom na monitorovanie zariadení môže príslušný tím monitorovať každé zariadenie z jedného miesta [2].

## 2.8 Dostupnosť zariadenia

Používateľ softvéru na monitorovanie zariadení môže tiež určiť, ktoré zariadenia sú nadmerne alebo nedostatočne využívané. To umožní správcovi pripraviť záložný plán pre prípad, že dôjde k zlyhaniu zariadenia. V ideálnom svete by program na monitorovanie zariadení dokázal presunúť niektoré pracovné úlohy na iné nedostatočne využívané zariadenia tým, že by sledoval zariadenia s malým množstvom voľného miesta na disku, zariadenia vo varovnom alebo kritickom stave, zariadenia s príliš vysokou alebo nízkou teplotou, zariadenia s kriticky nízkou funkčnosťou ventilátorov atď. [2].



## 2.9 Sledovanie zmien konfigurácie

Medzi ďalšie funkcie niektorých systémov monitorovania zariadení patrí sledovanie zmien konfigurácie spôsobených novými doplnkami, odstránenými alebo nahradenými komponentmi, aktualizáciami atď. [2].

## 3 Prehľad existujúcich softvérov

V tejto sekcii budú predstavené 3 existujúce systémy, ktoré sa využívajú na monitorovanie počítačov či už pre komerčné alebo osobné účely.

### 3.1 Sematext monitoring

Sematext je komerčná firma, ktorá vytvára systémy na monitorovanie výkonu aplikácií. Ponúka monitorovanie aplikácií a infraštruktúr s viac ako 100 integráciami, umožňuje zhromažďovať rozsiahly súbor udalostí a tisíce metrík v rámci celej aplikácie. V ponuke s mnohými funkciami ako upozornenia, pravidlá na detekciu anomálií, možnosťou analyzovať metriky pomocou množstva filtrov, centralizované protokolovanie, správu protokolov, analýzu a monitorovanie užívateľov. Integrácie zahŕňajú niekoľko kľúčových funkcií. K dispozícii je monitorovanie serverov, ktoré poskytuje úplný prehľad o využívaní serverov a cloudov. Monitorovanie kontajnerov je možné pomocou Sematext, ktorý je bezproblémovo nasadený so softvérmi ako je Docker alebo Kubernetes. Vďaka monitorovaniu databázy, je možné získať prehľad o stave databázy, či už bežiacej na vlastnej alebo infraštruktúre tretích strán. Transaction Tracing zobrazí pomalé databázové operácie, úplné príkazy SQL, kontext transakcie HTTP koniec-koniec, 10 najlepších operácií podľa priepustnosti, latencie alebo spotrebovaného času, filtrovanie databázových operácií a iné. Monitorovanie inventára sleduje všetky konfigurácie infraštruktúry, zhromažďuje údaje o zariadeniach a zoskupuje ich do množín na ľahší prístup a identifikáciu. Možnosť vytvoriť si vlastné ovládacie panely s údajmi v reálnom čase [3].

## 3.2 Microsoft System Center

Microsoft System Center je balík softvérových produktov navrhnutých na zjednodušenie nasadenia, konfigurácie a správy IT infraštruktúry a virtualizovaných softvérovo definovaných dátových centier na komerčné účely od firmy Microsoft. Produkty System Center sú vhodné pre lokálne aj hybridné cloudové prostredia a podporujú správcov IT pri zabezpečovaní infraštruktúry, monitorovaní infraštruktúry, automatizácii, zálohovaní a správe IT služieb. Najnovšia verzia Microsoft System Center 2022 je softvérový balík na správu dátových centier. Moderné prostredia dátových centier pozostávajú z viacerých komponentov, ako sú napríklad výpočtová technika, siete a úložiská. Často využívajú Windows Server, Azure Stack HCI, nasadenie VMWare a ďalšie. Tieto faktory zvyšujú zložitosť dátových centier a vytvárajú množstvo výziev týkajúcich sa správy. System Center znižuje zložitosť správy infraštruktúry a umožňuje správcovi dátových centier efektívne spravovať celé IT prostredie bez ohľadu na jeho rôznorodosť.

Komponenty služby Microsoft System Center 2022:

System Center Operations Manager (SCOM) monitoruje stav, kapacitu a využitie IT v rámci aplikácií, pracovných záťaží a infraštruktúry.

System Center Orchestrator (SCORCH) umožňuje manažerom IT automatizovať rôzne úlohy dátového centra a vytvárať skripty PowerShell.

System Center Virtual Machine Manager (VMM) poskytuje komplexný nástroj pre siete, úložiská, výpočty a zabezpečenie.

System Center Service Manager (SM) automatizuje a zefektívňuje riešenie incidentov, riadenie zmien a správu.

System Center Data Protection Manager (DPM) poskytuje zálohovanie, obnovu a ochranu údajov pre súkromné cloudy, fyzické počítače, klientov a serverové aplikácie.

Každý z týchto produktov slúži na špecializovaný účel a je k dispozícii samostatne. Najlepšie však fungujú spoločne, aby poskytovali jednotnú správu dátových centier a infraštruktúry [4].

### **3.3 New Relic**

New Relic je softvér s otvoreným zdrojovým kódom na monitorovanie výkonu aplikácií (APM) poskytuje službu monitorovania aplikácií a monitorovanie aj analýzu webových stránok. Taktiež ponúka syntetické monitorovanie v prípade, že je nutné využívať aplikáciu, ktorá musí zostať v prevádzke [5].

#### **3.3.1 Monitorovanie aplikácií a služieb**

Monitorujte všetko od stoviek aplikácií moderného systému až po jednoduchú dobu webových transakcií a priepustnosť aplikácie. Možnosť sledovať stav aplikácie v reálnom čase prostredníctvom monitorovania metrík, udalostí, protokolov a transakcií prostredníctvom vopred pripravených a vlastných ovládacích panelov. APM poskytuje flexibilitu na monitorovanie presne tých vecí, ktoré sú od aplikácie potrebné, využíva pri tom jedného z jazykových agentov. New Relic využíva týchto agentov na komunikáciu s aplikáciou na úrovni kódu. Na výber je z mnoho jazykov ako napríklad: Java, Node.js, PHP, Python, Ruby a iné. Agenti spĺňajú mnoho funkcií, automatické prijímanie protokolov, automapa a externé služby na sledovanie veľkého množstva aplikácií a iné [5].

#### **3.3.2 Monitorovanie webových stránok**

Monitorovanie prehliadača v službe New Relic poskytuje monitorovanie skutočných používateľov (RUM). Meria rýchlosť a výkon, keď koncoví používatelia prechádzajú na webové stránky prostredníctvom rôznych webových prehliadačov, zariadení, operačných systémov a sietí. Umožňuje monitorovať dáta z aktivity prehliadača a optimalizovať výkon v celom systéme. Pomocou monitorovania prehliadača je možné zabezpečiť úspešné nasadenie a rýchlo odstrániť problémy viditeľné zákazníkmi [6].

## 4 Zabbix

V tejto časti bude teoretické spracovanie monitorovacieho softvéru Zabbix.

### 4.1 Úvod do softvéru Zabbix

Zabbix vytvoril Alexej Vladishev a v súčasnosti ho aktívne vyvíja a podporuje spoločnosť Zabbix SIA. Je to softvér s otvoreným zdrojovým kódom pre distribuované monitorovanie pre podniky. Zabbix je softvér, ktorý monitoruje množstvo parametrov siete, stav a integritu serverov, virtuálnych počítačov, aplikácií, služieb, databáz, webových stránok, cloudu a ďalších. Používa flexibilný mechanizmus upozornení, ktorý používateľom umožňuje konfigurovať upozornenia na základe e-mailu prakticky pre akúkoľvek udalosť. To umožňuje rýchlo reagovať na problémy so servermi. Zabbix ponúka vynikajúce funkcie reportovania a vizualizácie dát na základe uložených údajov. Všetky reporty a štatistiky, ako aj konfiguračné parametre sú prístupné cez webové rozhranie. Správne nakonfigurovaný Zabbix môže zohrávať dôležitú úlohu pri monitorovaní IT infraštruktúry. Platí to rovnako pre malé organizácie, ako aj pre veľké spoločnosti [7].

### 4.2 Funkcie Zabbix

Zabbix je vysoko integrovaný softvér na monitorovanie siete, ktoré ponúka množstvo funkcií v jednom balíku [8]:

#### Zhromažďovanie dát

- kontrola dostupnosti a výkonnosti
- podpora SNMP (trapping a polling), IPMI, JMX, VMware
- vlastné kontroly
- zhromažďovanie požadovaných dát vo vlastných intervaloch
- vykonávané serverom/proxy serverom a agentami

#### Flexibilné definície hraničných hodnôt

#### Vysoko konfigurovateľné upozornenia

- odosielanie upozornení možno prispôsobiť pre príjemcu, typ média
- upozornenia môžu byť významné a užitočné pomocou premenných
- automatické činnosti obsahujú vzdialené príkazy

### **Grafické spracovanie v reálnom čase**

- sledované položky sa okamžite zobrazujú pomocou zabudovaných funkcií

### **Možnosti monitorovania webu**

- môže sledovať simuláciu kliknutí myšou na webovej stránke a kontrolovať funkčnosť a čas odozvy

### **Rozsiahle možnosti vizualizácie**

- možnosť vytvárať vlastné grafy, s možnosťou kombinovať viacero položiek do jedného
- reporty
- prezentácie v štýle prístrojovej dosky
- viacúrovňový pohľad pre monitorované zdroje

### **Ukladanie historických údajov**

- údaje uložené v databáze
- konfigurovateľná história
- zabudovaná správa údajov

### **Jednoduchá konfigurácia**

- pridávanie monitorovaných zariadení
- sieťové zariadenia sa vyberú na monitorovanie, keď sú v databáze
- použitie šablón na monitorované zariadenia

### **Používanie šablón**

- šablóny môžu dediť iné šablóny

### **Zisťovanie siete**

- automatické zisťovanie sieťových zariadení
- automatická registrácia agenta
- zisťovanie systémových súborov, sieťových rozhraní a SNMP OID

### **Rýchle webové rozhranie**

- webový frontend v PHP
- prístupné odkiaľkoľvek
- záznamy o auditoch
- možnosť preklikávať sa

## **Zabbix API**

- poskytuje programovateľné rozhranie k Zabbixu na hromadnú manipuláciu, implementáciu softvéru tretích strán a na iné účely

## **Systém oprávnení**

- bezpečné overovanie používateľov
- niektorí používatelia môžu mať obmedzený prístup

## **Kompletný a ľahko rozšíriteľný agent**

- nasadený na monitorovacie ciele
- môže byť nasadený v systémoch Linux aj Windows

## **Binárni daemoni**

- napísané v jazyku C
- ľahko prenosný

## **Pripravené na komplexné prostredia**

- Vzdialené monitorovanie je jednoduché vďaka Zabbix proxy serveru

## **4.3 Prehľad Zabbix**

Zabbix sa skladá z niekoľkých hlavných softvérových komponentov:

### **4.3.1 Server**

Zabbix server je centrálny komponent, ktorému agenti hlásia informácie a štatistiky o dostupnosti a integrite. Server je centrálnym úložiskom, v ktorom sú uložené všetky konfiguračné, štatistické a prevádzkové údaje [9].

### **4.3.2 Databázové úložisko**

Všetky informácie o konfigurácii, ako aj údaje zhromaždené systémom Zabbix sa ukladajú do databázy [9].

### **4.3.3 Webové rozhranie**

Na jednoduchý prístup k Zabbixu odkiaľkoľvek slúži webové rozhranie. Rozhranie je súčasťou servera Zabbix a zvyčajne beží na tom istom fyzickom počítači, na ktorom beží server [9].

#### **4.3.4 Proxy server**

Proxy server Zabbix môže zhromažďovať údaje o výkone a dostupnosti v mene servera Zabbix. Proxy server je voliteľnou súčasťou nasadenia Zabbixu, môže však byť veľmi prospešný na rozloženie záťaže jedného Zabbix servera [9].

#### **4.3.5 Agent**

Agenti Zabbix sú nasadení na monitorovacie ciele, aby aktívne monitorovali miestne zdroje a aplikácie a aby nahlasovali zozbierané údaje na server Zabbix. Sú k dispozícii dva typy agentov: Zabbix agent (jednoduchší, podporovaný na mnohých platformách, napísaný v jazyku C) a Zabbix agent 2 (mimoriadne flexibilný, ľahko rozširiteľný pomocou zásuvných modulov, napísaný v jazyku Go) [9].

#### **4.3.6 Tok dát**

Okrem toho je dôležité urobiť krok späť a pozrieť sa na celkový tok dát v rámci Zabbixu. Aby ste mohli vytvoriť objekt, ktorý zhromažďuje dáta, musíte najskôr vytvoriť hosta. Ak sa presuniete na druhý koniec spektra Zabbix, musíte mať najprv objekt, aby ste mohli vytvoriť trigger. Ak chcete vytvoriť akciu, musíte mať trigger. Ak teda chcete dostať upozornenie, že zaťaženie procesora na serveri X je príliš vysoké, musíte najprv vytvoriť objekt hosta pre server X, po ktorom nasleduje objekt na monitorovanie jeho procesora, potom trigger, ktorý sa aktivuje, ak je CPU príliš vysoké, a po ňom akcia, ktorá vám pošle e-mail. Hoci sa to môže zdať ako veľa krokov, s použitím šablón to tak naozaj nie je. Vďaka tomuto návrhu je však možné vytvoriť veľmi flexibilné nastavenie [9].

## 5 Implementácia systému Zabbix

### 5.1 Počiatočné spracovanie, potrebné nástroje

V tejto časti budeme hovoriť o úvodných požiadavkách a nutných podmienkach na implementáciu systému Zabbix ako plnohodnotného monitorovacieho systému.

#### 5.1.1 Úvaha a návrh potrebných nástrojov

V prvom rade bolo nutné vybrať správnu platformu na realizovanie a iniciovanie monitorovania, v iných slovách potrebovali sme server, na ktorom budú prebiehať všetky nutné procesy pre plnohodnotný beh monitorovacieho systému Zabbix.

Ako prvotný návrh bolo využívanie virtuálneho prostredia Oracle VM VirtualBox, kde sme si vytvorili virtuálne sieťové zariadenie, na ktorom sme následne inštalovali a konfigurovali všetky potrebné prostriedky. Tieto prostriedky boli nutnosťou pre realizáciu a beh monitorovania pomocou systému Zabbix. Virtuálny server pracoval na operačnom systéme Linux, distribúcii Ubuntu 22.04, pričom bol nasadzovaný monitorovací systém Zabbix 6.0.

Následne bola nutnosť monitorovania iných sieťových zariadení pričom na spĺňanie minimálnych požiadaviek sme potrebovali štyri sieťové zariadenia s rôznymi operačnými systémami. Úvodný návrh bol opäť realizovaný prostredníctvom Oracle VM VirtualBox, kde boli vytvorené potrebné virtuálne počítače. Na počítačoch boli inštalované operačné systémy Linux Ubuntu 22.04 a Microsoft Windows 10. Tieto počítače následne slúžili ako cieľ na monitorovanie a testovanie dostupných a požadovaných funkcií Zabbix pomocou Zabbix agentov nainštalovaných a nakonfigurovaných na jednotlivých zariadeniach.

Pri úvahe ohľadom monitorovanie webových lokalít a REST API bolo nutné vyhľadať voľne dostupné API na prvotné testovanie a správanie sa voči monitorovaniu pomocou systému Zabbix. V prípade webovej lokality stačilo použiť voľne dostupnú webovú stránku na účely testovania.



Hlavným významom monitorovania sú upozornenia a ich zasielanie. Pri monitorovaní sieťových zariadení sme teda využívali predpripravené šablóny, ktoré monitorovací systém Zabbix ponúka. Po úvahe sme dospeli k záveru, že hlavné položky na monitorovanie sieťových zariadení budú využitie procesoru, využitie RAM pamäte, zaplnenie a stav diskov, sieťová prevádzka a teploty hardvéru. Pri webových lokalitách a REST API stačilo monitorovať dostupnosť. Všetky tieto položky musia byť monitorované a následne zasielané upozornenia v prípade ich prahovej alebo kritickej hranice.

Na zasielanie upozornení boli využívané služby e-mail a Telegram. V prípade e-mailu bolo nutné vytvoriť SMTP server, pomocou ktorého boli odosielané automatizované maily prostredníctvom monitorovacieho systému Zabbix. V prípade Telegramu bola nutnosť vytvoriť Telegram skupinu, do ktorej následne zasielal Telegram bot automatizované správy a upozornenia z monitorovacieho systému Zabbix.

## 5.1.2 Virtuálny server

V tejto časti si povieme o inštaláciách, konfiguráciách a procesoch, ktoré prebiehajú na virtuálnom serveri. Virtuálny server je dostupný na verejnej IP adrese 158.193.214.162.

Keďže Zabbix používa webové rozhranie nutnosťou bola inštalácia a konfigurácia nástroju Apache2, ktorý slúži ako HTTP server. V rámci konfigurácie boli zmeny v parametroch ako napríklad názov serveru, admin serveru a podobne. Po úspešnej implementácii sme mohli sledovať úvodnú stránku Apache2 HTTP servera, viď. Obr. 1.



Obr. 1: Úvodná stránka HTTP servera

K Apache2 sa vzťahujú aj nástroje PHP 8.1 a PHP-FPM, ktoré bolo treba nainštalovať pre plnohodnotnú funkciu webového rozhrania Zabbix a rýchlosť spracovania PHP skriptov. V konfiguračných súboroch boli zmenené a doplnené hodnoty ako špecifikácia dátumu, času, maximálna možná veľkosť pre nahrávanie súboru a podobné. Po úspešnej inštalácii a konfigurácii sme mohli pozorovať úvodnú informačnú stránku PHP, vid'. Obr. 2.

**PHP Version 8.1.2-1ubuntu2.15**

System	Linux z100 5.15.102-1-pve #1 SMP PVE 5.15.102-1 (2023-03-14T13:48Z) x86_64
Build Date	Feb 23 2024 17:26:53
Build System	Linux
Server API	FPM/FastCGI
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc/php/8.1/fpm
Loaded Configuration File	/etc/php/8.1/fpm/php.ini
Scan this dir for additional .ini files	/etc/php/8.1/fpm/conf.d
Additional .ini files parsed	/etc/php/8.1/fpm/conf.d/10-mysqld.ini, /etc/php/8.1/fpm/conf.d/10-opcache.ini, /etc/php/8.1/fpm/conf.d/10-pdo.ini, /etc/php/8.1/fpm/conf.d/15-xml.ini, /etc/php/8.1/fpm/conf.d/20-bcmath.ini, /etc/php/8.1/fpm/conf.d/20-calendar.ini, /etc/php/8.1/fpm/conf.d/20-ctype.ini, /etc/php/8.1/fpm/conf.d/20-dom.ini, /etc/php/8.1/fpm/conf.d/20-exif.ini, /etc/php/8.1/fpm/conf.d/20-ffi.ini, /etc/php/8.1/fpm/conf.d/20-fileinfo.ini, /etc/php/8.1/fpm/conf.d/20-ftp.ini, /etc/php/8.1/fpm/conf.d/20-gd.ini, /etc/php/8.1/fpm/conf.d/20-gettext.ini, /etc/php/8.1/fpm/conf.d/20-iconv.ini, /etc/php/8.1/fpm/conf.d/20-ldap.ini, /etc/php/8.1/fpm/conf.d/20-mbstring.ini, /etc/php/8.1/fpm/conf.d/20-mysqli.ini, /etc/php/8.1/fpm/conf.d/20-pdo_mysql.ini, /etc/php/8.1/fpm/conf.d/20-phar.ini, /etc/php/8.1/fpm/conf.d/20-posix.ini, /etc/php/8.1/fpm/conf.d/20-readline.ini, /etc/php/8.1/fpm/conf.d/20-shmop.ini, /etc/php/8.1/fpm/conf.d/20-simplexml.ini, /etc/php/8.1/fpm/conf.d/20-sockets.ini, /etc/php/8.1/fpm/conf.d/20-sysvmsg.ini, /etc/php/8.1/fpm/conf.d/20-sysvsem.ini, /etc/php/8.1/fpm/conf.d/20-sysvshm.ini, /etc/php/8.1/fpm/conf.d/20-tokenizer.ini, /etc/php/8.1/fpm/conf.d/20-xmlreader.ini, /etc/php/8.1/fpm/conf.d/20-xmlwriter.ini, /etc/php/8.1/fpm/conf.d/20-xsl.ini
PHP API	20210902
PHP Extension	20210902
Zend Extension	420210902
Zend Extension Build	API420210902.NTS
PHP Extension Build	API20210902.NTS
Debug Build	no
Thread Safety	disabled
Zend Signal Handling	enabled
Zend Memory Manager	enabled
Zend Multibyte Support	provided by mbstring
IPv6 Support	enabled
DTrace Support	available, disabled
Registered PHP Streams	https, ftps, compress.zlib, php, file, glob, data, http, ftp, phar
Registered Stream Socket Transports	tcp, udp, unix, udg, ssl, tls, tlsv1.0, tlsv1.1, tlsv1.2, tlsv1.3
Registered Stream Filters	zlib.*, string.rot13, string.toupper, string.tolower, convert.*, consumed, dechunk, convert.iconv.*

This program makes use of the Zend Scripting Language Engine:  
 Zend Engine v4.1.2, Copyright (c) Zend Technologies  
 with Zend OPcache v8.1.2-1ubuntu2.15, Copyright (c), by Zend Technologies

zend engine

Obr. 2: Úvodná informačná stránka PHP

Ako ďalšia nutná časť bola databáza, pomocou ktorej bude Zabbix ukladať svoje dáta. Zvolili sme nástroj MariaDB server, ktorý slúži ako databázový server. Po inštalácii a overení parametrov konfiguračných súborov sme mohli vytvoriť databázu pre Zabbix s príslušným užívateľským menom, heslom a samozrejme administrátorskými právami atď., vid'. Obr. 3.

```
MariaDB [zabbix]> select userid, username, name, surname, autologin, autologout, lang, refresh, theme from users;
```

userid	username	name	surname	autologin	autologout	lang	refresh	theme
1	Admin	Zabbix	Administrator	1	0	default	30s	default
2	guest			0	15m	default	30s	default

```
2 rows in set (0.000 sec)
```

*Obr. 3: MariaDB zobrazenie tabuľky užívateľov*

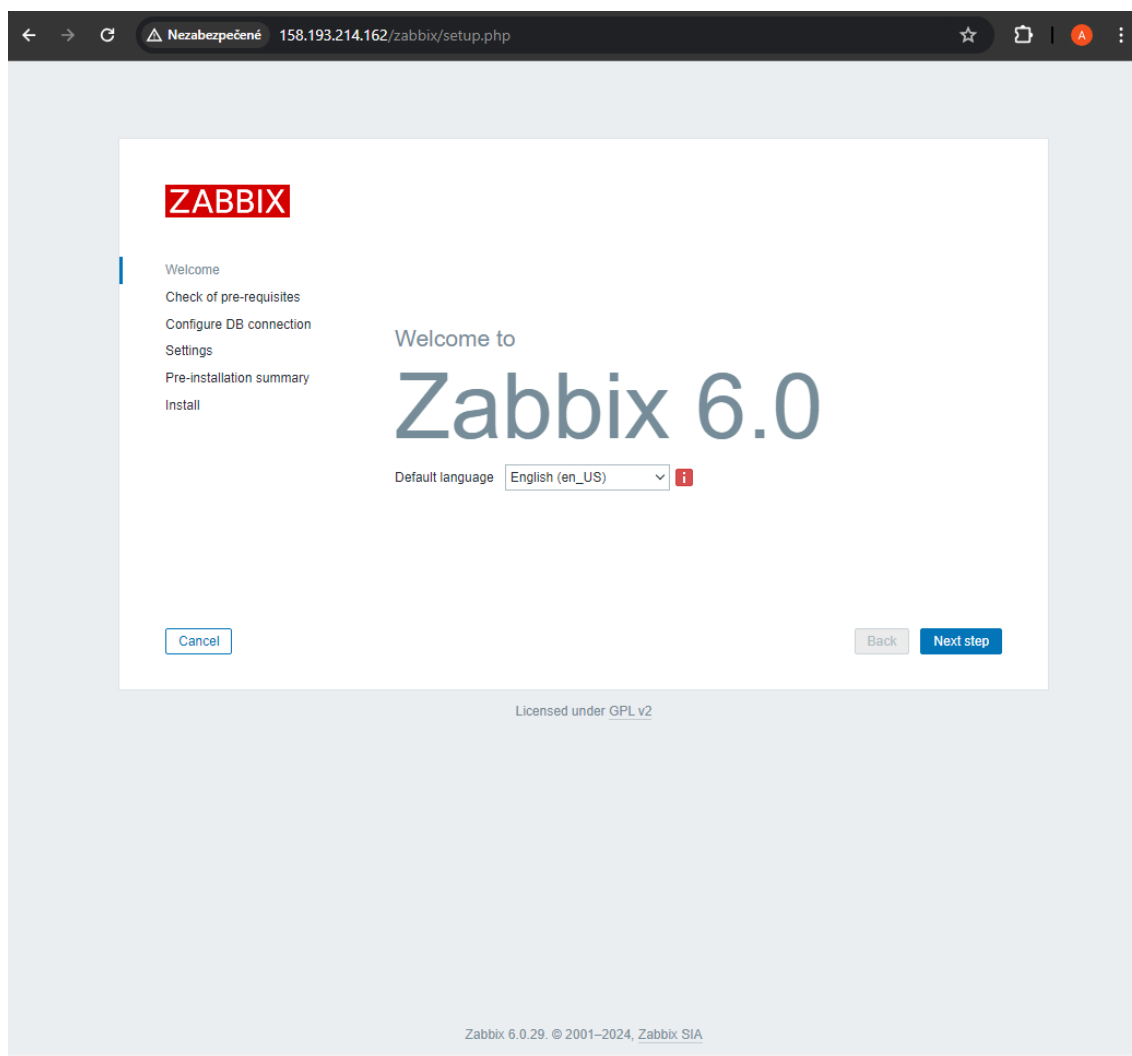
Posledný krok bol samotná inštalácia a konfigurácia Zabbix servera. Inštalácia prebehla pomocou Zabbix repozitára, ktorá obsahovala všetky potrebné balíčky a nástroje na inštaláciu. Po inštalácii bolo potrebné prepojiť Zabbix server s konkrétnou Zabbix databázou a administrátorom databázy pre prístup do databázy, načítanie skriptov a podobne. V Zabbix server konfiguračnom súbore sme teda špecifikovali názov databázy a prihlasovacie údaje. Následne sme povolili službu Zabbix server. V tomto stave boli všetky nutné podmienky pre funkciu monitorovacieho systému Zabbix splnené, vid'. Obr. 4.

```
root@z100:~# systemctl status zabbix-server
● zabbix-server.service - Zabbix Server
   Loaded: loaded (/lib/systemd/system/zabbix-server.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2024-04-26 13:28:41 UTC; 1 week 2 days ago
     Process: 7935 ExecStart=/usr/sbin/zabbix_server -c $CONFFILE (code=exited, status=0/SUCCESS)
    Main PID: 7942 (zabbix_server)
      Tasks: 48 (limit: 33547)
     Memory: 68.9M
        CPU: 42min 48.809s
```

*Obr. 4: Zobrazenia služby Zabbix-server*

Z dôvodu centralizácie všetkých služieb na jeden virtuálny server sme zaviedli aj monitorovanie tohto servera pomocou Zabbix agenta. Inštalácia agenta bola súčasťou balíčkov inštalovaných pri Zabbix serveri, takže bola potrebná iba konfigurácia samotných konfiguračných súborov Zabbix agenta. Špecifikovali sme IP adresu servera, hostname a iné.

Po úspešnej implementácii sme mohli prejsť na úvodnú stránku Zabbix webového rozhrania, kde sa nastavovali finálne kroky, vid'. Obr. 5. Úspešným dokončením finálnych krokov sme boli presmerovaní na prihlasovaciu stránku monitorovacieho systému Zabbix, kde sme sa prihlásili ako administrátor.



*Obr. 5: Úvodná stránka Zabbix – prvé spustenie*

### 5.1.3 Sieťové zariadenia

Konfigurácia jednotlivých sieťových zariadení bola rozdelená na zariadenia s operačným systémom Linux a Windows. V prípade Linuxového zariadenia išlo o jednoduchú inštaláciu Zabbix agenta, ktorému následne boli špecifikované parametre ako napr. IP adresa servera v konfiguračnom súbore, viď. Obr. 6. Úspešne implementovaného Zabbix agenta môžeme vidieť na obrázku Obr. 7.

```
Server=158.193.214.162

### Option: ListenPort
#   Agent will listen on this port for connections from the server.
#
# Mandatory: no
# Range: 1024-32767
# Default:
# ListenPort=10050

### Option: ListenIP
#   List of comma delimited IP addresses that the agent should listen on.
#   First IP address is sent to Zabbix server if connecting to it to retrieve list of active checks.
#
# Mandatory: no
# Default:
# ListenIP=0.0.0.0

### Option: StatusPort
#   Agent will listen on this port for HTTP status requests.
#
# Mandatory: no
# Range: 1024-32767
# Default:
# StatusPort=

#### Active checks related

### Option: ServerActive
#   Zabbix server/proxy address or cluster configuration to get active checks from.
#   Server/proxy address is IP address or DNS name and optional port separated by colon.
#   Cluster configuration is one or more server addresses separated by semicolon.
#   Multiple Zabbix servers/clusters and Zabbix proxies can be specified, separated by comma.
#   More than one Zabbix proxy should not be specified from each Zabbix server/cluster.
#   If Zabbix proxy is specified then Zabbix server/cluster for that proxy should not be specified.
#   Multiple comma-delimited addresses can be provided to use several independent Zabbix servers in parallel. Spaces are allowed.
#   If port is not specified, default port is used.
#   IPv6 addresses must be enclosed in square brackets if port for that host is specified.
#   If port is not specified, square brackets for IPv6 addresses are optional.
#   If this parameter is not specified, active checks are disabled.
#   Example for Zabbix proxy:
#       ServerActive=127.0.0.1:10051
#   Example for multiple servers:
#       ServerActive=127.0.0.1:20051;zabbix.domain[:1]:30051,:1,[12fc::1]
#   Example for high availability:
#       ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster.node3
#   Example for high availability with two clusters and one server:
#       ServerActive=zabbix.cluster.node1;zabbix.cluster.node2:20051;zabbix.cluster2.node1;zabbix.cluster2.node2;zabbix.domain
#
# Mandatory: no
# Default:
# ServerActive=

ServerActive=158.193.214.162

### Option: Hostname
#   List of comma delimited unique, case sensitive hostnames.
#   Required for active checks and must match hostnames as configured on the server.
#   Value is acquired from HostnameItem if undefined.
#
# Mandatory: no
# Default:
# Hostname=

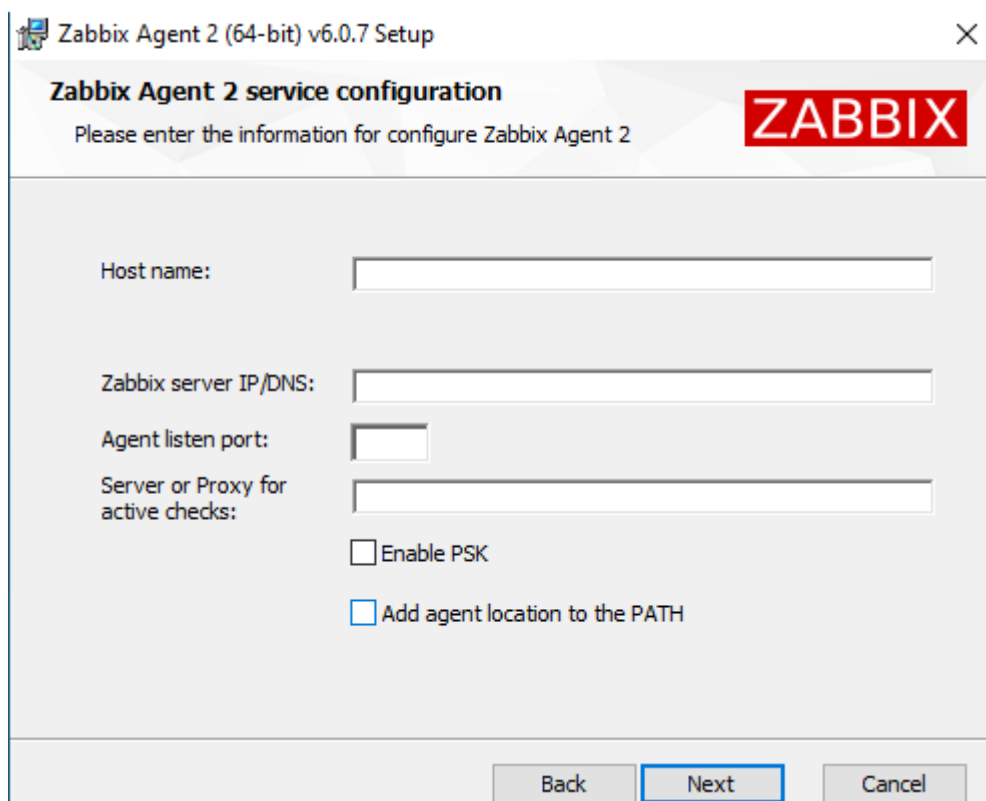
Hostname=z100
```

Obr. 6: Konfiguračný súbor Zabbix agenta - Linux

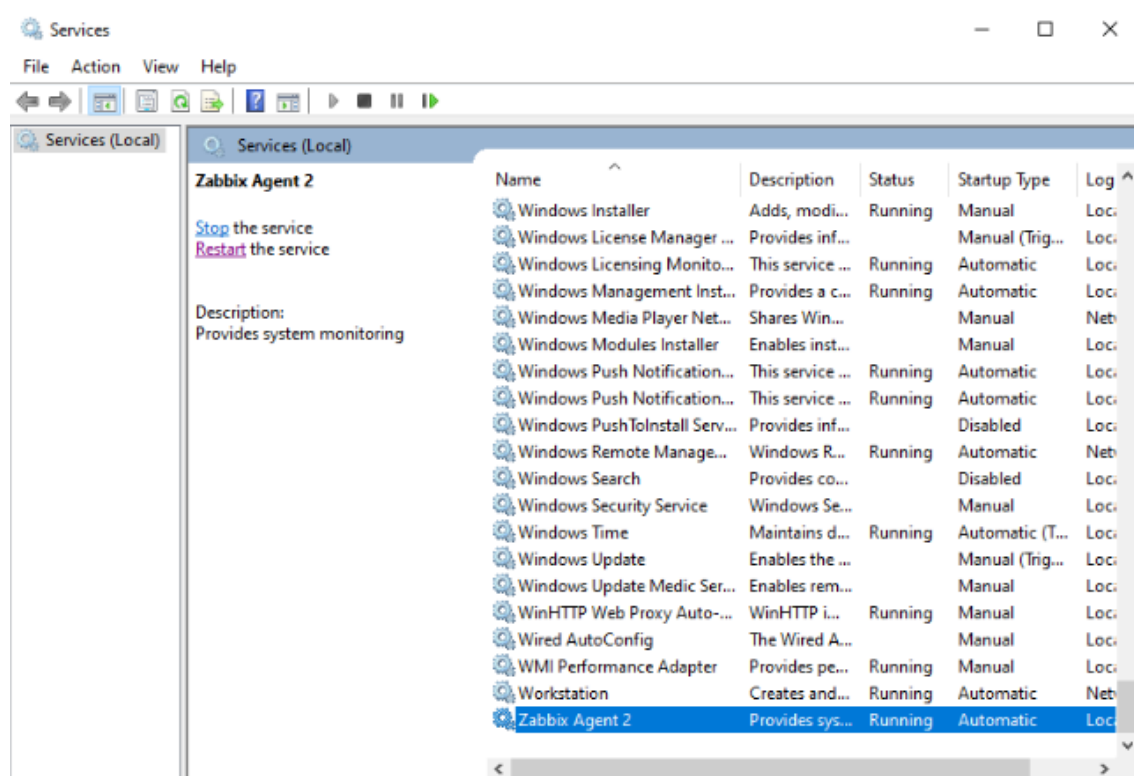
```
root@z100:~# systemctl status zabbix-agent2
● zabbix-agent2.service - Zabbix Agent 2
   Loaded: loaded (/lib/systemd/system/zabbix-agent2.service; enabled; vendor preset: enabled)
   Active: active (running) since Fri 2024-04-26 13:28:39 UTC; 1 week 2 days ago
     Main PID: 7781 (zabbix_agent2)
        Tasks: 8 (limit: 33547)
       Memory: 8.4M
          CPU: 7min 19.390s
     CGroup: /system.slice/zabbix-agent2.service
             └─7781 /usr/sbin/zabbix_agent2 -c /etc/zabbix/zabbix_agent2.conf
```

Obr. 7: Zobrazenie služby Zabbix agent - Linux

V prípade Windows zariadenia, bolo potrebné stiahnuť a nainštalovať Zabbix agenta z oficiálnej stránky Zabbix. Špecifikácia parametrov prebiehala počas inštalácie, vid'. Obr. 8. Úspešne implementovaného Zabbix agenta môžeme vidieť na obrázku Obr. 9.



Obr. 8: Konfiguračné okno Zabbix agenta - Windows



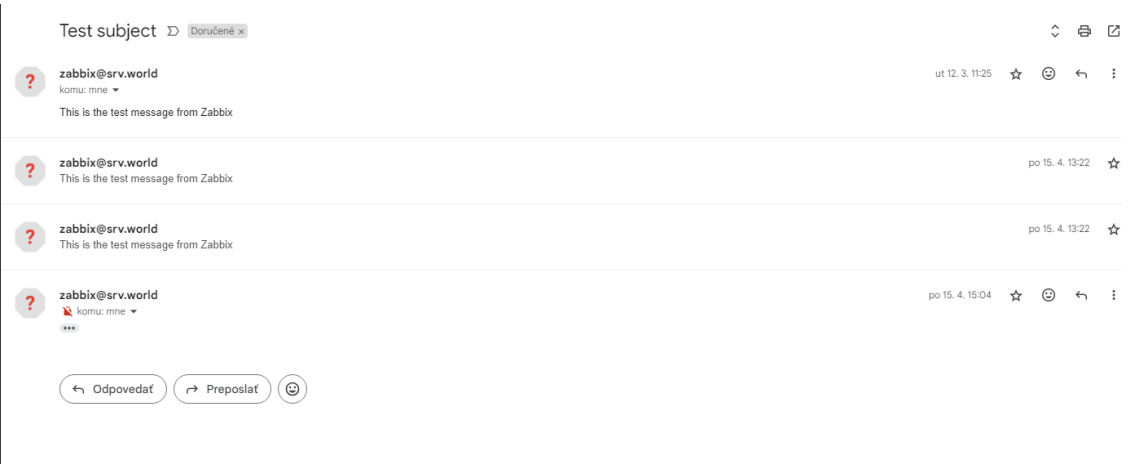
Obr. 9: Zobrazenie služby Zabbix agent - Windows

### 5.1.4 Upozornenia – e-mail

Zasielanie upozornení pomocou e-mailu fungovalo prostredníctvom nástroja Postfix. Tento nástroj slúži na konfiguráciu SMTP servera prostredníctvom konfiguračného súboru. Špecifikované bolo veľké množstvo parametrov ako adresa siete, povolenie IPv4 a IPv6, SMTP helo, autentifikácia a zákazy spamu a mnoho ďalších. Po úspešnej konfigurácii SMTP servera bolo nutnosťou nakonfigurovať e-mail vo webovom rozhraní Zabbix. Následne nám prichádzali upozornenia na špecifikovanú e-mailovú adresu, vid'. Obr. 10 a Obr. 11.

<input type="checkbox"/>	☆	zabbix 4	Test subject - This is the test message from Zabbix	15. 4.
<input type="checkbox"/>	☆	zabbix	Problem: Linux: Zabbix_server has been restarted (uptime < 10m) - Problem started at 13:14:00 on ...	15. 4.
<input type="checkbox"/>	☆	zabbix	Problem: Linux: Operating system description has changed - Problem started at 15:55:07 on 2024....	15. 4.
<input type="checkbox"/>	☆	zabbix	<b>Problem: Zabbix server: More than 100 items having missing data for more than 10 minutes - P</b>	<b>15. 4.</b>
<input type="checkbox"/>	☆	zabbix	Resolved in 9m 0s: Linux: Zabbix_server has been restarted (uptime < 10m) - Problem has been res...	15. 4.
<input type="checkbox"/>	☆	zabbix	Problem: Test down - Problem started at 18:00:00 on 2024.04.09 Problem name: Test down Host: Z...	15. 4.
<input type="checkbox"/>	☆	zabbix	<b>Problem: Zabbix server: More than 100 items having missing data for more than 10 minutes - P</b>	<b>15. 4.</b>
<input type="checkbox"/>	☆	zabbix	<b>Resolved in 21h 16m 0s: Zabbix server: More than 100 items having missing data for more th...</b>	<b>15. 4.</b>
<input type="checkbox"/>	☆	zabbix	<b>Problem: Zabbix agent: Zabbix agent is not available (for 3m) - Problem started at 10:12:57 on ...</b>	<b>15. 4.</b>

Obr. 10: Automatické e-mailové upozornenia Zabbix

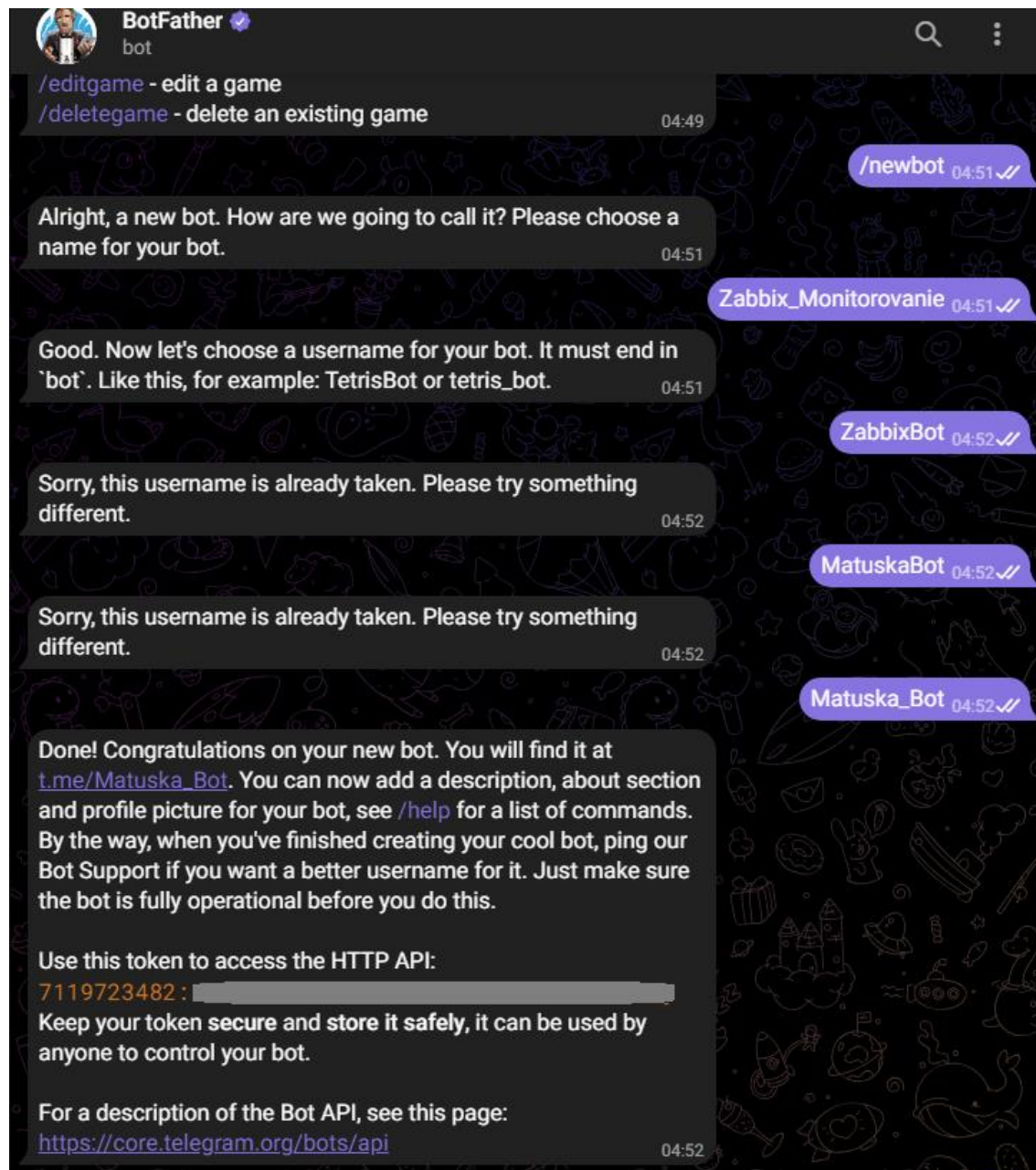


Obr. 11: Ukážka doručených e-mailov



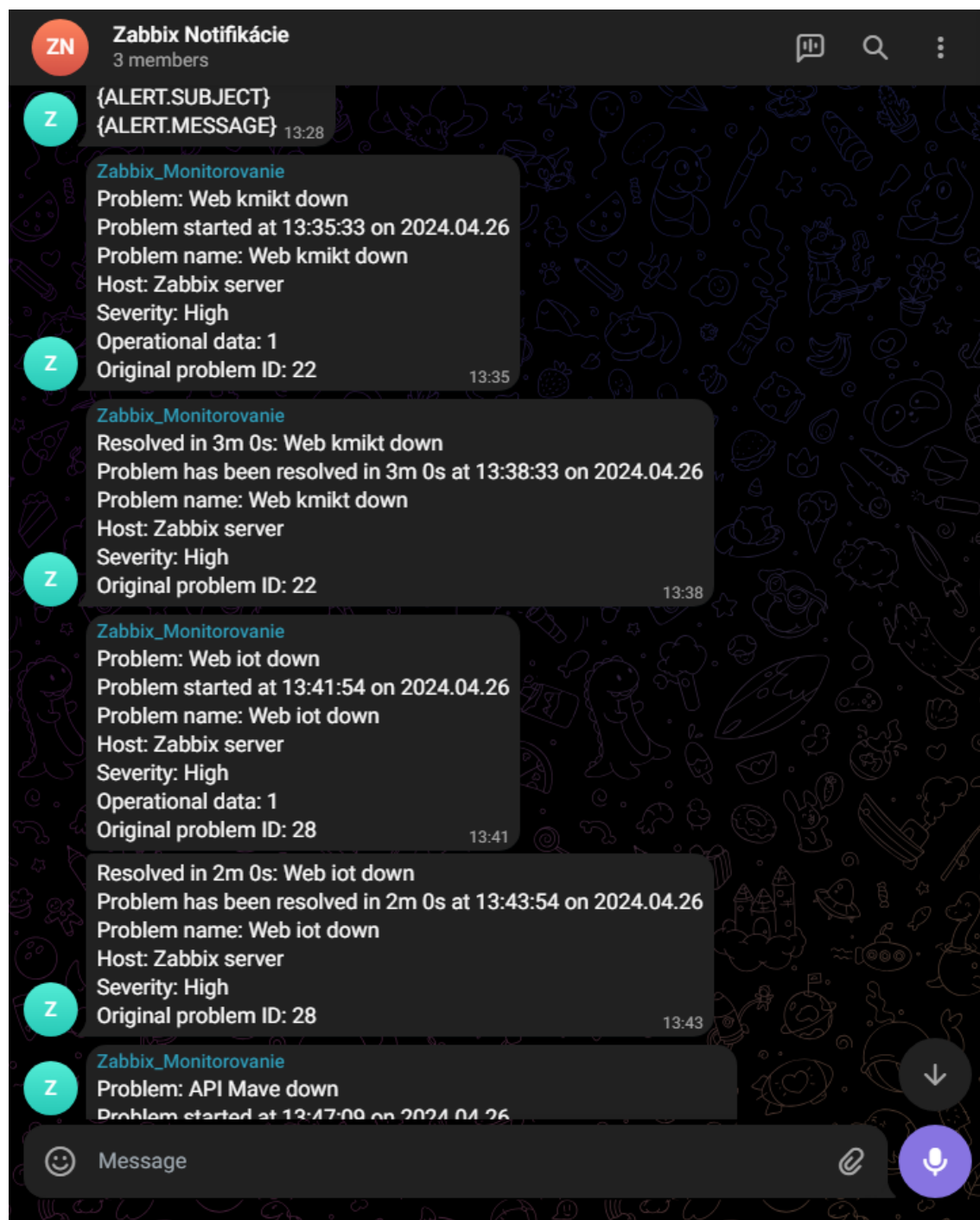
### 5.1.5 Upozornenia – Telegram

Možnosť zasielania automatických upozornení cez Telegram bola uskutočnená pomocou Telegram bota. Telegram bot bol vytvorený pomocou natívneho nástroju BotFather, ktorý slúži na vytváranie a správu Telegram botov, viď. Obr. 12.



Obr. 12: Vytváranie Telegram bota

Botovi bol pridelený unikátny token (kód), ktorý bol pridaný do Zabbixu na zasielanie automatizovaných správ cez Telegram vďaka nášmu botovi. Aby sme mali skupinu alebo miesto kde budú správy odosielané, vytvorili sme novú skupinu na komunikáciu, ktorej sme tohoto bota priradili, aby vedel kam odosielať správy, viď. Obr. 13.



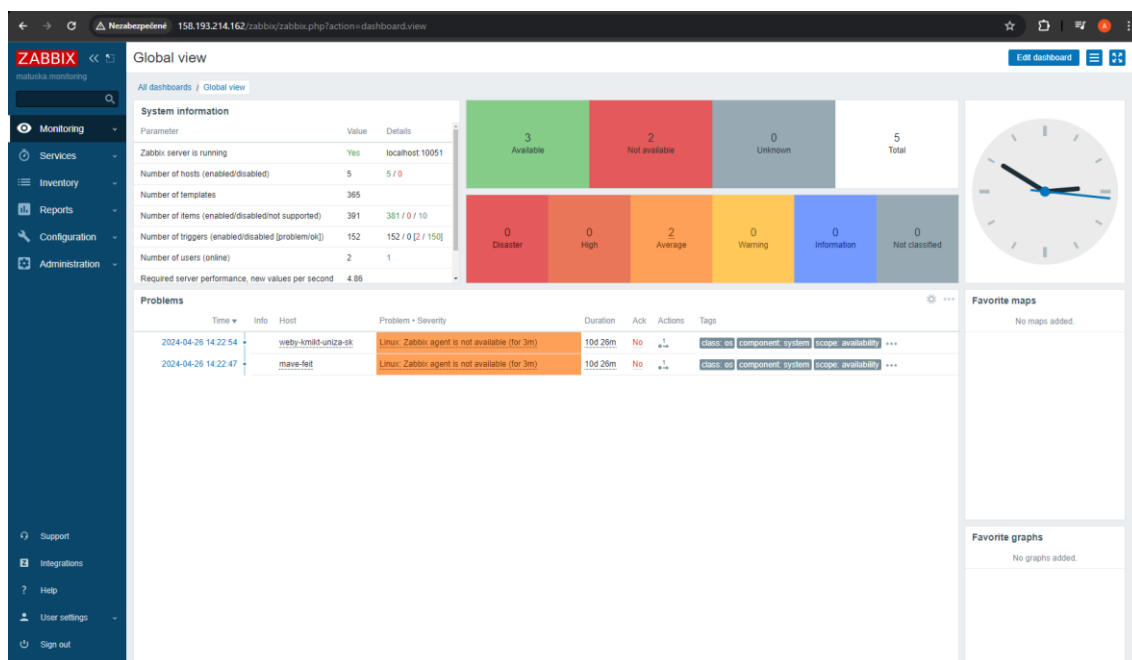
Obr. 13: Skupina pre automatické Zabbix upozornenia

## 5.2 Webové rozhranie Zabbix

V tejto časti si povieme ako prebiehalo konfigurovanie cez webové rozhranie Zabbix. Predstavíme si ukážky monitorovania a spracovania dát, ako sú vytvárané upozornenia a taktiež vysvetlíme akým spôsobom sú monitorované webové stránky a REST API. Vysvetlíme si ako sme vytvárali upozornenia na mieru.

### 5.2.1 Prehľad webového rozhranie Zabbix

Po úspešnej inštalácii a konfigurácii všetkých potrebných prostriedkov a nastavení úvodného webového rozhrania Zabbix sme sa mohli prihlásiť do finálnej verzie webového rozhrania. Tu prebiehali dodatočné konfigurácie. Po prvotnom prihlásení vidíme domovskú stránku Zabbix webového rozhrania, ktorá obsahuje súhrn najdôležitejších informácií rozdelených do blokov, vid'. Obr. 14.

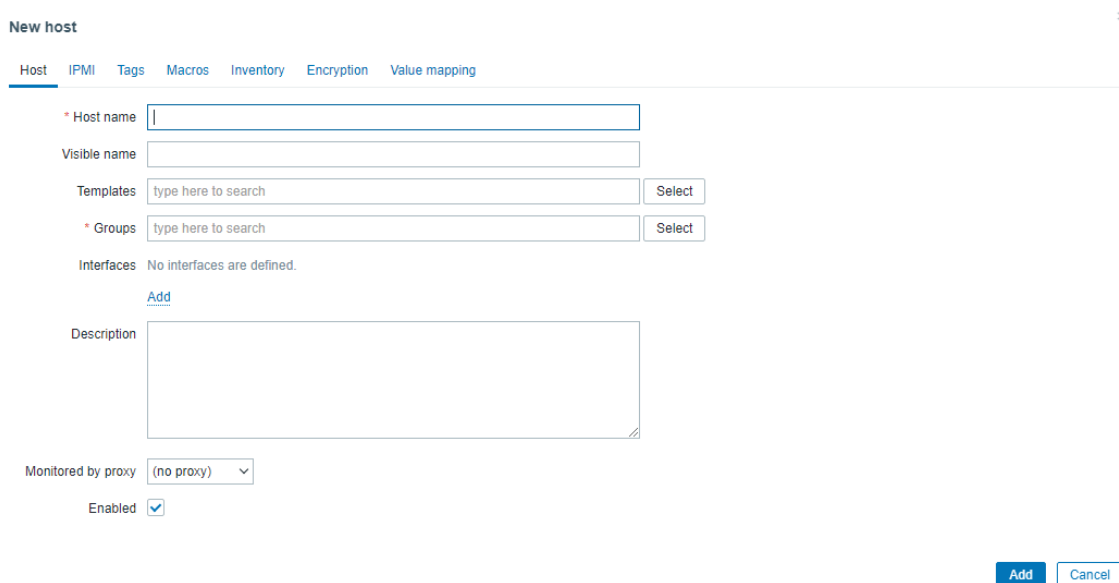


Obr. 14: Úvodná stránka Zabbix webového rozhrania

Na ľavej strane je hlavné menu v ktorom máme rozdelené jednotlivé možnosti ako napríklad sledovanie stavu zariadení v rôznych rovinách s množstvom filtrov (Monitoring). Pre nás ako administrátora bola hlavná záložka Configuration, kde prebiehala najväčšia časť konfigurácia na strane webového rozhrania. Ďalšiu dôležitú záložku predstavovala Administration, kde sme upravovali užívateľské nastavenia a taktiež v nej nájdeme Media Types, ktoré slúžili pri konfigurácii zasielania upozornení prostredníctvom e-mailu a Telegramu.

## 5.2.2 Pridávanie sieťových zariadení

Vysvetlili sme si ako sme na jednotlivých sieťových zariadenia inštalovali a konfigurovali Zabbix agenta. Tento proces bol na strane konkrétneho zariadenia. Teraz si ukážeme ako vyzerá pridávanie sieťových zariadení na strane servera alebo teda vo webovom rozhraní Zabbix. Predpokladajme, že na našom sieťovom zariadení bol úspešne nainštalovaný a nakonfigurovaný Zabbix agent. Ostáva nám už len zistiť hostname zariadenia a jeho IP adresu. Následne si v záložke Configuration otvoríme panel Hosts a v pravom hornom rohu klikneme na tlačidlo Create host. Vyskočí nám okno na priradenie parametrov pre konkrétne sieťové zariadenie. Špecifikujeme názov, šablónu, do akej skupiny chceme toto zariadenie priradiť, a ako posledné vyberieme interface Agent, ktorému priradíme IP adresu, DNS a port, viď. Obr. 15. Po kliknutí na tlačidlo Add pridáme sieťové zariadenie a po pár minútach môžeme sledovať všetky zozbierané dáta prostredníctvom záložky Monitoring. Týmto spôsobom pridávame sieťové zariadenia s operačným systémom Linux a taktiež Windows.



The screenshot shows the 'New host' form in the Zabbix web interface. The form is titled 'New host' and has a close button (X) in the top right corner. Below the title, there are tabs for 'Host', 'IPMI', 'Tags', 'Macros', 'Inventory', 'Encryption', and 'Value mapping'. The 'Host' tab is selected. The form contains the following fields and controls:

- \* Host name:** A text input field.
- Visible name:** A text input field.
- Templates:** A text input field with the placeholder 'type here to search' and a 'Select' button.
- \* Groups:** A text input field with the placeholder 'type here to search' and a 'Select' button.
- Interfaces:** A section with the text 'No interfaces are defined.' and an 'Add' link.
- Description:** A large text area.
- Monitored by proxy:** A dropdown menu with the selected value '(no proxy)'.
- Enabled:** A checkbox that is checked.
- Buttons:** 'Add' and 'Cancel' buttons at the bottom right.

Obr. 15: Pridávanie nového zariadenia

### 5.2.3 Zozbierané dáta z monitorovacích cieľov

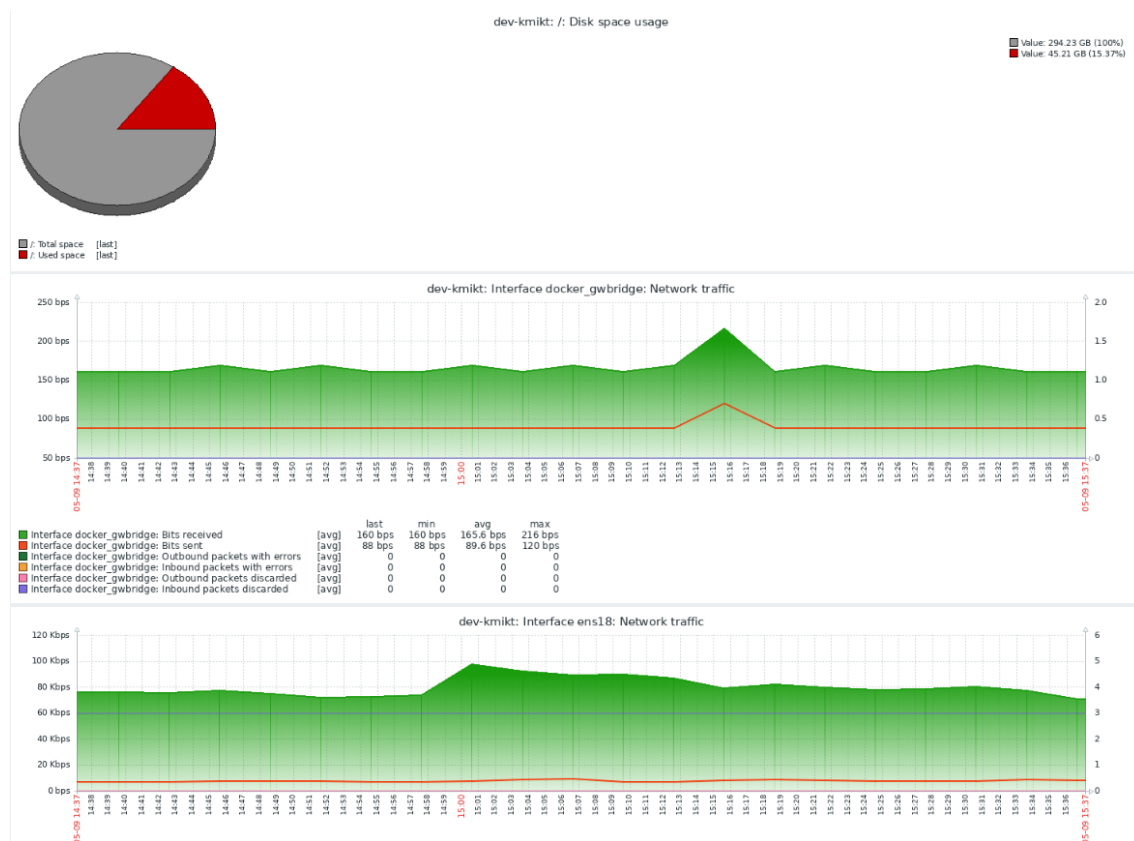
Webové rozhranie Zabbix ponúka širokú škálu možnosti zobrazovania zozbieraných dát od svojich zariadení, ktoré monitorujeme. Či už to je vo forme zápisov alebo grafickou formou pomocou grafov. Už len samotné prednastavené šablóny ponúkajú obrovské množstvo dát, ktoré sú cez Zabbix agentov posielané na server. Preto si ukážeme iba určitú časť dát v rôznych formách.

Ako prvé bude ukážka zozbieraných dát vo forme zápisov zo zariadenia dev-kmikt, vid'. Obr. 16.

<input type="checkbox"/> Host	Name ▲	Last check	Last value	Change	Tags	
<input type="checkbox"/> dev-kmikt	/: Free inodes in %	30s	97.2929 %		component: storage filesystem: /	Graph
<input type="checkbox"/> dev-kmikt	/: Space utilization <sup>?</sup>	29s	16.1922 %	+0.000006 %	component: storage filesystem: /	Graph
<input type="checkbox"/> dev-kmikt	/: Total space <sup>?</sup>	28s	294.23 GB		component: storage filesystem: /	Graph
<input type="checkbox"/> dev-kmikt	/: Used space <sup>?</sup>	27s	45.21 GB	+16 KB	component: storage filesystem: /	Graph
<input type="checkbox"/> dev-kmikt	Interface docker_gwbridge: Bits received	2m 25s	160 bps	-8 bps	component: network interface: docker_gw...	Graph
<input type="checkbox"/> dev-kmikt	Interface docker_gwbridge: Bits sent	2m 23s	88 bps		component: network interface: docker_gw...	Graph
<input type="checkbox"/> dev-kmikt	Interface docker_gwbridge: Inbound packets discarded	2m 21s	0		component: network interface: docker_gw...	Graph
<input type="checkbox"/> dev-kmikt	Interface docker_gwbridge: Inbound packets with errors	2m 19s	0		component: network interface: docker_gw...	Graph
<input type="checkbox"/> dev-kmikt	Interface docker_gwbridge: Interface type <sup>?</sup>	23h 26m 17s	Ethernet (1)		component: network interface: docker_gw...	Graph
<input type="checkbox"/> dev-kmikt	Interface docker_gwbridge: Operational status <sup>?</sup>	15s	up (6)		component: network interface: docker_gw...	Graph
<input type="checkbox"/> dev-kmikt	Interface docker_gwbridge: Outbound packets discarded	2m 13s	0		component: network interface: docker_gw...	Graph
<input type="checkbox"/> dev-kmikt	Interface docker_gwbridge: Outbound packets with errors	2m 11s	0		component: network interface: docker_gw...	Graph
<input type="checkbox"/> dev-kmikt	Interface docker_gwbridge: Speed <sup>?</sup>	1m 9s	10 Gbps		component: network interface: docker_gw...	Graph
<input type="checkbox"/> dev-kmikt	Interface ens18: Bits received	2m 26s	77.22 Kbps	-2.99 Kbps	component: network interface: ens18	Graph
<input type="checkbox"/> dev-kmikt	Interface ens18: Bits sent	2m 24s	8.55 Kbps	+1.38 Kbps	component: network interface: ens18	Graph
<input type="checkbox"/> dev-kmikt	Interface ens18: Inbound packets discarded	2m 22s	3		component: network interface: ens18	Graph
<input type="checkbox"/> dev-kmikt	Interface ens18: Inbound packets with errors	2m 20s	0		component: network interface: ens18	Graph
<input type="checkbox"/> dev-kmikt	Interface ens18: Interface type <sup>?</sup>	23h 26m 18s	Ethernet (1)		component: network interface: ens18	Graph
<input type="checkbox"/> dev-kmikt	Interface ens18: Operational status <sup>?</sup>	16s	up (6)		component: network interface: ens18	Graph
<input type="checkbox"/> dev-kmikt	Interface ens18: Outbound packets discarded	2m 14s	0		component: network interface: ens18	Graph
<input type="checkbox"/> dev-kmikt	Interface ens18: Outbound packets with errors	2m 12s	0		component: network interface: ens18	Graph
<input type="checkbox"/> dev-kmikt	Interface ens18: Speed <sup>?</sup>				component: network interface: ens18	Graph
<input type="checkbox"/> dev-kmikt	Linux: Available memory <sup>?</sup>	39s	31.68 GB	-12.07 MB	component: memory	Graph

Obr. 16: Ukážka zozbieraných dát zo zariadenia vo forme zápisov

Ako druhé bude ukážka zozbieraných dát vo forme grafov zo zariadenie dev-kmikt, vid'. Obr. 17.



Obr. 17: Ukážka zozbieraných dát zo zariadenia vo forme grafov

## 5.2.4 Monitorovanie webových lokalít

Konfigurácia monitorovania webových lokalít prebieha v záložke Configuration. Na serveri sme vybrali tlačidlo Web, kde môžeme pridávať web scenario alebo teda webové stránky na monitorovanie. Spôsobov ako a čo monitorovať je zase veľmi veľa. V našom prípade stačilo monitorovať dostupnosť stránky, čiže monitorovanie spočívalo v HTTP status kódoch. Na pridanie stránky na monitorovanie bolo potrebné vytvoriť web scenario. Špecifikovali sme názov, v akom intervale bude prebiehať zber dát, koľko krát po sebe v prípade zlyhania a agenta pomocou ktorého budú dáta zbierané, vid'. Obr. 18.

Web monitoring

All hosts / Zabbix server Enabled ZBX Items 113 Triggers 73 Graphs 21 Discovery rules 4 Web scenarios 8

Scenario Steps Tags Authentication

\* Name

\* Update interval

\* Attempts

Agent

HTTP proxy

Variables

Name	Value	
<input type="text" value="name"/>	<input type="text" value="value"/>	Remove

[Add](#)

Headers

Name	Value	
<input type="text" value="name"/>	<input type="text" value="value"/>	Remove

[Add](#)

Enabled ☒

[Add](#) [Cancel](#)

Obr. 18: Pridávanie novej webovej lokality



V hornej lište môžeme vidieť záložku Steps, v tejto záložke špecifikujeme názov, URL web stránky, Timeout hovorí o tom ako dlho sa bude čakať na odpoveď od webovej stránky, podmienky a spôsob monitorovania, vid'. Obr.19. V spodnej časti môžeme vidieť, že na konkrétnej URL môžeme vyhľadávať konkrétne slovo alebo súbor znakov. V našom prípade sme využívali HTTP status kódy.

Step of web scenario

\* Name

\* URL  Parse

Query fields

Name	Value	
name	=	value Remove

Add

Post type Form data Raw data

Post fields

Name	Value	
name	=	value Remove

Add

Variables

Name	Value	
name	=	value Remove

Add

Headers

Name	Value	
name	=	value Remove

Add

Follow redirects ☐

Retrieve mode Body Headers Body and headers

\* Timeout

Required string

Required status codes

Obr. 19: Parametre a podmienky monitorovania webovej lokality



## 5.2.5 Monitorovanie REST API

Konfigurácia monitorovania webových lokalít prebieha v záložke Configuration. Na serveri sme vybrali tlačidlo Items. Vytvorenie prebiehalo cez tlačidlo Create item, kde sme špecifikovali množstvo parametrov ako názov, typ agenta, URL, HTTP požiadavku, telo požiadavky a iné, vid'. Obr. 20.

The screenshot displays the 'Items' configuration page in Zabbix. The breadcrumb trail at the top reads: 'All hosts / Zabbix server / Enabled / 20x / Items 113 / Triggers 73 / Graphs 21 / Discovery rules 4 / Web scenarios 6'. The 'Item' tab is selected, and the 'Preprocessing' sub-tab is active. The configuration form includes the following fields and options:

- Name:** API Mave
- Type:** HTTP agent
- Key:** post
- Type of information:** Text
- URL:** https://mave.felt.uniza.sk/api/public/signin
- Query fields:** A table with columns 'Name' and 'Value'. One field is added with 'name' as the key and 'value' as the value.
- Request type:** POST
- Timeout:** 3s
- Request body type:** JSON data (selected), Raw data, XML data
- Request body:** {"email": "test@user.com", "password": "SuperHeslo"}
- Headers:** A table with columns 'Name' and 'Value'. One field is added with 'name' as the key and 'value' as the value.
- Required status codes:** 400
- Follow redirects:** Checked
- Retrieve mode:** Body (selected), Headers, Body and headers
- Convert to JSON:** Unchecked
- HTTP proxy:** [protocol://user[:password]@proxy.example.com[:port]]
- HTTP authentication:** None

*Obr. 20: Pridávanie nového REST API*

V našom prípade sme používali metódu POST, keďže sme monitorovali iba dostupnosť REST API na základe POST požiadavky a čakali sme na odpoveď.

## 5.2.6 Trigger – vlastné upozornenia

Tvorba vlastných upozornení spočíva v dobre vytvorených podmienkach. Ukážeme si trigger vytvorený pre monitorovanie webovej lokality a trigger pre monitorovanie REST API. V prvom rade bolo nutné vymyslieť logiku, na základe ktorej sa trigger aktivuje.

Pri webovej lokalite sme brali do úvahy HTTP status kód, čiže v prípade že stránka je funkčná očakávame kód 200 OK. Tu sme využili vytvorený Step vo web scenario, kde sme špecifikovali, že každú minútu sa pošle požiadavka na webovú stránku a v prípade, že odpoveďou nebude kód 200 OK, tak Step zlyhal a môžeme si zobrať jeho hodnotu do podmienky. V prípade, že Step zlyhal jeho hodnota je True, čiže v našom prípade 1. V opačnom prípade nezlyhal a jeho hodnota je False, čiže 0. Na základe tejto hodnoty vieme vytvoriť trigger s podmienkou, viď. Obr. 21. Špecifikovali sa rôzne parametre ako názov, závažnosť upozornenia, samotná podmienka a iné.

The screenshot shows the Zabbix Triggers configuration page. The trigger is named "Web kmikt down" and has a severity of "High". The expression is "last(/Zabbix server/web.test.fail[kmikt],#5)=1". The trigger is enabled and has buttons for Update, Clone, Delete, and Cancel.

Triggers

All hosts / Zabbix server Enabled ZBX Items 113 Triggers 73 Graphs 21 Discovery rules 4 Web scenarios 8

Trigger Tags Dependencies

Name Web kmikt down

Event name Web kmikt down

Operational data

Severity Not classified Information Warning Average High Disaster

Expression last(/Zabbix server/web.test.fail[kmikt],#5)=1 Add

Expression constructor

OK event generation Expression Recovery expression None

PROBLEM event generation mode Single Multiple

OK event closes All problems All problems if tag values match

Allow manual close

URL

Description

Enabled

Update Clone Delete Cancel

Obr. 21: Trigger na monitorovanie webovej lokality

Je nutné dodať, že v podmienke metóda „last“ predstavuje poslednú odpoveď od webovej stránky a hodnota „#5“ predstavuje koľko po sebe nasledujúcich odpovedí musí zlyhať aby sa trigger aktivoval.

Pri tvorbe triggera pre REST API sme postupovali podobne. Keďže monitorujeme dostupnosť, ale nemáme k dispozícii možnosť monitorovania iba cez HTTP status kódy, pretože používame POST požiadavku, tak očakávame odpoveď vo forme textu. V našom prípade sme použili ako telo správy POST požiadavky prihlasovacie údaje a očakávali sme HTTP status kód 400. V prípade, že REST API bola dostupná, dostali sme odpoveď vo forme textu. Na základe tohoto sme mohli vytvoriť podmienku na náš trigger. Spočívalo to v tom, že ak sme dostali nejaké dáta ako odpoveď, považovali sme ju za dostupnú. Tento proces sa opakoval každú minútu. V prípade, že odpoveď prišla, predstavovalo to hodnotu False, čiže 0. V prípade, že odpoveď neprišla, predstavovalo to hodnotu True, čiže 1. Podmienka teda vyzerala ako na obrázku Obr. 22.

Triggers

All hosts / Zabbix server Enabled ZBX Items 113 Triggers 73 Graphs 21 Discovery rules 4 Web scenarios 8

Trigger Tags Dependencies

Name API Mave down

Event name API Mave down

Operational data

Severity Not classified Information Warning Average High Disaster

\* Expression nodata(/Zabbix server/post,5m)=1 Add

Expression constructor

OK event generation Expression Recovery expression None

PROBLEM event generation mode Single Multiple

OK event closes All problems All problems if tag values match

Allow manual close ☐

URL

Description

Enabled ☒

Update Clone Delete Cancel

Obr. 22: Trigger na monitorovanie REST API

Metóda „nodata“ predstavovala práve True / False podmienku, kde sme monitorovali, či vôbec nejaké dáta boli prijaté. Hodnota „5m“ predstavuje ako dlho nemôže prísť odpoveď od REST API, kým sa trigger aktivuje.

## 5.2.7 Telegram

Telegram slúži ako služba na zasielanie upozornení. Vo webovom rozhraní Zabbix, je nutná jeho konfigurácia. V zložke Administration zvolíme panel Media Types a vyberieme Telegram. Hlavné zložky na špecifikovanie parametrov sú unikátne číslo skupiny, kde sa budú automatizované správy odosielať a unikátny token Telegram bota, na základe ktorého Zabbix vie akého bota použiť, vid'. Obr. 23.

### Media types

The screenshot shows the 'Media types' configuration page in Zabbix. The 'Media type' tab is selected, showing the configuration for 'Telegram'. The 'Type' is set to 'Webhook'. The 'Parameters' table lists fields for Message, ParseMode, Subject, To, and Token, each with a 'Remove' button. The 'Script' field contains a placeholder 'var Telegram = {...}'. The 'Timeout' is set to '10s'. There are checkboxes for 'Process tags' and 'Include event menu entry'. The 'Menu entry name' and 'Menu entry URL' fields are empty. The 'Description' field contains a URL and instructions for setting up the bot. The 'Enabled' checkbox is checked. At the bottom are buttons for 'Update', 'Clone', 'Delete', and 'Cancel'.

Name	Value	Action
Message	{ALERT.MESSAGE}	Remove
ParseMode		Remove
Subject	{ALERT.SUBJECT}	Remove
To	-4	Remove
Token	7119723482	Remove

Script: `var Telegram = {...`

Timeout: 10s

Process tags: ☐

Include event menu entry: ☐

Menu entry name:

Menu entry URL:

Description: <https://git.zabbix.com/projects/ZBX/repos/zabbix/browse/templates/media/telegram>

1. Register bot: send "/newbot" to @BotFather and follow instructions  
2. Copy and paste the obtained token into the "Token" field above  
3. If you want to send personal notifications, you need to get chat id of the user you want to send messages to:

Enabled: ☒

Update Clone Delete Cancel

Obr. 23: Konfigurácia automatických upozornení cez Telegram

Dôležité je podotknúť, že je nutné aby mal užívateľ povolené zasielanie upozornení prostredníctvom monitorovacieho systému Zabbix. Taktiež je nutné pridelit' konkrétnym užívateľom, cez ktoré služby im môžu byť zasielané upozornenia, v našom prípade je nutnosť pridelit' Telegram.

## 5.3 Výsledky práce a diskusia

Výsledkom našej práce je implementácia monitorovacieho systému Zabbix na reálnu sieťovú infraštruktúru Žilinskej univerzity. Môžeme zhodnotiť, že všetky ciele práce sa nám podarilo uskutočniť. Beh Zabbixu prebieha bez prestávky na virtuálnom serveri. To znamená, že v prípade výpadku alebo iného problému monitorovaných cieľov vieme nepretržite sledovať každé zariadenie cez webové rozhranie. Spôsob monitorovania pomocou webového rozhrania nie je úplne efektívny, preto sme využili odosielanie upozornení. Znamená to, že v prípade problému, ktorý bol definovaný ako vhodný na upozornenie administrátora, odošle sa správa prostredníctvom Telegramu alebo e-mailu. Výhoda takéhoto riešenia spočíva v tom, že administrátor môže dostať upozornenie, ktoré môže vidieť aj na mobile a následne tento problém vyriešiť vzdialene prostredníctvom webového rozhrania v prípade ak má prihlasovacie údaje do administrátorského účtu a prístup na internet.

Pri inštalácii Zabbixu sme sa stretli s viacerými prekážkami. Keďže sme na účely testovania používali najskôr virtuálne prostredie, vytvorili sme si lokálnu sieť (LAN). V tejto sieti sme nemuseli brať do úvahy bezpečnosť, pričom po aplikácii Zabbixu do reálnej sieťovej infraštruktúry s verejnou IP adresou, sme museli dbať na bezpečnosť a nastavenie hesiel na príslušných službách ako napr. databáza a účet administrátora.

Vo virtuálnom prostredí sme sieťové zariadenie vytvárali s rovnakými distribúciami operačného systému Linux, takže nevznikali problémy na základe kompatibility Zabbix agentov a samotného Zabbixu. S týmto problémom kompatibility medzi verziami distribúcií sme sa stretli pri implementovaní Zabbixu na reálne zariadenia.

Ako hlavný výsledok práce môžeme považovať vytvorenie užívateľského manuálu, ktorý je súčasťou tejto práce a bol priložený v elektronickej forme ako príloha. Manuál obsahuje kompletnú inštrukciu na inštaláciu nástrojov na serveri použitých v tejto práci ako napr. Apache2, PHP, PHP-FPM, MariaDB a iné. Súčasťou je aj kompletný návod na nutnú konfiguráciu týchto nástrojov. Ďalším obsahom manuálu je návod na inštaláciu a konfiguráciu Zabbix agenta, či už na sieťových zariadeniach s operačným systémom Microsoft Windows alebo Linux Ubuntu. Následne bola uvedená podrobná konfigurácie vo webovom rozhraní. Pridávanie monitorovacích cieľov, pridávanie monitorovania webových stránok, pridávanie monitorovania REST API, konfigurácie upozornení, povolenie upozornení a iné. Manuál slúži ako možnosť replikácie tejto práce.

## Záver

V tejto práci sme podrobne preskúmali tému monitorovania zariadení a výkonnosti IT infraštruktúry, ako aj existujúce softvérové nástroje na túto úlohu. Sme presvedčení, že monitorovanie zariadení je kľúčové pre riadne fungovanie a správu moderných informačných technológií. Monitorovanie zariadení je veľmi efektívny spôsob na udržiavanie stability monitorovacích cieľov pre komerčné aj osobné účely.

V prvej časti sme si povedali vo všeobecnosti o princípe monitorovania sieťových zariadení alebo o monitorovaní ako celku. Následne sme spracovali už konkrétne funkcie a procesy, o ktorých môžeme hovoriť pri monitorovacích softvéroch. Zahrňujeme tu monitorovanie využitia CPU, disku a RAM pamäte, analýzu siete a chybovosti, ako aj monitorovanie šírky pásma a konfigurácie zariadení. Okrem toho sme sa venovali aj prístrojovým panelom, vzdialenému prístupu a sledovaniu dostupnosti zariadení.

V druhej časti sme si spravili rozbor iných dostupných monitorovacích softvérov. Ako prvý bol Sematext Monitoring. Sematext ponúka komplexné monitorovanie aplikácií a infraštruktúr, čo je jeho silnou stránkou. Avšak, jeho cena môžu byť nevýhodami, najmä pre menšie firmy s obmedzeným rozpočtom. Ako druhý sme skúmali monitorovací systém Microsoft System Center. Microsoft System Center je softvérový balík od firmy Microsoft, navrhnutý na zjednodušenie správy IT infraštruktúry a virtuálnych dátových centier. Jeho výhody sú široké pokrytie funkcií od monitorovania po zálohovanie a podporu hybridných cloudových prostredí. Nevýhodou môže byť cena, najmä pre menšie firmy. Ako posledný spracovaný monitorovací softvér sme zvolili New Relic. New Relic je softvér s voľne dostupným zdrojovým kódom. Ponúka široké spektrum funkcií vrátane monitorovania aplikácií, analýzy webových stránok a monitorovania. Jeho výhody zahŕňajú sledovanie stavu aplikácií v reálnom čase, flexibilitu v monitorovaní pomocou jazykových agentov a možnosť monitorovať skutočných používateľov pomocou monitorovania prehliadača. Avšak, nevýhodou môže byť potenciálna zložitosť a náklady na implementáciu a používanie.

V tretej časti sme už podrobnejšie rozoberali monitorovací systém Zabbix, ktorý bol náplňou našej záverečnej práce. Zabbix je softvér s voľne dostupným zdrojovým kódom, ktorý umožňuje monitorovanie rôznych parametrov IT infraštruktúry, vrátane siete, serverov, aplikácií a ďalších. S jeho flexibilným mechanizmom upozornení a možnosťou ich zasielania prostredníctvom rôznych služieb ako napríklad e-mail a Telegram.

Poskytuje funkcie reportovania a vizualizácie dát cez webové rozhranie, množstvo iných funkcií. Ako posledné sme zhrnuli z akých sieťových komponentov sa skladá.

Poslednú časť môžeme rozložiť na „serverovú časť“ a „časť webového rozhrania Zabbix“. V serverovej časti sme sa už zamerali na implementáciu softvéru Zabbix v reálnej sieťovej infraštruktúre. Najskôr sme sa zaoberali úvahou a spracovaním tejto práce, ako aj nutnými podmienkami pre dosiahnutie funkčného monitorovacieho softvéru Zabbix. Tieto podmienky predstavujú hlavne inštalácie a konfigurácie serveru a všetkých služieb na ňom, databázy, služieb na zasielanie upozornení a sieťových zariadení. V časti webového rozhrania Zabbix sme hovorili o úvodných nastaveniach webového rozhrania a následne konfigurácií v ňom. Konkrétne pridávanie sieťových zariadení na monitorovanie, webových stránok, REST API, vytváraní vlastných upozornení a služieb na ich zasielanie. Taktiež v tejto časti boli znázornené ukážky prijatých upozornení cez e-mail a Telegram, ukážka zozbieraných dát a ich spracovanie.

Budúce možné vylepšenia monitorovacieho systému Zabbix, by mohli spočívať napríklad v monitorovaní teplôt sieťových zariadení prostredníctvom softvérových doplnkov, ktoré nie sú natívne.

V praktickej časti našej práce sme využívali virtuálne prostredie na účely testovania, následne sme otestovaný systém implementovali na reálnu sieťovú infraštruktúru Žilinskej univerzity. Vďaka možnosti využitia softvéru v praxi bola táto práca vhodná pre osobný ale aj kariérny rozvoj.

## Zoznam použitej literatúry

- [1] Function Overview of System Monitor - Performance Monitoring Services. In: *System Monitor - Performance Monitoring Services Users Guide 5.6* [online]. First edition. s. 4-9. Dostupné z: [https://www.nec.com/en/global/prod/sigmasystemcenter/en/support/pdf/System\\_Monitor\\_UsersGuide\\_56E-1](https://www.nec.com/en/global/prod/sigmasystemcenter/en/support/pdf/System_Monitor_UsersGuide_56E-1)
  - [2] Server Monitoring Software. *G2 - Bussiness Software Reviews* [online]. 2023. Dostupné z: <https://www.g2.com/categories/server-monitoring>
  - [3] Sematext Monitoring. SEMATEXT GROUP. *Sematext Documentation* [online]. 2023. Dostupné z: <https://sematext.com/docs/monitoring/>
  - [4] Microsoft System Center. AWATI, Rahul. *TechTarget* [online]. 2023. Dostupné z: <https://www.techtarget.com/searchwindowsserver/definition/Microsoft-System-Center>
  - [5] Get Started with APM. NEW RELIC INC. *New Relic* [online]. Dostupné z: <https://docs.newrelic.com/docs/apm/new-relic-apm/getting-started/introduction-apm/>
  - [6] Get Started with browser monitoring. NEW RELIC INC. *New Relic* [online]. Dostupné z: <https://docs.newrelic.com/docs/browser/browser-monitoring/getting-started/introduction-browser-monitoring/>
  - [7] What is Zabbix. ZABBIX SIA. *Zabbix Documentation* [online]. Dostupné z: <https://www.zabbix.com/documentation/current/en/manual/introduction/about>
  - [8] Zabbix features. ZABBIX SIA. *Zabbix Documentation* [online]. Dostupné z: <https://www.zabbix.com/documentation/current/en/manual/introduction/features>
  - [9] Zabbix overview. ZABBIX SIA. *Zabbix Documentation* [online]. Dostupné z: <https://www.zabbix.com/documentation/current/en/manual/introduction/overiew>  
[w](#)
-



# Čestné vyhlásenie

Vyhlasujem, že som zadanú bakalársku prácu vypracoval samostatne, pod odborným vedením vedúceho bakalárskej práce Ing. Slavomíra Matušku, PhD. a používal som len literatúru uvedenú v práci.

V Žiline dňa 10. mája 2024

Andrej Pastorák

---

## **Prílohová časť**

---

## Zoznam príloh

Príloha: obsah priloženého USB.....	II
-------------------------------------	----

## **Príloha: obsah priloženého USB**

Obsahuje vypracovaný manuál v elektronickej forme.