Česká zemědělská univerzita v Praze

Provozně ekonomická fakulta

Katedra informačních technologií

Diplomová práce

Nasazení systému Zabbix

Bc. Petr Maroušek

ČESKÁ ZEMĚDĚLSKÁ UNIVERZITA V PRAZE

Provozně ekonomická fakulta

ZADÁNÍ DIPLOMOVÉ PRÁCE

Bc. Petr Maroušek

Informatika

Název práce

Nasazení systému Zabbix

Název anglicky

Deployment of Zabbix system

Cíle práce

Cílem této práce je instalace a konfigurace monitorovacího systému Zabbix v nejmenované společnosti. V průběhu vypracovávání práce je přiblíženo monitorovací prostředí a jsou popsány jednotlivé prvky a funkce, které tento system nabízí.
Dílčími cíli práce jsou:
- analýza a popis sledovaných zařízení
- stanovení sledovaných funkcí

Metodika

Metodika teoretické části této diplomové práce se zakládá na literární rešerši. V praktické části je pak založena na instalaci a konfiguraci aktuální verze monitorovacího systému Zabbix na virtuální server. Instalace a následná konfigurace je provedena na virtuálním serveru na platformě VMWare, který je pro tento systém vyhrazen v nejmenované společnosti. Při instalaci je použit instalační balíček systému Zabbix z oficiálních stránek. Jsou předvedeny jednotlivé kroky instalace a nakonec samotné spuštění programu.

Doporučený rozsah práce

60-80 stran

Klíčová slova

Zabbix, monitoring, nasazení, Cacti, Nagios, SNMP, agent, dohledový systém

---

Doporučené zdroje informací

DALLE VACCHE, Andrea a KEWAN LEE, Stefano. Mastering Zabbix monitor your large IT environment efficiently with Zabbix. Birmingham: Packt Pub, 2013. ISBN 9781783283491.

DALLE VACCHE, Andrea a KEWAN LEE, Stefano. Zabbix Network Monitoring Essentials. Birmingham: Packt Pub, 2015. ISBN 9781784399764.

SOSINSKY, Barrie. Mistrovství – počítačové sítě. Computer Press, Albatros Media, 2016. ISBN 9788025139165.

UYTTERGIEVEN, Patrik. Zabbix cookbook. Birmingham: Packt Publishing Limited, 2015. ISBN 9781784397586.

Předběžný termín obhajoby

2017/18 LS – PEF

Vedoucí práce

Ing. Martin Havránek, Ph.D.

Garantující pracoviště

Katedra informačních technologií

Elektronicky schváleno dne 7. 11. 2017

Ing. Jiří Vaněk, Ph.D.

Vedoucí katedry

Elektronicky schváleno dne 8. 11. 2017

Ing. Martin Pelikán, Ph.D.

Děkan

V Praze dne 12. 03. 2018

Čestné prohlášení

Prohlašuji, že svou diplomovou práci "Nasazení systému Zabbix" jsem vypracoval samostatně pod vedením vedoucího diplomové práce a s použitím odborné literatury a dalších informačních zdrojů, které jsou citovány v práci a uvedeny v seznamu použitých zdrojů na konci práce. Jako autor uvedené diplomové práce dále prohlašuji, že jsem v souvislosti s jejím vytvořením neporušil autorská práva třetích osob.

V Praze dne _____

Poděkování

Rád bych touto cestou poděkoval vedoucímu této práce Ing. Martinu Havránkovi, Ph.D. za odbornou pomoc, věnovaný čas, velmi cenné připomínky a vstřícný přístup při zpracování této diplomové práce.

Nasazení systému Zabbix

Abstrakt

Tato diplomová práce se zabývá problematikou monitoringu lokální počítačové sítě. Teoretická východiska analyzují rozdělení druhů monitoringu, monitorovacích protokolů, možností notifikace a specifikace požadavků na bezpečnost dohledových systémů. Jsou představeny open source monitorovací systémy a popsány vlastnosti, které poskytují.

V praktické části práce je provedena vícekriteriální analýza variant. Je zjištěn aktuální stav monitorovaného prostředí ve společnosti a na základě zjištěných skutečností je proveden návrh monitoringu. Jsou navrženy hardwarové požadavky na výkon serverů pro instalaci Zabbixu. Je provedena jeho následná instalace, konfigurace a uvedení do provozu. Jsou konfigurována sledovaná zařízení, nastaveny notifikace, vytvořeny uživatelské grafy, předvedeny výstupy dohledového systému a zabezpečení systému pomocí certifikátů. Na závěr je popsána aktualizace systému a diskuze dosažených výsledků.

Klíčová slova: Zabbix, monitoring, dohledový systém, nasazení, server, agent, Cacti, Nagios, protokol, SNMP

Deployment of Zabbix system

Abstract

This diploma thesis deals with the issue of the local computer network monitoring. The theoretical basis analyzes the distribution of monitoring types, monitoring protocols, notification options, and specification of safety requirements for monitoring systems. Open source monitoring systems are presented and the properties they provide are described.

In the practical part of the thesis, a multi-criteria analysis of variants is performed. The current status of the monitored environment in the company is analyzed and a monitoring proposal is made on the basis of the facts available. Furthermore, hardware requirements for server performance of Zabbix installation are designed. Its subsequent installation, configuration and commissioning are carried out. The monitored devices are configured, notices are set up, user graphs are created and the monitoring system outputs are displayed. The system is secured by certificates. Finally, the system update is described and discussion of the results is provided.

Keywords: Zabbix, monitoring, monitoring system, deployment, server, agent, Cacti, Nagios, protocol, SNMP

Obsah

Seznam obrázků

Seznam tabulek

1 Úvod

Monitoring funkčnosti informačních technologií je dnes již běžnou součástí fungování IT oddělení firem. Pokud se snažíme řídit investice do oddělení informačních technologií a sledovat práci, jsou výsledky monitoringu jedním ze základních parametrů, které by měly být kontrolovány.

Síťoví administrátoři čelí zajímavé výzvě. Na jednu stranu jsou počítačové sítě, jejich fyzické komponenty a protokoly všeobecně známé, zařízení jsou stále levnější a jejich nastavení jednodušší. Není potřeba shánět certifikovaného odborníka k instalaci a konfiguraci jednoduché sítě nebo k zapojení zařízení do sítě již existující. Samotný koncept síťování je rozšířený a dobře osvojený. Uživatelé a vývojáři dnes považují za samozřejmost počítačový systém alespoň částečně přístupný online.

Na druhou stranu jsou požadavky na stále jednodušší a přístupnější sítě velice obdobného charakteru a ve skutečnosti tlačí sítě k tomu, aby byly stále více komplexní. Počet připojených zařízení má tendenci konstantního růstu a současně roste i objem přenášených dat (streamy médií, aplikační data, zálohy, databázové dotazy a replikace), což má za následek zahlcování sítě a zároveň růst objemu dat na úložišti. Z hlediska kvality se taktéž vyskytují různorodé požadavky, jako jsou například požadavky na řízení různých fyzických médií (optický kabel, síťový kabel a bezdrátové připojení) či potřeba vysokého výkonu, bezpečnosti, spolehlivosti, dostupnosti a integrity dat.

Tyto dvě zmíněné tendence nutí správce sítí poskytovat: více služeb, větší dostupnost, vyšší výkon při současně nižším ekonomickém rozpočtu a nižší časové dotaci managementu v porovnání s novějšími technologiemi. V současné době je kladen důraz především na to, aby byl správce sítě schopen udržovat síť v dlouhodobě provozuschopném stavu a zároveň rychle identifikoval a vyřešil různorodé problémy. V ideálním případě se od správců sítě očekává iniciativa, aby byly problémy ošetřeny a v budoucnu bylo možné jim předejít. Současně je podstatná integrace systémů s různými nástroji a prostředím, které se nacházejí mimo přímý dosah správce sítě, např. databáze, účetnictví, systémy řízení incidentů a podobné. Dalším významným faktorem je také schopnost správce sítě vysvětlit a prezentovat nálezy a výsledky své práce jasným a srozumitelným způsobem i pro jednotlivce bez technického vzdělání. (Dalle Vacche, 2015, Dalle Vacche, 2013)

Systém Zabbix umožňuje nejen získat aktuální přehled o stavu celé sítě, ale také získat historické přehledy funkčnosti jednotlivých částí informačních technologií či připravit předpovědi pro budoucí vývoj. Jedná se hlavně o monitoring následujících technologií:

- servery (hardware i jednotlivé aplikace),
- firewally,
- aktivní síťové prvky,
- konektivita (internet, propojení poboček),
- periferie (stav tiskáren),
- teplotní a vlhkostní čidla,
- webové prezentace,
- platnost SSL certifikátů.

Zabbix je opensource distribuované monitorovací řešení, které svými vlastnostmi vyhovuje jak malým, tak velkým společnostem. Zabbix je postaven na systému šablon, které se ručně nebo automaticky přiřazují k jednotlivým monitorovaným zařízením. Do systému lze velmi snadno přidat i vlastní skripty pro monitorování speciálních zařízení, což umožňuje kontrolu prakticky všech zařízení v síti.

Správně implementovaný monitoring sítě umožní nejen získání přehledu o stavu sítě a včasné řešení případných problémů, ale především umožní značné omezení času vynaloženého pracovníky IT oddělení na běžnou údržbu provozu celého IT oddělení.

2 Cíl práce a metodika

2.1 Cíl práce

Cílem této práce je instalace a konfigurace open source monitorovacího systému Zabbix v nejmenované společnosti. V průběhu zpracování práce je přiblíženo monitorovací prostředí a jsou popsány jednotlivé prvky a funkce, které tento systém nabízí.
Dílčími cíli práce jsou:

- zjištění současného stavu monitoringu,
- stanovení požadavků na kritičnost prvků.

2.2 Metodika

Metodika teoretické části této diplomové práce se zakládá na studiu odborné literatury. V praktické části je pak založena na instalaci a konfiguraci aktuální verze monitorovacího systému Zabbix na virtuální server. Instalace a následná konfigurace je provedena na virtuálním serveru na platformě VMware, který je pro tento systém vyhrazen v nejmenované společnosti. Při instalaci je použit instalační balíček systému Zabbix z oficiálních webových stránek dohledového systému Zabbix. Jsou předvedeny jednotlivé kroky instalace a nakonec samotné spuštění programu.

3 Teoretická východiska

V současné době počítačových sítí jsou kladeny velké požadavky na dostupnost síťových zařízení a služeb s tím spojených. Jen málokteré odvětví se dokáže obejít bez internetu. V komerčním odvětví se každé zpoždění může projevit finanční ztrátou, a proto je důležité sledovat stav všech zařízení připojených do sítě. Zejména důležitý je dohled nad stavem serverů. Sledovat je možné sítě lokální i zapojené v internetu. Sledování serverů má za cíl poskytnout řešení problému v nejkratší možné době. Sledování síťových zařízení ale není jen o dostupnosti a funkci nabízených služeb, dovoluje plánování a budování síťové infrastruktury a zároveň ji pomáhá udržovat v žádoucím stavu. Po zpracování získaných informací lze projektovat rozvržení zařízení, upravovat výkon dle konkrétních potřeb a předcházet možným problémům. (Kolísek, 2013)

3.1 FCAPS

FCAPS je zkratka určená k členění pojmu správa sítě na další podskupiny této oblasti. Jednotlivá písmena akronymu znamenají:

- F = Fault (níže popsané jako správa chyb),
- C = Configuration (níže popsané jako správa a údržba konfigurací),
- A = Accounting (níže popsané jako správa účtování),
- P = Performance (níže popsané jako správa výkonu),
- S = Security (níže popsané jako správa bezpečnosti).

Níže je uveden popis jednotlivých částí:

- Správa chyb. Cílem správy chyb je záznam, detekce a řešení chybových stavů v síti. Hranice mezi správou chyb a správou výkonu je téměř neznatelná. Na správu chyb lze nahlížet jako na okamžité řešení přechodných problémů sítě (například výpadky spojení, serveru, routeru či softwaru), zatímco správa výkonu řeší dlouhodobější pohled na poskytování přijatelných stupňů výkonu v závislosti na různých požadavcích výkonu nebo příležitostných selhání sítě. Ve spravování chyb hraje důležitou roli protokol SNMP[1], který se uplatňuje i u správy výkonu.

[1] Simple Network Management Protocol

- Správa a údržba konfigurací. Umožňuje správci sítě sledovat zařízení na spravované síti a poskytuje mu informace o konfiguracích hardwaru a softwaru provozovaných na zařízení. Přehled správy konfigurací a požadavků pro sítě založené na IP[2] lze nalézt v RFC[3] 3139.
- Správa účtování. Umožňuje správci sítě určovat, zaznamenávat a řídit přístup uživatelů a zařízení k síťovým zdrojům. Využívá k tomu kvóty, zpoplatnění na základě využívání a také přidělení oprávnění k přístupu.
- Správa výkonu. Cílem správy výkonu je kvantifikovat, měřit, vykazovat, analyzovat a řídit výkon různých síťových komponent (například využití a propustnost). Tyto komponenty zahrnují jednotlivá zařízení (například router či server) nebo cesty skrz síť. Podstatným protokolem pro řízení výkonu je protokol SNMP specifikovaný v RFC 3410.
- Správa bezpečnosti. Cílem správy bezpečnosti je kontrolovat přístup k síťovým zdrojům v souladu s definovanými zásadami. Zabývá se také zabezpečením distribuce zpráv, autentizací, kryptografií a digitálními podpisy. Jednou ze stěžejních součástí jsou také firewally sloužící ke sledování a kontrole externích přístupových bodů do konkrétní sítě. (Kurose, 2013)

3.2 Dohledový systém

Pavel Bezpalec ve své publikaci Nové trendy v elektronických komunikacích, Management ICT[4] systémů definuje dohledový systém jako „prvek, který je implementován do monitorované sítě a který periodicky zjišťuje dostupnost a stav jednotlivých uzlů a spojů“. V okamžiku výskytu problému či zjištění nedostupnosti určitého prvku odesílá informaci administrátorovi sítě, který na základě toho provede odpovídající úkony. S využitím dohledového systému je také možné předem určit postupy, jež je možné vykonat v případě nastalých problémů, například odeslat informaci administrátorovi sítě a zároveň restartovat službu. Tímto způsobem lze síť v závislosti na typu nasazeného dohledového systému do jisté míry automatizovat. Dohledový systém lze

[2] Internet Protocol
[3] Request for Comments (tj. žádost o komentáře)
[4] Information and Communication Technologies (tj. Informační a komunikační technologie)

používat jako nainstalovanou aplikaci na serveru nebo jako samostatné jednoúčelové zařízení. (Bezpalec, 2016)

Dohledové systémy lze členit na:

- základní,
- pokročilé,
- proaktivní,
- sledující datový tok.

3.2.1 Základní dohledové systémy

Základní dohledové systémy obvykle pracují s protokolem ICMP[5]. Tyto systémy pravidelně za časový úsek zjišťují stav sledovaného prvku. Na základě zjištěných informací o době jeho odezvy a dostupnosti vyhodnotí sytém prvek jako dostupný či nedostupný. Monitorovací systém tohoto typu se používá u menších sítí typu LAN[6] nebo u sítí, ve kterých nám sledovaná zařízení neposkytují více informací než zmiňovanou dostupnost a odezvu. (Bezpalec, 2016)

3.2.2 Pokročilé dohledové systémy

Pokročilé dohledové systémy jsou vhodnější zejména pro větší sítě. Tyto systémy pracují s větším počtem protokolů, mezi které patří například SNMP, CDP[7] či SSH[8].

O zařízení připojeném v síti lze získat širokou škálu informací, jako například využití systémových prostředků, stav spuštěných služeb nebo aktuální tok dat. Data ze serverů lze získat za využití lokálně spuštěných agentů, kteří shromažďují informace o spuštěných službách a hardwarových komponentách, které nelze získat pomocí síťových protokolů. Tyto dohledové systémy mají více informací o provozu v síti, a tak mohou včas administrátory upozornit na neobvyklé situace. Často je administrátor informován před kolapsem sledovaného zařízení. Pokročilé dohledové systémy dokáží vyhodnotit a rozpoznat útok na námi sledovanou síť již v jeho počátku. Pokud se administrátor rozhodne nasadit tento typ dohledového systému, získá velké množství informací o stavu sítě (oproti základním dohledovým systémům). Zpravidla se pokročilé dohledové systémy

[5] Internet Control Message Protocol
[6] Local Area Network
[7] Cisco Discovery Protocol
[8] Secure Shell

využívají ve větších sítích LAN, nebo WAN[9] ve vlastnictví jednoho subjektu. (Bezpalec, 2016)

3.2.3 Proaktivní dohledové systémy

Proaktivní dohledové systémy jsou ve své podstatě pokročilé dohledové systémy, na rozdíl od kterých dokáží síťová zařízení také vzdáleně spravovat. Nejvíce se používají v prostředí, které je z velké části automatizované, zejména datacentra, rozsáhlé sítě nebo vysoce dostupné clustery. V souvislosti se současným rozvojem cloudových služeb dochází i k vývoji dohledových systémů. Tento trend přiměl výrobce dohledových systémů přizpůsobit se ke zvýšené poptávce po tomto typu služby. Majoritní podíl systémů dokáže implementovat předem vytvořené scénáře, které automaticky reagují na předem nastavené události. Tímto způsobem lze docílit velkého snížení nákladů určených na údržbu, spravování sítě a růst kvality nabízených služeb. Proaktivní dohledové systémy se využívají především u cloudových řešení, v datacentrech a rozsáhlých sítích. (Bezpalec, 2016)

3.2.4 Systémy sledující datový tok

Systémy sledující datový tok jsou systémy založené na sledování veškeré komunikace odehrávající se na síti. Jedná se o systémy velmi náročné na výkon, zároveň je pro jejich nasazení zapotřebí vlastnit nezbytné technické zázemí. Nelze provádět odposlouchávání jen na jednom centralizovaném místě, a proto se používají inteligentní přepínače, které zrcadlí tok dat na další port s připojenou jednotkou pro sběr dat, což je důvod, proč jsou tyto systémy tak náročné na zázemí a výkon. Existují i systémy používající zvláštní průchozí jednotky zapojené přímo na přenosové médium. Ty následně odešlou data do centralizovaného místa, kde je provedeno uložení, analýza a prezentace pro administrátora systému. Těchto systémů využívají společnosti především v případech, kdy se klade důraz na sledování veškerého toku informací v síti. (Bezpalec, 2016)

3.3 Metody monitorování

Metody monitorování lze rozdělit na ty, které využívají k monitoringu agenty, a ty, které nikoliv. Agent je ve své podstatě jednoduchý software, který je spuštěn na každém

[9] Wide Area Network

sledovaném zařízení. Monitorování s agentem a bez agenta má své pro a proti. Velmi málo dohledových systémů kombinuje současně oba tyto způsoby. Ve většině případů je upřednostňováno právě jedno řešení před druhým.

3.3.1 Metoda monitorování s agentem

Dohledové systémy založené na využití agentů mají tu výhodu, že nejsou závislé na připojení do sítě. Agent je spuštěn na sledovaném zařízení tak dlouho, dokud je samotné zařízení spuštěné, a lokálně ukládá informace o jeho běhu. Tímto způsobem pracuje do té doby, než data odešle na server. Systémy využívající agenta mají větší pravomoc, rychlý čas vyhodnocení a jsou schopny vykonávat určité úkony, které nelze provést mezi dvěma zařízeními skrz síťové připojení.

Slabou stránkou řešení založených na agentech je skutečnost, že musí být vynaloženo značné úsilí k nasazení, konfiguraci, údržbě a monitoringu stavu agenta samotného. V ojedinělých případech může dokonce docházet k tomu, že agent sám způsobuje problémy na zařízení, které monitoruje. (Adato, 2015)

3.3.2 Metoda monitorování bez agenta

Monitoring bez agenta je oproti metodě monitorování s agentem neškodný vůči sledovanému zařízení, zpravidla na něm tedy nemůže způsobovat poruchy. Jedinou možností, která by mohla mít nepředvídatelný negativní dopad na výkon zařízení, by byl spíše chybně napsaný skript než samotný monitorovací systém. Vzhledem k tomu, že není potřeba nasazovat žádné agenty, je monitoring čistě otázkou údržby centralizované databáze všech sledovaných zařízení, která se aktualizuje v pravidelných časových intervalech. Přidání dalších zařízení znamená pouze rozšíření databáze o další záznam.

Určitým nedostatkem je fakt, že monitoring bez agenta často není schopen dosáhnout takové granularity shromažďovaných dat a stejně tak ani robustnosti automatických odpovědí v případě výskytu problému, jako je tomu u monitoringu s agentem. Sběr dat na centralizovaný server také způsobuje zvýšení zátěže na celou monitorovanou infrastrukturu, což má za následek potřebu více než jednoho serveru tak, aby došlo k rovnoměrnější distribuci zátěže.

Dohledový systém s agentem či bez agenta mají vždy svá specifická uplatnění. Nelze ani o jednom z nich říci, že jeden je užitečnější než druhý, neboť vždy záleží na konkrétních požadavcích kladených na monitorované prostředí. (Adato, 2015)

3.3.3 Aktivní monitoring

Monitorovací systém se aktivně stará o získávání hodnot z monitorovaného zařízení. Funguje na principu rozesílání paketů do sítě a jejich následného přijetí na místě jiném. Tímto způsobem lze měřit například ztrátovost paketů, maximální propustnost nebo zpoždění při prostupnosti sítí. Nevýhodou je zvýšená zátěž sítě a možné ovlivnění uživatelů. Nepatrnou nevýhodou je také to, že měříme vždy jen naše testovací pakety a ne přímo pakety od uživatelů, které mohou být od testovacích paketů velmi odlišné. (Ubik, 2006)

3.3.4 Pasivní monitoring

Při pasivním monitoringu zasílá samo monitorované zařízení požadované informace o stavu sítě, žádné pakety se do sítě neposílají, tudíž nehrozí zvýšená zátěž sítě ani ovlivnění uživatelského provozu. Vyhodnocují se časové a objemové vlastnosti uživatelského provozu. Pasivní monitoring může sledovat charakteristiky, které aktivním monitoringem nelze získat. Mezi takové charakteristiky patří například objem a dynamika volné kapacity v síti, zda v síti dochází k bezpečnostním útokům nebo které aplikace uživatelů kladou největší nároky na kapacitu sítě. (Ubik, 2006)

3.4 Používané protokoly

Protokoly jsou důležité pro dorozumívání jednotlivých prvků sítě mezi sebou. Různá zařízení mohou pro komunikaci využívat různé protokoly. Protokol jako takový definuje formát a pořadí zpráv vyměňovaných mezi komunikujícími entitami a taktéž definuje, co se má stát při přijetí či přenosu zprávy nebo výskytu události.

3.4.1 TCP

Protokol TCP[10] používá většina běžných síťových služeb. Tento protokol je založen na spolehlivosti přenosu, která je zajištěna kontrolou každého odeslaného paketu a následným zjištěním, zda byl paket doručen či nikoliv. Spojení je navázáno na základě znalosti IP adresy koncového zařízení a čísla portu. Využívá k tomu příznaky (CWR,

[10] Transmission Control Protocol

ECE, URG, ACK, PSH, RST, SYN, FIN) v hlavičce TCP paketu. K navázání komunikace skrze TCP se obvykle používá takzvaný Three-Way Handshake (volně přeložené jako třícestné potřesení rukou), který probíhá podle následujících kroků:

- Odesílající strana odešle paket s příznakem SYN a uvedeným číslem sekvence *x*.
- Přijímající strana potvrdí zprávu odesláním příznaku SYN-ACK. Zároveň si uloží číslo sekvence *x,* jako číslo sekvence nastaví svojí hodnotu *y* a do čísla odpovědi nastaví další očekávanou hodnotu *x+1*.
- Odesílající strana pošle zprávu s příznakem ACK a tím indikuje, že bylo navázáno spojení s přijímající stranou. Číslo sekvence je nastaveno na *x+1* a číslo odpovědi je *y+1*.

Spojení je definováno na základě čtyř parametrů: IP adresy a portu odesílající strany a IP adresy a portu strany přijímající. Může být vytvořena i jednostranná komunikace, v takovém případě se použije vždy pouze jedna dvojice parametrů: IP adresa a port odesílající strany nebo IP adresa a port stany přijímající. (Sosinsky, 2016, Kurose, 2013)

V dohledových systémech se využívá TCP protokol například k poslouchání na určitém portu a testování, zda se podaří na sledovaném portu navázat TCP spojení. Dalším možným využitím může být:

- Sledování, zda je možné navázat spojení na určený port.
- Testování, zda je na určeném portu aktivní služba a je schopna přijímat TCP připojení.
- Sledování výkonu služby v podobě sledování doby odezvy z určeného portu. (Zabbix SIA, 2017)

3.4.2 SNMP

Protokol SNMP je asynchronní, transakčně orientovaný protokol založený na modelu klient-server komunikující skrze TCP/IP sítě. Může pracovat i nad protokoly OSI CLNS[11], AppleTalk DDP či Novel IP, avšak nejrozšířenější je použití právě nad TCP/IP sítěmi. Tento protokol je jednoduchý, značně rozšířený a užitečný k získávání nebo nastavování hodnot na určitém zařízení. SNMP podporuje velký počet zařízení, jako jsou

[11] Open Systems Interconnection Connectionless Network Service

například: switche, routery, UPS[12], počítačová čidla, tiskárny, přístupové body, modemy, osobní počítače a servery. K přenosu dat slouží protokol UDP[13], jehož výhodou je rychlost, na druhou stranu však oproti TCP neposkytuje ověření doručení. Informace lze v případě potřeby získat jednorázově nebo v pravidelných intervalech a ukládat je pak do databáze. Pro komunikaci jsou vždy zapotřebí právě dvě komunikující strany. Strana, která požadavky odesílá, se nazývá SNMP klient. Zpráva je vytvořena s dynamicky zvoleným portem, poté je zapouzdřena do UDP datagramu a je odeslána na port 161. Na straně zařízení je SNMP agent odpovídající na požadavky. Agent odpovídá z portu 161 na dynamicky zvolený port klientem. Výjimkou je takzvaná SNMP trap (v překladu past na událost), která komunikuje přes port 162. SNMP trap se vysílá pouze při výskytu určitých událostí (nevzniká na základě požadavku klienta) například výpadek proudu, překročení maximální teploty, výskyt nového zařízení a podobně.

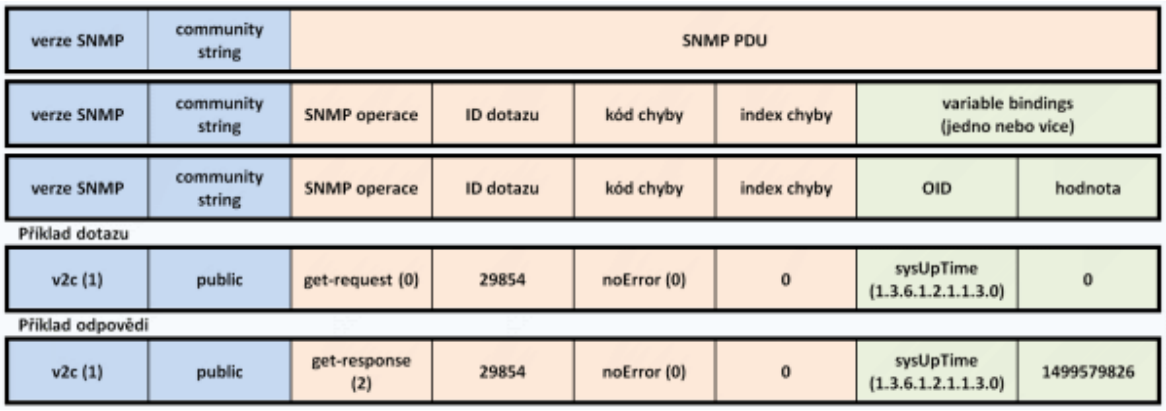
Existují 3 verze SNMP:

- **SNMPv1** používá pro autentizaci textový řetězec community string (pro uživatele s povolením nastavování hodnot se používá write community string, pro povolení pouze čtení se používá read community string). Tato prvotní verze obsahuje řadu nedostatků a v dnešní době se především z bezpečnostních důvodů nedoporučuje její použití.
- **SNMPv2c** již zahrnuje kontrolu doručení, stále však využívá community string.
- **SNMPv3** používá jméno a heslo pro autentizaci, šifrování CBC-DES a CFB-AES-128 pro komunikaci, kontrolní součty MD5[14] a SHA[15] pro ochranu proti změně, řízení přístupu k jednotlivým objektům (View Access Control Model) a možnost přenosu přes SSH (Secure Shell) nebo TLS (Transport Layer Security)/DTLS (Datagram Transport Layer Security).

Pro ukládání informací používá tento protokol takzvanou MIB databázi (Management information base). Soubory uložené v MIB jsou stromově strukturované pomocí jedinečného číselného identifikátoru OID (Object identifier). Například hodnota 1.3.6.1.2.1.2.2.1.6.1 je popsána textovou podobou iso.org.dod.internet.mgmt.mib-2.interfaces.ifTable.ifEntry.ifPhysAddress. (Sosinsky, 2016, Bouška, 2006, Kolísek, 2013)

[12] Uninterruptible Power Supply
[13] User Datagram Protocol
[14] Message-Digest
[15] Secure Hash Algorithm

Na obrázku níže je znázorněno složení UDP datagramu, na kterém lze vidět hlavičku složenou z verze SNMP a community stringu. V části SNMP PDU (Protocol Data Unit) jsou uložena uživatelská data, která jsou složena z dalších částí. SNMP operace určuje, jaký úkon se má provést. ID dotazu slouží ke spojení dotazu s odpovědí. Kód chyby obsahuje chybový kód pouze za předpokladu, že k nějaké chybě došlo. Index chyby označuje místo, kde se chyba vyskytla. Variable bindings je složeno vždy z OID a hodnoty. Může zde být jedna nebo více svázaných hodnot v případě operace s využitím více OID. (Bouška, 2009)



Obrázek 1 - Formát datagramu SNMP (Zdroj: samuraj-cz.com, 2009)

SNMP používá tyto operace:

- **Get** pro získání jedné nebo více instancí objektu. Pokud u SNMPv1 dojde k chybě u jedné hodnoty, operace Get nemá žádnou návratovou hodnotu. U SNMPv2 je v případě chyby u jedné hodnoty nastavena návratová hodnota operace Get jako vše, co se podařilo načíst bezchybně.
- **GetNext** vrátí instanci dalšího objektu v seznamu nebo tabulce.
- **Set** nastaví hodnotu instanci objektu.
- **Trap** asynchronně informuje o události.
- **GetBulk** nově od SNMPv2, vrací větší blok dat (jako několik řádků z tabulky).
- **Inform** nově od SNMPv2, slouží pro komunikaci mezi dvěma správci.

3.4.3 ICMP

Protokol ICMP (Internet Control Message Protocol) je definován v RFC 792. Je součástí IP protokolu, což znamená, že každé zařízení v síti s implementovaným IP protokolem zároveň podporuje i ICMP protokol. Může tak přijímat, odesílat i zpracovávat

ICMP zprávy. Hlavním úkolem protokolu je informovat o výskytu chyb při přenosu IP datagramů. Chyba může vzniknout hned z několika příčin, například:

- Došlo k překročení TTL (Time To Live).
- Směrovač nemá dostatečný buffer pro přeposlání datagramu.
- Směrovač musí fragmentovat datagram, ale ten má nastaven příznak *Don´t Fragment*.
- Směrovač nebo uzel zjistí chybu v syntaxi IP hlavičky.
- Směrovači chybí v tabulce záznam o cílové síti.

Existují dva typy ICMP protokolů: ICMPv4 a ICMPv6, v závislosti na použití protokolu IPv4 či IPv6. Hlavička ICMP protokolu je umístěna hned za hlavičkou IPv4 nebo IPv6 a je identifikována jako číslo 1. Komplexní protokol obsahuje tři pole:

- Hlavní typ, který identifikuje ICMP zprávu.
- Rozšiřující informace o typu pole.
- Kontrolní součet, který pomáhá detekovat chyby vzniklé při přenosu.

Další tři pole, která následují, obsahují ICMP data a původní IP hlavičku ke zjištění, který paket ve skutečnosti chyboval.

ICMP protokol byl také využíván k takzvaným DoS (Denial of Service) útokům. Útok spočíval v odeslání IP paketu většího, než povoloval počet bytů IP protokolu, což mělo za následek nedostupnost služby. (Odvárka, 2001, Postel, 1981)

3.4.4 IPMI

IPMI (Intelligent Platform Management Interface) je standard umožňující vzdálený monitoring a správu serverů z hlediska hardware. Vše je řízeno BMC (Baseboard Management Controller), který je propojen sériovou sběrnicí s ostatním hardware. Je tak možné sledovat základní parametry hardware jako jsou například teplota, otáčky ventilátorů nebo hodnoty napájení. Lze i nainstalovat, odinstalovat, aktualizovat software či spravovat připojené hardware periferie. Tento protokol je k dispozici u většiny serverů od velkých výrobců (Dell, HP, Lenovo, Oracle, IBM, Fujitsu). BMC má tu výhodu, že se jedná o samostatný hardware. Dokáže tak pracovat bez ohledu na to, zda je server v činnosti či nikoliv. (Kolísek, 2013)

IPMI protokol obsahoval mnoho bezpečnostních děr, které se v pozdějších verzích protokolu podařilo opravit. Za zmínku stojí ukládání hesel BMC v nešifrované podobě,

přednastavené výchozí jméno a heslo (například jméno: admin, heslo: admin). Získání přístupu k BMC představuje závažný problém, neboť BMC má přímý přístup k základní desce i hostujícímu systému, kde by útočník mohl napáchat mnoho škod. Doporučuje se také vymezit pro BMC oddělenou síť LAN nebo VLAN[16] pro komunikaci. (Moore, 2013)

3.4.5 Syslog

Syslog je protokol pro přeposílání zpráv z logů po síti pracující na bázi klient-server. Logy z mnoha zařízení se soustředí na jednom místě a jsou následně využity k analýzám, ladění systému či podpoře rozhodování ICT systémového managementu. Ke sběru záznamů z logu slouží klient aplikace, která je poté odesílá na syslog server. Takových zpráv je zpravidla velké množství (mnohdy stovky za minutu), z tohoto důvodu se pro zpracování vytváří skripty, které analyzují logy a upozorňují na problémy. Zprávy se posílají v otevřeném textu přes protokoly UDP nebo TCP. Pro šifrovanou komunikaci lze využít takzvaný SSL wrapper, který zajistí šifrovací vrstvu SSL/TLS. Komunikace probíhá skrze port 514. (Gerhards, 2009)

3.4.6 Další protokoly

Výše zmíněné protokoly jsou ty, které se používají nejčastěji a jsou podporovány největším počtem dohledových systémů a zařízení v síti. Existuje mnoho dalších protokolů a služeb, které lze k monitoringu sítě využít. Jedná se například o JMX[17], NETCONF[18], CMIP[19], HTTP[20] a další. Jednou z možností monitoringu sítě je vytvoření vlastního komunikačního protokolu, pomocí kterého bude prováděn dohled nad sítí. Tato možnost je ale ze všech možností nejméně používaná a vyskytuje se jen zřídka.

3.5 Notifikace

Notifikace je zpráva o nově vzniklém problému nebo události. Zpráva obvykle obsahuje informace o problému a zařízení, na kterém se problém projevil. Nejběžnějším

[16] Virtual Local Area Network
[17] Java Management Extensions
[18] Network Configuration Protocol
[19] Common Management Information Protocol
[20] Hypertext Transfer Protocol

způsobem doručení notifikací je e-mail nebo zpráva SMS. Dohledové systémy zobrazují notifikace také přímo v aplikaci pro správu monitoringu. Velmi často se jedná o webovou stránku, přes kterou se provádí veškerá správa systému. Některé dohledové systémy podporují i zasílání upozornění na různé instant messengery, pagery, RSS[21] kanály nebo tvorbu log souborů.

3.6 Výstupy monitoringu

Forma výstupu dohledového systému je důležitým prvkem při rozhodování, jaký konkrétní monitorovací systém zvolit. Výstup musí být v první řadě přehledný a užitečný. Pokud systém dokáže získávat velké množství informací, ale nedokáže je následně přehledně prezentovat, nepřinese to žádný užitek, a takový systém je tudíž neúčelný.
Možné výstupy monitorovacích systémů:

- grafy,
- tabulky,
- okamžité číselné nebo procentuální hodnoty,
- reporty.

3.7 Bezpečnost dohledových systémů

Bezpečnost dohledových systémů je velmi důležitým prvkem. Dohledový systém ukládá informace o provozu a zařízeních v síti a tato data je nutné důkladně zabezpečit. Dohledový systém zpravidla shromažďuje a ukládá data na jednom místě v databázi. Přístup ke všem těmto informacím je vhodné zabezpečit na takové úrovni, aby k datům měli zřízený přístup pouze autorizovaní uživatelé.

3.7.1 Autentizace a autorizace

Systém by měl umožňovat nastavení různých uživatelských rolí. Na nižším stupni řízení by uživateli měla být nastavena možnost pouze prohlížení dat, zatímco na vyšším stupni by měl mít oprávnění i měnit nastavení systému. Automatické odhlašování při nečinnosti uživatele také určitou mírou přispívá ke zvýšení zabezpečení a doporučuje se jej

[21] Rich Site Summary

aplikovat na každém uživatelském účtu. Další formou zabezpečení je například přihlašování do systému s využitím dvoufaktorové autentizace nebo povolení přístupu do systému jen z předem definované IP adresy. Dohledové systémy ve většině případů dovolují přihlášení do systému pomocí LDAP[22] serveru, HTTP webového serveru nebo přes interní přihlašovací systém, kde se o správu uživatelů stará přímo dohledový systém.

3.7.2 Šifrovaná komunikace

Při sběru dat by se mělo využívat šifrovaných komunikačních protokolů. Příkladem může být využití protokolu SNMPv3 namísto protokolu SNMPv2, neboť druhý zmíněný protokol šifrovanou komunikaci nepodporuje. Šifrování komunikace je vhodné aplikovat na veškerou komunikaci probíhající po síti. Při používání dohledového systému skrze webové stránky se doporučuje využívat připojení pomocí protokolu HTTPS[23] namísto nešifrovaného HTTP.

3.7.3 Oddělení dohledového systému

Dohledový systém a jeho komponenty by měly být odděleny od stávající infrastruktury. Dohledový systém by měl mít například vyhrazenou svoji zabezpečenou VLAN síť, přes kterou bude spravován. Dále pak oddělit server, databázi a frontend (uživatelská část) dohledového systému takovým způsobem, aby všechny tyto komponenty nebyly provozovány na jednom hardwarovém stroji. Toto opatření má výhodu především ve snazší škálovatelnosti při pozdějším nárůstu objemu dat a přibývajícími nároky na výkon dohledového systému. Takto rozdělená architektura je sice složitější na prvotní instalaci tří serverů namísto jednoho, ale v budoucnu je mnohem lépe rozšiřitelná. V případě poškození jednoho ze serverů je pak také jednodušší obnova provozu jedné komponenty systému, než kdybychom přišli o celý dohledový systém.

3.7.4 Zálohování a obnovení

Nejdůležitější částí dohledových systémů k zálohování je databáze a konfigurační soubory serveru. Doporučuje se provádět zálohu databáze každý den, a to ideálně v době,

[22] Lightweight Directory Access Protocol
[23] Hypertext Transfer Protocol Secure

kdy je databáze nejméně používána. Toto časté zálohování je vhodné především pro menší firmy s nižším počtem sledovaných zařízení. Pro firmy s vyšším počtem sledovaných zařízení je takto četné zálohování nevhodné a je vhodnější provést replikaci databáze nebo výpis.

Agenti pro server a proxy nejsou považovány za důležité pro zálohování, neboť je lze jednoduše obnovit z dostupných balíčků či je znovu kompilovat. Stejný princip lze aplikovat i na konfigurační soubory frontendu, které jsou snadno obnovitelné z oficiálních zdrojů výrobců systému.

3.7.5 Aktualizace

Udržovat dohledový systém aktualizovaný je velmi důležité. Vývojáři systémů průběžně vydávají aktualizace a opravné balíčky chyb, které se vyskytují v průběhu času. V praxi se lze setkat s tím, že od prvotní instalace systémů nedochází k pravidelné údržbě. V důsledku zanedbání aktualizací pak dochází k vyšší tendenci zranitelnosti dohledového systému vůči útokům. Doporučení o udržování aktualizovaného software se netýká pouze dohledových systémů, týká se veškerého software napříč celým IT odvětvím.

3.7.6 Bezpečnostní hrozby dohledových systémů

Na webové stránce https://www.cvedetails.com lze dohledat zjištěné bezpečnostní hrozby různých výrobců software a hardware. Jednotlivé hrozby jsou na webové stránce publikovány současně s uvedením jejich závažnosti a dalších informací. Lze se dozvědět, zda byl problém vyřešen, kdo problém objevil a co může daný problém způsobit. Například záznam s označením CVE-2016-10134 představuje hrozbu napadení systému Zabbix tzv. SQL[24] injekcí. Této zranitelnosti umožňoval Zabbix využít útočníkům ve verzích systému zveřejněných před verzí 3.0.4. Útočník mohl provádět libovolné SQL příkazy změnou pole *toggle_ids* v souboru *latest.php*. Dále se lze dočíst, že na problém upozornil Alexander Vladishev dne 22.7.2016. Tato chyba dohledového systému byla opravena a v dalších verzích se již nevyskytovala. (Vladishev, 2017)

[24] Structured Query Language

3.7.7 Best practices dohledových systémů

Takzvané best practices neboli osvědčené postupy se dají aplikovat i na dohledové systémy. Existuje několik základních doporučení, která by se měla v dohledových systémech aplikovat, a přispět tak k zabezpečení a bezproblémovému provozu. Best practices jsou shrnuty v následujících bodech:

1) Používat k monitoringu oddělený server. Chránit tento server jako jednu z nejdůležitějších částí celé sítě, nejlépe tak, aby byl server chráněn firewallem, nespouštět na něm žádné nepotřebné služby a uzavřít všechny nevyužívané porty.
2) Nepoužívat dohledový systém pod administrátorským účtem, pokud není vyloženě potřeba provedení takových změn v nastavení, které nelze provést bez administrátorských práv. K prohlížení a dohledu tedy vždy využívat jen uživatelský účet s omezenými pravomocemi.
3) Zabezpečit komunikační kanály. Komunikace by měla být zašifrována, tak aby útočník nemohl odposlouchávat informace přenášené po síti. K šifrování lze využít komunikační protokoly podporující šifrování. Lze využít i bezpečnostní certifikáty vydávané certifikační autoritou (např. PKI – public key infrastructure) nebo šifrovat takzvaným pre-shared key (PSK). Útočníkovi tak bude zamezeno číst zprávy zasílané skrz síť, měnit zprávy nebo se vydávat za skutečného agenta.
4) Zabezpečit přístup k vzdáleným agentům. Přístup by měl být zabezpečen například firewally nebo access listy.
5) Sledovat jen ty informace, na základě kterých lze učinit rozhodnutí nebo započít určitou akci. Mezi takové informace lze zařadit například: chyby při čtení z pevného disku, příliš vysoká teplota CPU, nedostatek místa na disku, nadměrné využití swapovacího oddílu, zařízení nereagující na ICMP ping. Naopak nelze nijak reagovat na následující události: nedostupnost veřejné Google DNS, změněný stav portu switche, nižší propustnost přístupového bodu, ztráta jednoho datového paketu, služba odpovídající zpravidla jednou nebo dvakrát pomaleji než obvykle. (Nagios Enterprises, 2018, Zabbix SIA, 2018, Moerch, 2016)

3.8 Open source monitorovací systémy

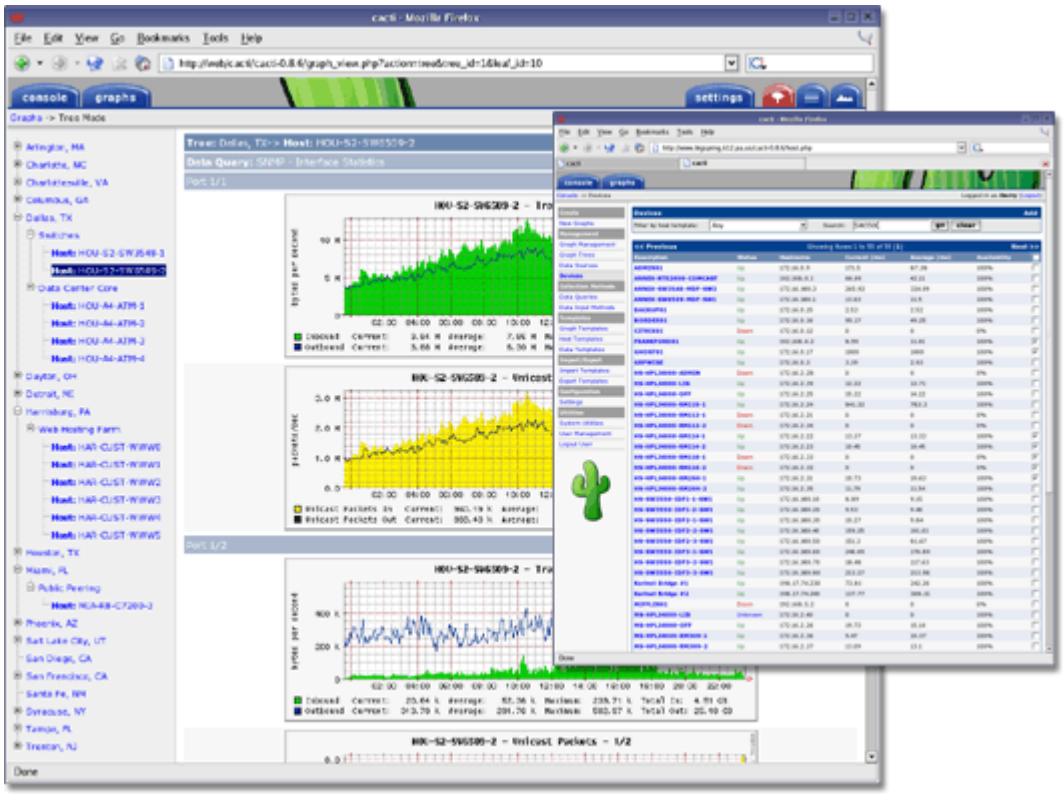
V následujícím textu teoretické části práce jsou představeny jedny z nejpoužívanějších open source monitorovacích systémů. Všechny zmíněné monitorovací systémy jsou dostupné pod volně šiřitelnou licencí GNU GPL[25]. Jedinou výjimkou je OpenNMS, který je šířen pod licencí AGPLv3[26]. General Public Licence umožňuje úpravu kódu a jeho následné šíření. Níže zmíněné dohledové systémy jsou tedy poskytovány zdarma i se svými zdrojovými kódy. Některé společnosti starající se o vývoj těchto systémů pak nabízí možnost placené podpory. Může se jednat o placené konzultace, instalaci systému nebo jeho údržbu.

3.8.1 Cacti

Cacti je open source systém pro tvorbu velkého množství grafů ze získaných dat. Jeho vývoj začal v roce 2001, kdy Ian Berry pracoval na programu, který by poskytl větší flexibilitu než doposud používaný program k vytváření grafů MRTG (Multi Router Traffic Grapher).

Tento systém je v podstatě frontend pro RRDTool (Round Robin Database Tool). RRDTool je open source nástroj, který využívá takzvanou round robin databázi (databáze s konstantní velikostí po celou dobu existence). Cacti získaná data z RRDTool ukládá do MySQL databáze. Je zde schopen ukládat grafy, zdroje dat a round robin archivy. Frontend je napsán v jazyce PHP[27]. Pro implementaci systému je dále nutný net-SNMP a určitý webový server, například Apache. Data lze získávat buď s využitím SNMP a nebo skriptem. Pro použití periodického sběrače je nutný takzvaný cron (proces který automaticky spustí skript/příkaz v určitý systémový čas). Data se ve výchozím nastavení sbírají v pětiminutových intervalech, lze ale nastavit i jiný interval (nejkratší možný je 60 vteřin).

[25] General Public Licence
[26] Affero General Public Licence
[27] Hypertext Preprocessor – programovací jazyk



Obrázek 2 - Prostředí Cacti (Zdroj: cacti.net)

Nejdůležitější funkce Cacti:

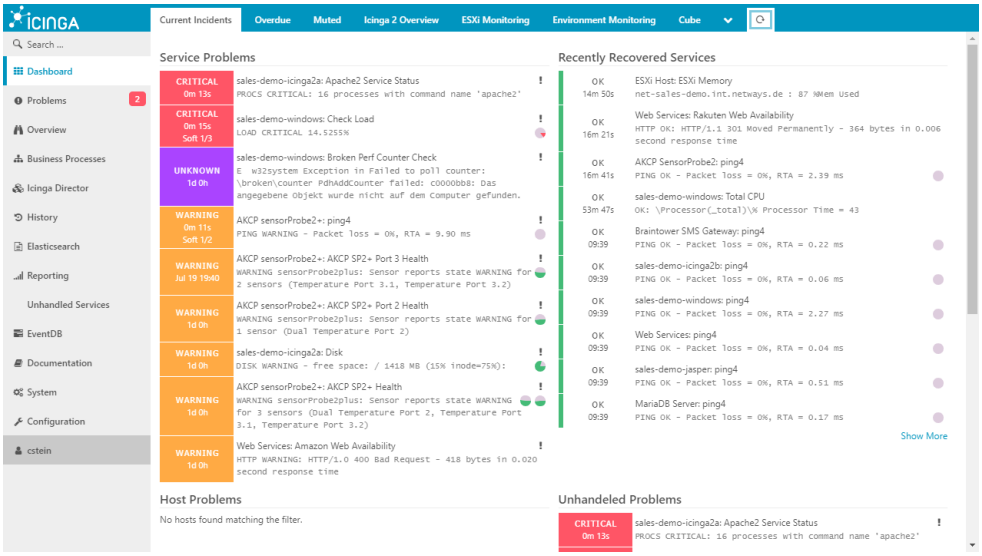
- Podpora velkého množství pluginů.
- Na problémy upozorňuje e-mailem.
- Podporuje RRD soubory s více než jedním zdrojem dat a může také použít RRD soubor uložený lokálně v souborovém systému.
- Stromová hierarchie grafů, které lze sdružovat do přehledů za delší období.
- Tvorba šablon grafů či celých zařízení.
- Skripty pro shromažďování vlastních dat.
- Šablony lze importovat/exportovat ve formátu xml.
- Nastavení uživatelských práv k přístupu jen k určitým grafům.

(The Cacti Group, Inc., 2017)

3.8.2 Icinga 2

Icinga 2 je open source dohledový systém, který kontroluje dostupnost síťových zdrojů, upozorňuje uživatele na výpadky a generuje údaje o výkonu pro reporting. Je škálovatelný, rozšiřitelný pomocí pluginů a schopný monitorovat rozlehlá a komplexní

prostředí. Je publikovaný pod GNU licencí verze 2. Původně vznikl jako oddělení se od Nagiosu v roce 2009. Ve verzi 2 byl však zdrojový kód od počátku přepsán. Přesto byla zachována podpora všech pluginů, které jsou dostupné pro Nagios.



Obrázek 3 - Prostředí Icinga2 (Zdroj: icinga.com)

Hlavní přednosti systému Icinga 2 jsou:

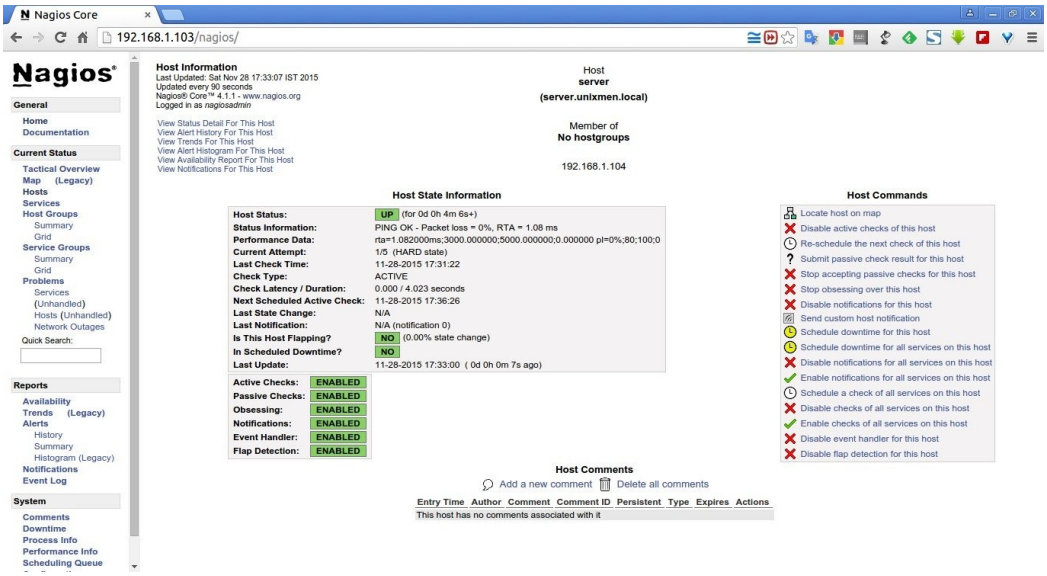
- Monitoring síťových služeb, serverových komponent a služeb spuštěných na serverech.
- Provádí monitoring pluginy Icinga 2.
- Podporuje obsluhu událostí a notifikace.
- Podpora mnoha operačních systémů.
- Paralelizované servisní kontroly.
- Výběr ze dvou uživatelských prostředí (Classic UI a Icinga web).
- Reporty tvořené na základě šablon.

Zákazníci Icinga: Puppet Labs, Audi, Spacex, Debian, Snapdeal, McGill, RIPE Network Coordination Centre, M[a]ller, Adobe, Siemens, Deloitte. (Icinga, 2018)

3.8.3 Nagios

Nagios je open source dohledový systém využívaný velkými organizacemi. Jedná se patrně o jeden z nejznámějších a nejpoužívanějších open source dohledových systémů, jehož vývoj začal již v roce 1996.

Nagios monitoruje problémy v síti, jako jsou například přetížené datové linky, síťové připojení, routery, switche a další sledovaná zařízení. Dokáže sledovat dostupnost, dobu provozu a odezvu každého uzlu v síti. Získané informace lze interpretovat mnoha vizuálními prezentacemi nebo pomocí reportů. Tento dohledový systém je schopen sledovat servery s využitím agenta i bez agenta. Pro Nagios existuje více jak 5000 doplňků, což poskytuje velmi dobrý prostor k získání maximálního počtu informací ze sledovaného zařízení. Nagios poskytuje sledování webových aplikací, služeb či procesů a je schopen sledovat aplikace operačních systémů Windows, Linux, Unix. (Nagios Enterprises, LLC, 2017)



Obrázek 4 - Prostředí Nagios (Zdroj: unixmen.com)

Hlavní výhody Nagiosu jsou:

- Centralizovaný pohled na celou monitorovanou IT strukturu.
- Automatické restartování aplikací v případě výskytu problému.
- Přístup pro více uživatelů v jeden okamžik.
- Selektivní přístup uživatelům pouze ke komponentám infrastruktury, ke kterým mají přidělena práva.
- Aktivní komunita s více než 1 milionem uživatelů.
- Rozšiřitelná architektura.

Zákazníci Nagios: AOL, DHL, McAfee, MCI, MTV, Yahoo!, Universal, Toshiba, Sony, Siemens, and JPMorgan Chase.

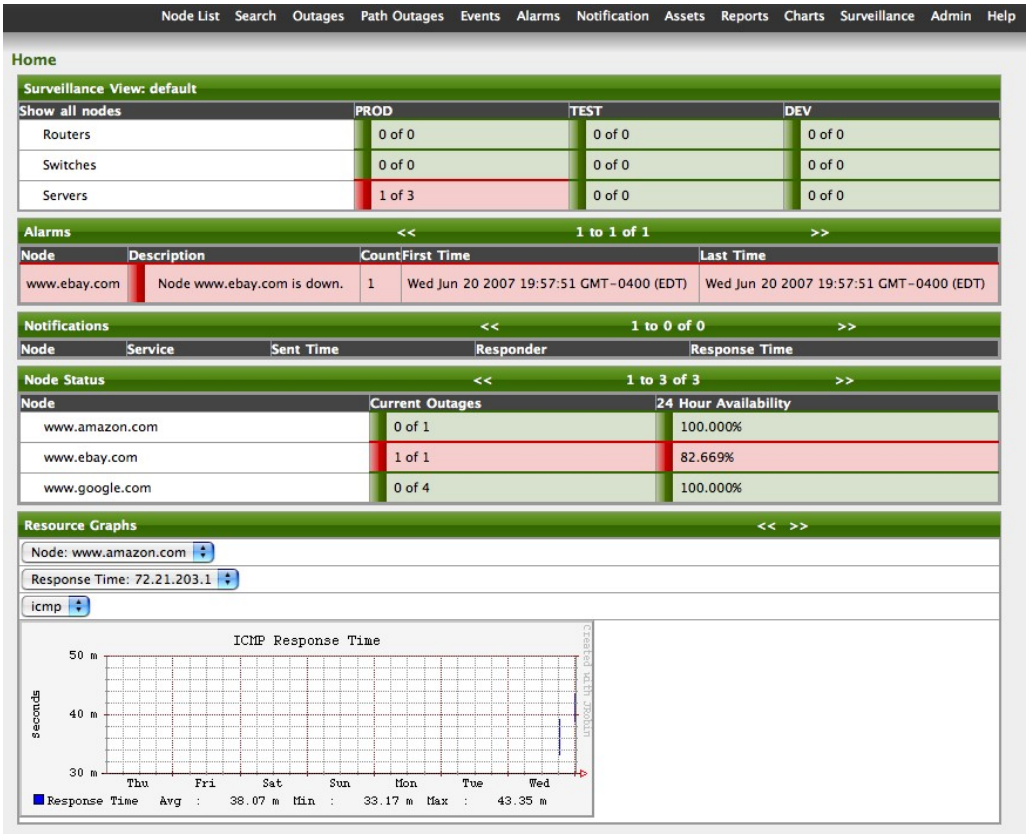
3.8.4 OpenNMS

OpenNMS je platforma s otevřeným zdrojovým kódem určená pro vytváření síťových monitorovacích řešení. Existují dvě distribuce OpenNMS: Meridian a Horizon. Použití distribuce Meridian je vhodné pro podniky a firmy, které hledají stabilitu a dlouhodobou podporu. Horizon je vhodný pro firmy, které jsou charakteristické častou implementací inovací a jsou ideální pro sledování nových technologií a ekosystémů IT. Obě verze jsou šířeny pod volně dostupnou licencí AGPLv3. Sběr dat je možný přes SNMP, JMX, WMI, NRPE[28], NSClient++ a XMP[29]. Sběr dat z aplikací probíhá přes HTTP, JDBC[30], XML nebo JSON[31]. OpenNMS je postaven na architektuře řízené událostmi.

Událost je vytvořena vždy při výpadku služby, uzlu, rozhraní, nebo když jsou překročeny limitní hodnoty. SNMP trapy a zprávy syslog jsou normalizovány do událostí a mohou být propojeny tak, aby vytvořily pracovní postupy upozornění.

Díky podporované architektuře REST API je snazší integrovat OpenNMS do již existující infrastruktury. Zjišťování topologie síťové vrstvy 2 je založeno na informacích SNMP a standardů jako jsou například LLDP[32], CDP a Bridge-MIB. Podporováno je také zjišťování topologie směrování vrstvy 3 na základě OSPF a IS-IS. (THE OPENNMS GROUP, 2017)

[28] Nagios Remote Plugin Executor
[29] Extensible Metadata Platform
[30] Java Database Connectivity – API pro programování v jazyce Java
[31] JavaScript Object Notation
[32] Link Layer Discovery protocol



Obrázek 5 - Prostředí OpenNMS (Zdroj: wiki.opennms.org)

Hlavní výhody systému OpenNMS jsou:

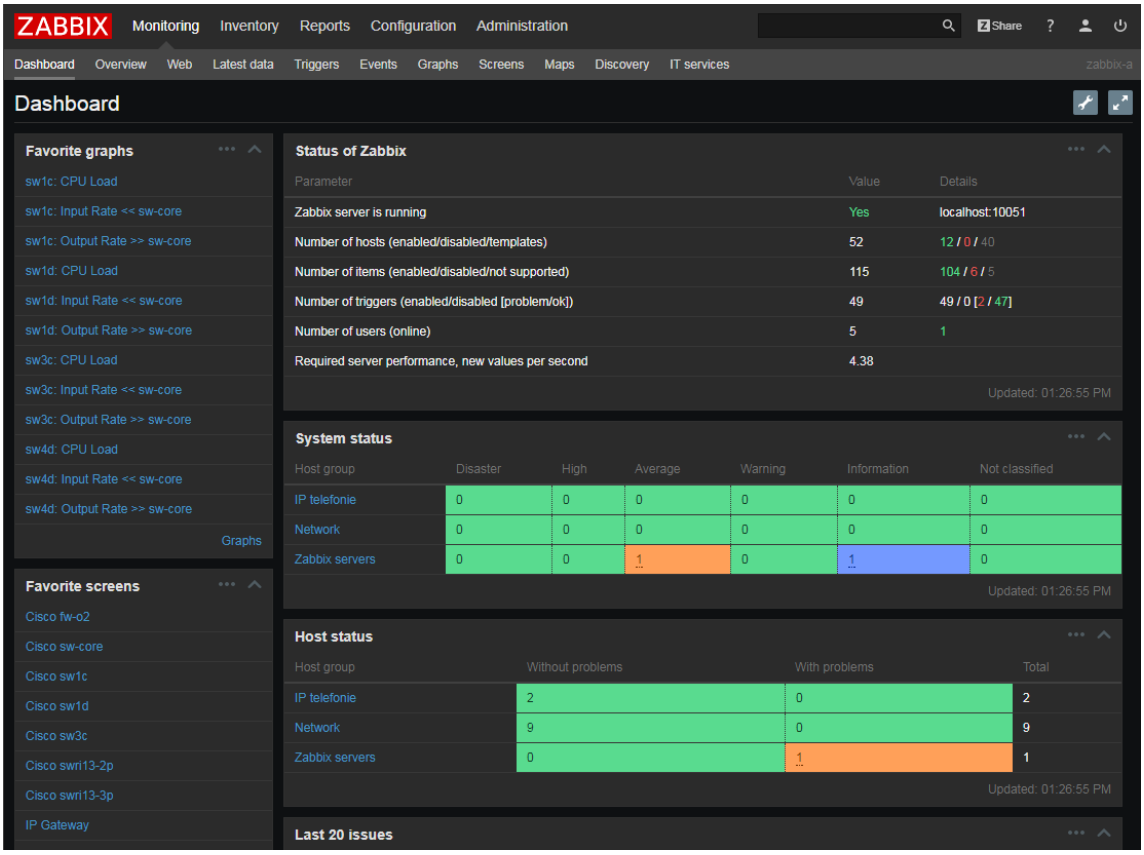
- Původně vyvinuto pro Linux, ale jsou podporovány i Windows, Solaris a OSX.
- Sledování teploty zařízení.
- Upravitelná administrátorská stránka.
- Monitoring napětí zařízení.
- Podpora IPv4 a IPv6.
- Události mohou zasílat upozornění přes e-mail, SMS, XMPP[33].
- Podporuje geografické zobrazení uzlů v mapách Open Street Map, Google Maps nebo Mapquest.

3.8.5 Zabbix

Zabbix je open source dohledový systém napsaný v jazyce PHP. Je zaměřený na monitoring dostupnosti a výkonu IT infrastruktury. Vyvinula ho společnost Zabbix LLC

[33] Extensible Messaging and Presence Protocol

v roce 2005 a šíří ho pod GPLv2 licencí. Je to multiplatformní produkt (OpenBSD, FreeBSD, Linux, Solaris / Illumos / OmniOS, macOS, Windows). Je schopen sledovat servery, virtuální stroje, síťové zařízení a zároveň všechny získané informace ukládat nebo je v reálném čase vykreslovat do grafů, přehledů nebo map. Zabbix je škálovatelný i pro velmi rozlehlá prostředí. Je zde možnost distribuovaného monitoringu skrze proxy, správy přes webové prostředí, bezpečné autentizace uživatelů a správy uživatelských rolí. Podporuje takzvaný polling i trapping skrze monitoring s agentem. Podporována je také možnost bez agenta. Dále je dostupný monitoring internetových stránek nebo například virtuálních strojů VMware. Automaticky dokáže objevovat servery a zařízení v síti, stejně tak automaticky přidělovat výkonnostní testy a testy dostupnosti k objeveným entitám. Zabbix se skládá ze čtyř základních stavebních bloků: server, webové prostředí, agenti a proxy. Notifikace mohou být doručovány e-mailem, SMS, instant messengery nebo zobrazeny na webové stránce systému. (Zabbix LLC, 2017, Uyttergieven, 2015)



Obrázek 6 - Prostředí Zabbix (Zdroj: Autor)

Hlavní přednosti Zabbixu jsou:

- Open source řešení včetně potřebných komponent (Linux, Apache, MySQL/PostgreSQL, PHP).
- Snadná instalace a správa.
- Vysoce účinní agenti pro Unix a Windows (x32, x64, Itanium) platformy poskytují široké možnosti monitoringu.
- Centralizovaný dohledový systém ukládá všechny informace (konfigurace a výkonnostní záznamy) v relační databázi pro jednodušší zpracování a opětovné využití dat.
- Přímý monitoring Java aplikačních serverů přes JMX.
- Frontend má vlastní ochranu proti útokům hrubou silou.
- Automatizace lze provádět pomocí skriptů v různých jazycích například Ruby, Python, Perl, PHP, Java nebo shell skripty.
- Podpora MySQL, PostgreSQL, Oracle, IBM DB2 a SQLite databází.
- Integrace nástrojů pro správu systému jako jsou Puppet, cfengine, Chef a bcfg2.
- Velké množství vizualizací dat.
- Volitelnost časového intervalu sběru dat.
- Možnost nastavení frekvence sběru dat podle pracovní doby, dnů v týdnu.
- Vestavěné postupy k údržbě dobře organizovaných dat.
- Zabezpečení komunikace mezi agentem a serverem PKI (public key infrastracture nebo PSK (pre-shared key).
- Kvalitní a početná komunita uživatelů.
- Zabbix Share – webová stránka k sdílení šablon od uživatelů.
- Oddělený aplikační kód od kódu grafického uživatelského prostředí.
- Automatické hledání zařízení v síti.
- Nastavení prototypu (šablony) pro určitý typ zařízení, který se poté automaticky nastaví na ostatní zařízení stejného typu. (Knight, 2017)

Zákazníci Zabbixu: DELL, Orange, ICANN, Salesforce, Bodybuilding.com, LLC., ITtelligent Consulting Services, Eltele AS, Total Server Solutions, LLC., ChinaNetCloud

4 Vlastní práce

V této části práce je nejprve provedeno srovnání vybraných open source nástrojů k monitoringu sítě. V následující kapitole je poté zdůvodněn výběr systému Zabbix s ohledem na požadavky nejmenované společnosti, kde byla vypracována praktická část práce. Stěžejní částí vlastní práce bylo samotné nasazení dohledového systému Zabbix, jeho správná konfigurace, použití šablon, triggerů, nasazení agentů ke sledování komponent v rozhlehlé infrastruktuře společnosti a v neposlední řadě také vyhotovení reportů a přehledů ze získaných dat.

4.1 Porovnání vybraných open source systémů

Pro základní srovnání vybraných open source dohledových systémů byla po konzultaci se zaměstnanci nejmenované firmy odpovědných za síťovou infrastrukturu vybrána následující kritéria:

- IP SLA[34] reporty,
- automatické zjišťování síťových zařízení,
- SNMP,
- Syslog,
- VMware,
- databáze,
- podnikové zaměření dohledového systému,
- správa skrze webové rozhraní.

4.1.1 Srovnávací kritéria

V následujících podkapitolách jsou popsána jednotlivá srovnávací kritéria.

4.1.1.1 IP SLA

V nejmenované společnosti je využíváno několik druhů zařízení s podporou IP Service Level Agreement. Pro společnost je důležité, aby skrze dohledový systém bylo

[34] Internet Protocol Service Level Agreement

možné monitorovat IP SLA na těchto zařízeních, a případně ze zjištěných informací tvořit reporty či zasílat upozornění odpovědným osobám.

4.1.1.2 Automatické zjišťování zařízení

Toto kritérium je důležité z hlediska častého připojování nových zařízení do stávající sítě. Jejich automatické zjišťování tak mnohonásobně usnadní práci s dohledovým systémem. Ve spojení s použitím přednastavených šablon pro různé druhy zařízení se jedná o propracovaný způsob, jak v několika krocích přidat nové sledované zařízení.

4.1.1.3 SNMP

K získávání dat ze sledovaných zařízení bude ve velké míře využíván tento komunikační protokol. Pro výsledky hodnocení je tedy důležitá podpora protokolu a jednoduchost jeho implementace do dohledového systému.

4.1.1.4 Syslog

V nejmenované společnosti je v provozu značné množství serverů sloužících k různým účelům. Pro společnost bude výhodné mít možnost monitorovat jednotlivé logy z těchto serverů. Dohledový systém by měl toto umožňovat co možná nejjednodušším způsobem ať už z hlediska implementace či nároků na provoz.

4.1.1.5 VMware

V nejmenované společnosti je podpora monitoringu VMware důležitým kritériem především z hlediska využívání několika VMware clusterů. Přestože samotný VMware má v sobě implementovaný určitý dohledový systém, bude pro společnost výhodné mít tyto VMware clustery pod dohledem na centralizovaném místě v podobě dohledového systému.

4.1.1.6 Databáze

U tohoto kritéria bude zvýhodněno takové řešení, které poskytuje větší počet podporovaných databází. V nejmenované společnosti se používá několik typů databází a je kladen důraz na to, aby dohledový systém podporoval co možná nejvyšší počet typů databází, a v případě potřeby bylo možné migrovat databázi na některý z těchto typů.

4.1.1.7 Podnikové zaměření

Nejmenovaná společnost, ve které probíhá zpracování praktické části této práce, disponuje několika regionálními pobočkami s celkovým počtem 1500 zaměstnanců, a tudíž je kritérium podnikového zaměření dohledového systému klíčové. Dohledové systémy, které jsou již od počátku svého vývoje zaměřené na velké podniky, budou v celkovém hodnocení upřednostněny.

4.1.1.8 Správa skrze webové rozhraní

Z hlediska vyššího komfortu a možnosti snadného přístupu do dohledového systému skrze webové rozhraní je toto hodnotící kritérium pro výsledné rozhodování důležité. Webové rozhraní poskytuje uživatelům možnost přistupovat z jakéhokoliv zařízení (s přístupem na internet) do dohledového systému, měnit jeho nastavení či prohlížet získaná data.

4.1.2 Data pro srovnání dohledových systémů

Data pro srovnání jednotlivých dohledových systémů byla získána z volně dostupných dokumentací nebo z oficiálních internetových stránek srovnávaných dohledových systémů.

Číslo	Název / Kritéria	Cacti	Icinga	Nagios	OpenNMS	Zabbix
1	**IP SLA**	ANO	Plugin	Plugin	ANO	ANO
2	**Automatické zjišťování zařízení**	ANO	Plugin	Plugin	ANO	ANO
3	**SNMP**	ANO	Plugin	Plugin	ANO	ANO
4	**Syslog**	ANO	Plugin	Plugin	ANO	ANO
5	**VMware**	Šablona	Plugin	Plugin	ANO (s úpravou ve VMware)	ANO (autodis-covery)
6	**Databáze**	RRDtool, MySQL	MySQL, Oracle, PostgreSQL	Flat file, SQL (přes ndoutils), MySQL	JRobin / RRDTool / Apache Cassandra, PostgreSQL	Oracle, MySQL, PostgreSQL, DB2, SQLite
7	**Podnikové zaměření**	NE	NE	ANO	ANO	ANO
8	**Webové rozhraní**	ANO	ANO	ANO	ANO	ANO

Tabulka 1 - Kritéria pro srovnání dohledových systémů (Zdroj: Autor)

4.1.3 Ohodnocení variant

V této kapitole je přiděleno bodové ohodnocení jednotlivým variantám v závislosti na předem stanovených kritériích. Mezi vybranými srovnávacími kritérii nejsou žádná kvantitativní kritéria. Všechna kritéria jsou kvalitativního charakteru.

Ohodnocení kvalitativních kritérií proběhlo na základě vlastních zkušeností, nabytých vědomostí při zpracovávání literární rešerše a konzultace s odpovědnými zaměstnanci společnosti, ve které je praktická část práce vypracována.

Všechna kritéria jsou bodově ohodnocena na stupnici bodů od 1 do 10, přičemž hodnota 10 znázorňuje nejvyšší možné hodnocení, hodnota 1 je nejnižší bodový zisk. Ohodnocení kritérií u jednotlivých variant je uvedeno v následující tabulce.

			Kritérium							
			1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
Varianta	**1**	**Cacti**	10	10	10	10	7	3	1	10
	2	**Icinga**	7	7	7	7	6	6	1	10
	3	**Nagios**	7	7	7	7	6	4	10	10
	4	**OpenNMS**	10	10	10	10	8	7	10	10
	5	**Zabbix**	10	10	10	10	10	9	10	10

Tabulka 2 - Ohodnocení variant podle jednotlivých kritérií (Zdroj: Autor)

4.1.4 Váhy kritérií

V této kapitole je stanoveno, která kritéria mají největší vliv na volbu dohledového systému. Ohodnocení jednotlivých kritérií proběhlo Saatyho metodou. Váha každého kritéria je stanovena jako geometrický průměr řádků Saatyho matice, která je zobrazena v Tabulce 3 pod tímto textem. Následně jsou tyto váhy použity ve srovnání variant metodou AHP.

		Kritérium								Váha kritéria	Normalizovaná váha
		1	**2**	**3**	**4**	**5**	**6**	**7**	**8**		
Kritérium	**1**	1	1/5	1/6	2	1/3	1/3	1/2	3	0,679	0,046
	2	5	1	1/2	3	2	1/3	1/2	3	2,460	0,165
	3	6	2	1	5	1/2	2	1	3	2,318	0,156
	4	1/2	1/3	1/5	1	1/7	1/5	1/5	1/2	1,000	0,067
	5	3	½	2	7	1	3	5	3	2,938	0,197
	6	3	3	1/2	5	1/3	1	3	1	2,265	0,152
	7	2	2	1	5	1/5	1/3	1	3	1,979	0,133
	8	1/3	1/3	1/3	2	1/3	1	1/3	1	1,260	0,085
									Součet	14,899	1

Tabulka 3 - Váhy vybraných kritérií (Zdroj: Autor)

V nejmenované společnosti, kde probíhalo vypracování praktické části práce, byl kladen důraz na to, aby vybraný systém disponoval monitoringem VMware, a aby implementace byla co nejjednodušší. Dalšími preferovanými požadavky byla možnost plné kontroly přes webové rozhraní, jednoduchá implementace monitoringu skrze SNMP, jednoduchá implementace funkce automatického zjišťování nově připojených zařízení a podpora Oracle databáze. V případě, že se provoz dohledového systému ve společnosti osvědčí, uvažuje se o přechodu právě na Oracle databázi, neboť je ve společnosti již ve velké míře používána.

4.1.5 Srovnání metodou AHP

V této kapitole je provedena analýza metodou AHP a její následné vyhodnocení. Pro každé kritérium je vytvořena individuální tabulka, ve které jsou mezi sebou za pomoci Saatyho metody porovnány jednotlivé varianty. Jednotlivé preference a dispreference v každé tabulce jsou určeny na základě bodů z ohodnocení variant (viz Tabulka 2).

IP SLA		Varianta 1	2	3	4	5	Váha varianty	Normalizovaná váha
Varianta	1	1	4	4	1	1	1,741	0,286
	2	¼	1	1	1/4	1/4	0,435	0,071
	3	¼	1	1	1/4	1/4	0,435	0,071
	4	1	4	4	1	1	1,741	0,286
	5	1	4	4	1	1	1,741	0,286
						Součet	6,093	1,000

Tabulka 4 - Rozdělení váhy IP SLA mezi varianty (Zdroj: Autor)

Rozdělení váhy pro kritéria: automatické zjišťování zařízení, SNMP a Syslog vyšlo se stejnými výsledky, jako tomu je u kritéria IP SLA. Je to způsobeno shodným bodovým ohodnocením variant (viz Tabulka 2). Proto je v této části práce uvedena pouze tabulka s vyhodnocením kritéria IP SLA, aby nedocházelo ke zbytečné duplicitě tabulek se shodnými výsledky. Dále jsou uvedeny tabulky pro jednotlivá kritéria.

VMware		Varianta 1	2	3	4	5	Váha varianty	Normalizovaná váha
Varianta	1	1	2	2	1/2	1/4	0,871	0,135
	2	1/2	1	1	1/3	1/5	0,506	0,079
	3	1/2	1	1	1/3	1/5	0,506	0,079
	4	2	3	3	1	1/3	1,431	0,222
	5	4	5	5	3	1	3,129	0,486
						Součet	6,443	1,000

Tabulka 5 - Rozdělení váhy VMware mezi varianty (Zdroj: Autor)

Databáze		Varianta					Váha varianty	Normalizovaná váha
		1	2	3	4	5		
Varianta	1	1	1/4	1/2	1/5	1/7	0,324	0,046
	2	4	1	3	1/2	1/4	1,084	0,154
	3	2	1/3	1	1/4	1/6	0,488	0,069
	4	5	2	4	1	1/3	1,679	0,238
	5	7	4	6	3	1	3,471	0,493
						Součet	7,046	1,000

Tabulka 6 - Rozdělení váhy Databáze mezi varianty (Zdroj: Autor)

Podnikové zaměření		Varianta					Váha varianty	Normalizovaná váha
		1	2	3	4	5		
Varianta	1	1	1	1/9	1/9	1/9	0,268	0,035
	2	1	1	1/9	1/9	1/9	0,268	0,035
	3	9	9	1	1	1	2,408	0,310
	4	9	9	1	1	1	2,408	0,310
	5	9	9	1	1	1	2,408	0,310
						Součet	7,760	1,000

Tabulka 7 - Rozdělení váhy Podnikové zaměření mezi varianty (Zdroj: Autor)

Webové rozhraní		Varianta					Váha varianty	Normalizovaná váha
		1	2	3	4	5		
Varianta	1	1	1	1	1	1	1	0,2
	2	1	1	1	1	1	1	0,2
	3	1	1	1	1	1	1	0,2
	4	1	1	1	1	1	1	0,2
	5	1	1	1	1	1	1	0,2
						Součet	5	1,000

Tabulka 8 - Rozdělení váhy Webové rozhraní mezi varianty (Zdroj: Autor)

V Tabulce 9 jsou obsaženy výsledné hodnoty pro jednotlivé varianty a ohodnocení pro jejich jednotlivá kritéria.

Varianta	Kritérium								Výsledný součet
	1	2	3	4	5	6	7	8	
1 Cacti	0,286	0,286	0,286	0,286	0,135	0,046	0,035	0,2	0,179
2 Icinga	0,071	0,071	0,071	0,071	0,079	0,154	0,035	0,2	0,091
3 Nagios	0,071	0,071	0,071	0,071	0,079	0,069	0,310	0,2	0,115
4 OpenNMS	0,286	0,286	0,286	0,286	0,222	0,238	0,310	0,2	0,262
5 Zabbix	0,286	0,286	0,286	0,286	0,486	0,493	0,310	0,2	0,353
Váhy kritérií	0,046	0,165	0,156	0,067	0,197	0,152	0,133	0,085	

Tabulka 9 - Vyhodnocení variant metodou AHP (Zdroj: Autor)

Varianta	Výsledný součet	Výsledné pořadí
1 Cacti	0,179	3.
2 Icinga	0,091	5.
3 Nagios	0,115	4.
4 OpenNMS	0,262	2.
5 Zabbix	0,353	1.

Tabulka 10 - Výsledné pořadí variant (Zdroj: Autor)

Jak lze vyčíst z Tabulky 10, nejlepšího výsledku dosáhl dohledový systém Zabbix, a stal se tak nejlepší kompromisní variantou podle vyhodnocení metodou AHP. Na druhém místě se umístil dohledový systém OpenNMS, na třetím místě systém Cacti, na čtvrtém místě systém Nagios a na posledním pátém místě se umístil systém Icinga.

4.2 Zdůvodnění výběru Zabbixu

Vzhledem k tomu, že veškeré srovnávané systémy poskytují do značné míry obdobnou funkčnost, hraje při výběru dohledového systému určitou roli také například velikost uživatelské základny, přívětivé uživatelské prostředí či předchozí zkušenosti s některým ze zmíněných dohledových systémů.

Dohledový systém Zabbix byl vybrán na základě srovnání metodou AHP. Pro preferenční porovnání kritérií a jednotlivých variant bylo využito Saatyho metody. Kritéria byla vybrána především na základě požadavků na dohledový systém od společnosti, kde je vypracována praktická část této práce.

4.3 Aktuální stav monitorovaného prostředí

Pro sestavení přehledu o aktuálním stavu monitorovaného prostředí ve společnosti jsou zařízení rozdělena do tří skupin: fyzická zařízení, virtuální zařízení a aplikace. Další skupinou pro monitoring je plnění SLA aplikacemi a časový sběr provozních parametrů. Pro zjištění aktuálního stavu prostředí byl společností poskytnut dokument o těchto skupinách a prvcích v nich zahrnutých, na jehož základě je vypracován plán pro monitoring dohledovým systémem Zabbix. Níže jsou podrobněji rozepsány jednotlivé skupiny. Požadavky na dostupnost jednotlivých prvků jsou znázorněny v Tabulce 11.

4.3.1 Fyzická zařízení

Do skupiny fyzických zařízení spadají především servery, síťová infrastruktura, disková pole, knihovny, tiskárny, LAN switche, WiFi AP, VPN koncentrátory, firewall a BGP routery.

Servery:

- Linux, Windows, AIX, Solaris a VMware.

Síťová infrastruktura:

- Switche, routery, firewall, LAN, WAN, WiFi AP, VPN koncentrátory a BGP routery.

Disková pole:

- V7000, V5010, Spectrum Scale, DS5100, Synology, SAN – Brocade 6520 a BNA.

Knihovny:

- TS4500, SL150 a TS3100.

Další fyzická zařízení:

- Tiskárny Konica Minolta.

4.3.2 Virtuální zařízení

Ve společnosti jsou používány dvě virtualizační platformy:

- VMware virtuální servery.
- AIX LPAR virtuální servery AIS, USYS, PROVYS a SAP.

4.3.3 Aplikace

Ve společnosti jsou používány následující aplikace:

- Databáze: Oracle, MS-SQL, LDAP a Fast.
- Infrastrukturní aplikace: ISE, Cisco Prime, TSM, Doména, tisk Konica Minolta, DNS, DHCP, Upload, ServiceDesk, AIS, USYS, SAP, iNEWS, DALET, SELECTOR, RSCR, Intranet, Logging, IP telefonie a Kontaktní centrum.

4.3.4 Plnění SLA aplikacemi

Mezi sledované plnění Service-level agreement aplikacemi patří:

- Odezva vybraných systémů, odezva na tisk do Konica Minolta, odezva internet a WAN.

4.3.5 Časový sběr pracovních parametrů

Sledované prvky:

- Uplink porty páteřního switche, WAN, internet, zaplnění diskových prostorů, zaplnění pamětí a využití CPU.

4.3.6 Požadavky na dostupnost prvku

V tabulce zobrazené níže je uveden výčet prvků užívaných ve společnosti. Kritičnost prvků je obodována a je uveden současný stav jejich monitoringu. Body jsou udělovány podle důležitosti, kde 1 bod představuje nejnižší kritičnost a 10 bodů nejvyšší kritičnost. Zjištění současného stavu monitoringu a následné obodování bylo stanoveno na základě konzultací s odpovědnými zaměstnanci z oddělení infrastruktury IT ve společnosti a správci jednotlivých systémů.

Prvek	Kritičnost	Současný stav monitoringu
WAN	10	ISP – Nagios
WiFi AP	5	Cisco Prime
Routery	10	ISP – Nagios
VPN koncentrátory	8	ISP – Nagios
Firewall	10	ISP – Nagios
VMware	10	vSOM
IBM servery (AIX LPAR)	10	XORUX LPAR2RRD, e-mail
DNS	10	ISP – Nagios, vSOM
DHCP	10	-
Switche	8-10 podle použití	Cisco Prime
SAN – Brocade	10	-
V7000	10	XORUX STOR2RRD, e-mail
V5010	9	XORUX STOR2RRD, e-mail
TS4500	7	e-mail
TS3100	5	e-mail
Spectrum Scale	9	-
Synology	3	e-mail
Oracle	10	Skripty, e-mail
MS-SQL	10	-
LDAP	10	-
ISE	10	e-mail
Kontaktní centrum	9	-
IP telefonie	10	-
Intranet	8	-

RSCR	10	-
DALET	10	vSOM
iNEWS	10	e-mail
SAP	9	-
USYS	9	-
Logging	10	-
AIS	9	-
ServiceDesk	7	vSOM
Upload FTP	8	-
Tisk Konica Minolta	9	SafeQ, e-mail
Doména	10	e-mail
TSM backup	7	e-mail
Cisco Prime	1	e-mail
Microsoft Exchange	10	e-mail, vSOM

Tabulka 11 - Současný stav monitoringu (Zdroj: Autor)

4.4 Návrh monitorování

Z hlediska důležitosti jednotlivých částí infrastruktury ve společnosti je monitoring sítě zaměřen především na sledování dostupnosti důležitých switchů, VMware a IBM serverů. Postupně jsou do dohledového systému přidávány další části infrastruktury podle společností stanovených priorit a potřeb.

4.4.1 SNMP

Monitoring přes SNMP je možný pouze přes SNMPv2 a je zabezpečen s využitím community stringu. U switchů bude sledováno vytížení procesoru, datový tok a teploty jednotlivých modulů. U firewallu bude sledován datový tok. Dále budou SNMP protokolem monitorovány vybrané servery, u kterých bude kontrolováno především jejich vytížení. Jedná se například o IBM servery, jejichž funkčnost a dostupnost je pro společnost podstatná. Tímto protokolem bude také možné sledovat brány používané pro IP telefonii. Do budoucna lze očekávat další rozšiřování počtu zařízení, která budou tímto protokolem sledována, neboť je u většiny používaných zařízení protokol podporován.

4.4.2 ICMP

Tímto protokolem je u všech sledovaných zařízení zajištěno sledování jejich dostupnosti. Funguje na principu zasílání dotazů na sledované zařízení.

4.4.3 Linux agent

Ve společnosti se vyskytují Linux servery, které jsou monitorovány systémem vSOM. Z tohoto důvodu není v době vypracovávání této práce požadován jejich další monitoring. Monitorovány budou pouze dva linuxové servery s instalací dohledového systému Zabbix.

4.4.4 VMware

Přes takzvanou funkci autodiscovery umožňuje Zabbix automaticky vyhledat hypervizory a hostitelské servery vytvořené ve virtuálním prostředí. Ke sledování je nutné poskytnout Zabbixu username, heslo a URL adresu, kde se nachází SDK VMware. Ve společnosti je používán VMware cluster, který bude touto funkcí sledován.

4.5 Návrh Zabbix instalace

V následujících podkapitolách jsou popsány požadavky na instalaci dohledového systému Zabbix. Je zde znázorněn výpočet potřebného úložného prostoru pro server s databází a server se systémem Zabbix a frontendem. Také je navržen potřebný výpočetní výkon obou serverů.

4.5.1 Požadavky k instalaci

Zabbix je možno provozovat na následujících platformách:

- Linux,
- IBM AIX,
- FreeBSD,
- NetBSD,
- OpenBSD,
- HP-UX,
- Mac OS X,
- Solaris,
- Windows od verze XP (pouze zabbix agent).

Zabbix podporuje následující databáze:

- MySQL 5.0.3 nebo novější. Pokud je pro Zabbix použita MySQL databáze je vyžadován InnoDB[35] engine.
- Oracle 10g nebo novější.
- PostgreSQL 8.1 nebo novější. Je doporučeno použít alespoň verzi PostgreSQL 8.3, která dosahuje lepšího výkonu při spouštění příkazu VACUUM.
- IBM DB2 9.7 nebo novější. (IBM DB2 je zatím funkční pouze experimentálně a vývojáři Zabbix jí nedoporučují k nasazení do ostrého provozu)
- SQLite 3.3.5 nebo novější. SQLite je podporována k využití pouze pro Zabbix proxy.

Zabbix frontend ke svému běhu vyžaduje Apache ve verzi 1.3.12 nebo novější a PHP 5.4.0 nebo novější. K PHP je ještě potřeba doinstalovat rozšíření popsaná v následující tabulce.

Název	Verze	Popis
gd	2.0 nebo novější	PHP GD musí podporovat PNG obrázky (--with-png-dir), JPEG obrázky (--with-jpeg-dir) a FreeType 2 (--with-freetype-dir)
bcmath		php-bcmath (*--enable-bcmath*)
ctype		php-ctype (*--enable-ctype*)
libXML	2.6.15 nebo novější	php-xml nebo php5-dom, pokud je poskytnut jako oddělený balík distributorem
xmlreader		php-xmlreader, pokud je poskytnut jako oddělený balík distributorem
xmlwriter		php-xmlwriter, pokud je poskytnut jako oddělený balík distributorem
session		php-session, pokud je poskytnut jako oddělený balík distributorem
sockets		php-net-socket (*--enable-sockets*). Vyžadován pro podporu uživatelských skriptů.
mbstring		php-mbstring (*--enable-mbstring*)
gettext		php-gettext (*--with-gettext*)
ldap		php-ldap. Vyžadováno pouze při LDAP autentizaci ve

[35] formát úložiště dat

		frontendu.
ibm_db2		Vyžadováno pouze pokud je IBM DB2 použito jako Zabbix databáze.
mysql		Vyžadováno pouze pokud je MySQL použito jako Zabbix databáze.
oci8		Vyžadováno pouze pokud je Oracle použito jako Zabbix databáze.
pgsql		Vyžadováno pouze pokud je PostgreSQL použito jako Zabbix databáze.
sqlite3		Vyžadováno pouze pokud je SQLite použito jako Zabbix databáze.

Tabulka 12 - Potřebná PHP rozšíření (Zdroj: Dokumentace Zabbix)

Požadavky pro provoz Zabbix serveru jsou sepsány v následující tabulce.

Název	Popis
OpenIPMI	Vyžadováno pro podporu IPMI protokolu.
libevent	Vyžadováno pro monitoring IPMI protokolem. Verze 1.4 nebo novější.
libssh2	Vyžadováno pro podporu SSH protokolu. Verze 1.0 nebo novější.
fping	Vyžadováno pro funkčnost ICMP pingu.
libcurl	Vyžadováno pro web monitoring, VMware monitoring, SMTP autentizaci a ElasticSearch. Pro SMTP autentizaci je vyžadována verze 7.20.0 nebo novější.
libiksemel	Vyžadováno pro podporu Jabber.
libxml2	Vyžadováno pro VMware monitoring.
net-snmp	Vyžadováno pro podporu SNMP protokolu.
libpcre3	Knihovna PCRE je vyžadována pro podporu regulárních výrazů PCRE.

Tabulka 13 – Požadavky pro funkčnost Zabbix serveru (Zdroj: Dokumentace Zabbix)

4.5.2 Databáze

Velikost	Platform	CPU/Paměť	Databáze	Počet sledovaných zařízení
Malá	CentOS	Virtualizované prostředí	MySQL InnoDB	100
Střední	CentOS	2 CPU jádra/2GB	MySQL InnoDB	500
Velká	RedHat Enterprise Linux	4 CPU jádra/8GB	RAID10 MySQL InnoDB nebo PostgreSQL	>1000
Rozsáhlá	RedHat Enterprise Linux	8 CPU jádra/16GB	RAID10 MySQL InnoDB nebo PostgreSQL	>10000

Tabulka 14 - Hardwarové požadavky databáze (Zdroj: Dokumentace Zabbix)

Databáze MySQL svým výkonem umožňuje sledování do přibližně 500 položek. Ke sledování sítě v nejmenované společnosti bude takový výkon výhledově dostatečný.

Pro výpočet potřebné velikosti databáze lze uvést následující příklad: Stanovme si potřebu sledovat 3000 položek. Od každé položky bude každých 60 vteřin získána hodnota. Z toho plyne, že každou vteřinu je potřeba uložit do databáze 50 hodnot (3000 / 60 = 50). Pro každou hodnotu je potřeba kromě dat ukládat také index sledované položky. V závislosti na tom, jak dlouho potřebujeme uchovávat historii všech těchto dat, lze spočítat přibližnou velikost databáze. Velikost databáze závisí také na zvoleném typu databáze a na formátu jednotlivých ukládaných položek. Jedna hodnota může nabývat zpravidla velikosti od 40 bajtů až do několika stovek bajtů. Jedná se například o datové typy integer, float, double, decimal, date, varchar, char a podobné. Ve sledované síti se bude historie uchovávat 3 měsíce (90 dní). Budeme-li získávat 50 hodnot za vteřinu, za 90 dní nasbíráme 388,8 milionů hodnot ((90 · 24 · 3600) · 50 = 388 800 000). Numerické datové typy zaberou přibližně 90 bajtů. Za takového předpokladu by všechny naše sledované položky za 3 měsíce zabraly 34,992 gigabajtů (388,8 · 90 = 34,992).

Podobným způsobem můžeme získat i hodnoty pro výpočet velikosti dat potřebných pro určení objemu dat trendů a pro ukládání dat o vzniklých událostech. Podrobnější popis jednotlivých výpočtů je uveden v tabulce na následující straně.

Parametr	Vztah pro výpočet potřebného místa na disku v bajtech
Konfigurační soubory	Konfigurační soubory mají pevnou velikost. Dosahují velikosti maximálně 10MB.
Historické data	*dny · (položky / frekvence sběru) · 24 · 3600 · bajty* dny: počet dní k ukládání historie položky: počet položek frekvence sběru: průměrná doba sběru dat bajty: počet bajtů potřebný k uložení hodnoty sledované položky, závisí na použité databázi, obvykle se pohybuje kolem 90 bajtů.
Data trendů	*dny · (položky / 3600) · 24 · 3600 · bajty* dny: počet dní k ukládání historie položky: počet položek frekvence sběru: průměrná doba sběru dat bajty: počet bajtů potřebný k uložení jednoho trendu, závisí na použité databázi, obvykle se pohybuje kolem 90 bajtů.
Data událostí	*dny · události · 24 · 3600 · bajty* události: počet událostí za vteřinu (nejhorší možný případ je jedna položka za vteřinu) dny: počet dní k ukládání historie bajty: počet bajtů potřebný k uložení jednoho trendu, závisí na použité databázi, obvykle se pohybuje kolem 170 bajtů.

Tabulka 15 - Vztahy pro výpočet místa na disku (Zdroj: Dokumentace Zabbix)

Pro databázi je vyhrazen zvláštní virtuální server Linux Debian 9. Jako databáze je použita MySQL. Požadavek na hardwarové prostředky je stanoven na: 2 CPU jádra, 2 GB operační paměti a 200 GB úložného prostoru.

4.5.3 Zabbix server a frontend

Pro Zabbix server a frontend je vyhrazen jeden společný virtuální server Linux Debian 9. Frontend vyžaduje instalaci Apache serveru a dalších potřebných knihoven pro správný běh. Požadavek na hardwarové prostředky je stanoven na: 2 CPU jádra, 2 GB operační paměti a 20 GB úložného prostoru.

4.6 Instalace serverů

Všechny součásti Zabbixu (databáze, server a frontend) budou nasazeny na virtuální Linuxové stroje vytvořené v prostředí VMware. Byl zvolen operační systém Linux Debian 9, který je v nejmenované společnosti již využíván u jiných serverů. Instalace serverů

proběhla z předem připravených iso souborů poskytnutých společností a nebude v této práci popsána z důvodu rozsahu práce. Byl dodržen standardní postup vytváření virtuálních strojů a použití předem připraveného obrazu disku s Linuxovou distribucí dle oficiální dokumentace VMware.

4.7 Nastavení serverů

Pomocí příkazů `nano /etc/hostname` a `nano /etc/hosts` byl nastaven hostname na *zabbix-db* a *zabbix-frontend*. Pro *zabbix-db* byla nastavena pevná IP adresa 192.168.80.80 a pro *zabbix-frontend* byla nastavena pevná IP adresa 192.168.80.81. IP adresa byla nastavena příkazem `nano /etc/network/interfaces` a následnou úpravou souboru na následující obsah:

```
iface ens192 inet static
address 192.168.80.80
netmask 255.255.255.0
gateway 192.168.80.1
dns-server 192.168.100.100
```

Nastavení pro *zabbix-frontend* proběhlo obdobným způsobem se změnou pouze v nastavení již zmiňované IP adresy na adresu 192.168.80.81.

4.8 Instalace databáze MySQL (Mariadb)

V této kapitole je popsána instalace databáze Mariadb. Jedná se o databázi postavenou na základech databáze MySQL, která je využívána v Linuxové distribuci Debian 9. Instalace je provedena s použitím následujících příkazů pod administrátorským účtem `root` na serveru *zabbix-db*:

```
shell> apt-get update
shell> apt-get install mysql-client mysql-server
```

V průběhu instalace je po výzvě instalačního procesu nutné nastavit heslo k přístupu do databáze. Heslo je nastaveno na hodnotu: *Zabbix.234*

Instalace pokračuje následujícími příkazy:

```
shell> mysql_secure_installation
Remove anonymous users? [Y/n] y
 ... Success!
```

```
Disallow root login remotely? [Y/n] y
... Success!
Remove test database and access to it? [Y/n] y
 - Dropping test database...
 ... Success!
 - Removing privileges on test database...
 ... Success!
Reload privilege tables now? [Y/n] y
 ... Success!
```

U příkazů zobrazených výše jsou po dotazech instalačního procesu postupně zvoleny možnosti odstranění anonymních uživatelů, zakázání vzdáleného přihlášení uživatele `root`, odstranění testovací databáze, zamezení přístupu k testovací databázi a následně znovu načtení přístupových práv k databázi.

Pro analýzu síťových spojení je využito nástroje `netstat`. Aby bylo možné tento nástroj používat, je nutné tuto službu nejdříve nainstalovat pomocí příkazu:

```
shell> apt-get install -y net-tools
```

Následujícím příkazem je možno zjistit, na které IP adrese je databáze spuštěna. Po provedení příkazu se vypíše hodnota 127.0.0.1, což znamená, že databáze je spuštěna na lokální IP adrese.

```
shell> netstat –nl | grep –E "(Local|3306)"
```

Následující příkaz provede zastavení běhu procesu databáze.

```
shell> systemctl stop mariadb
```

Následujícím příkazem je spuštěna editace souboru `50-server.cnf`. Jedná se o konfigurační soubor databáze, ve kterém je u řádku `bind adress` změněna hodnota z 127.0.0.1 na 0.0.0.0, a tím je docíleno toho, že databáze bude spuštěna na všech IP adresách.

```
shell> nano /etc/mysql/mariadb.conf.d/50-server.cnf
```

Po uložení konfiguračního souboru je opět spuštěn proces databáze a následně je příkazem `netstat` provedena kontrola, zda se projevila změna konfigurace.

```
shell> systemctl start mariadb
shell> netstat –nl | grep –E "(Local|3306)"
```

K zabezpečení přístupu do databáze je vhodné vynutit zadávání hesla k přihlášení do databáze i v případě, kdy je uživatel přihlášen jako `root`. Tuto problematiku řeší následující sekvence příkazů, ve kterých je nejdříve provedeno přihlášení do databáze a poté pomocí SQL příkazu změněno nastavení přístupových práv k databázi.

```
shell> mysql –u root –p
mysql> update mysql.user set plugin=null where user='root';
mysql> flush privileges;
```

Vytvoření databáze pro Zabbix je zajištěno následující posloupností příkazů. Nejdříve přihlášení do MySQL, poté vytvoření databáze s nastavením kódování znaků utf8, vytvoření uživatele `zabbix` s heslem `Zabbix.234` a nastavení plných práv pro tohoto uživatele k právě vytvořené databázi `zabbix`.

```
shell> mysql -uroot –pZabbix.234
mysql> create database zabbix character set utf8 collate
utf8_bin;
mysql> create user 'zabbix'@'%' identified by 'Zabbix.234';
mysql> grant all privileges on zabbix.* to 'zabbix'@'%'
identified by 'Zabbix.234';
mysql> quit;
```

Pro další konfiguraci databáze je nutné spustit instalační skripty (*schema.sql, images.sql* a *data.sql*) sloužící k vytvoření tabulek a dalšího nastavení databáze pro dohledový systém Zabbix. Skripty lze získat stažením zdrojových souborů dostupných z oficiálních webových stránek Zabbixu. Celý proces od stažení zdrojových souborů až po spuštění požadovaných skriptů je zajištěn následujícími příkazy:

```
shell> wget  -O /etc/stazene/zabbix-3.4.4.tar.gz
https://downloads.sourceforge.net/project/zabbix/ZABBIX%20Lat
est%20Stable/3.4.4/zabbix-
3.4.4.tar.gz?r=&ts=1511453813&use_mirror=10gbps-io
shell> tar -zxvf zabbix-3.4.4.tar.gz
shell> cd etc/stazene/zabbix-3.4.4/database/mysql
shell> mysql -uzabbix -pZabbix.234 zabbix <
etc/stazene/zabbix-3.4.4/database/mysql /schema.sql
```

```
shell> mysql -uzabbix -pZabbix.234 zabbix <
etc/stazene/zabbix-3.4.4/database/mysql /images.sql
shell> mysql -uzabbix -pZabbix.234 zabbix <
etc/stazene/zabbix-3.4.4/database/mysql /data.sql
```

4.9 Instalace Zabbix serveru a frontendu

Instalace Zabbix serveru a frontendu je provedena obdobným způsobem jako instalace databáze. Následujícími příkazy je zajištěno přidání úložiště, odkud jsou poté staženy instalační balíčky, ze kterých je vybrán a instalován Zabbix server.

```
shell> wget http://repo.zabbix.com/zabbix/3.4/debian/pool/mai
n/z/zabbix-release/zabbix-release_3.4-1+stretch_all.deb
shell> dpkg -i zabbix-release_3.4-1+stretch_all.deb
shell> apt-get update
```

Následující příkaz provede instalaci Zabbix serveru bez instalace MySQL databáze, která je již instalována na druhém serveru.

```
shell> apt-get --no-install-recommends install zabbix-server-
mysql
```

Po instalaci serveru následuje jeho konfigurace a nastavení přístupových informací k již vytvořenému databázovému serveru. Konfigurace je provedena spuštěním editace souboru `zabbix_server.conf` a následným vepsáním údajů zobrazených níže:

```
shell> nano /etc/zabbix/zabbix_server.conf
DBHost=192.168.80.80
DBName=zabbix
DBUser=zabbix
DBPassword=Zabbix.234
```

Následně je provedeno spuštění procesu Zabbix serveru, instalace `net-tools` k analýze síťových spojení příkazem `netstat`, kontrola běhu Zabbix serveru na portu 10051 a povolení automatického startu procesu Zabbix serveru po restartu virtuálního serveru, na kterém je Zabbix server nainstalován. Kroky popsané v tomto odstavci jsou realizovány pomocí následujících příkazů:

```
shell> service zabbix-server start
shell> apt-get install -y net-tools
```

```
shell> netstat -nl | grep -E "(Local|10051)"
shell> update-rc.d zabbix-server enable
```

Po instalaci Zabbix serveru následuje instalace frontendu, která je realizována na stejném virtualizovaném stroji jako Zabbix server. Frontend i Zabbix server se dělí o výkon, který je jim přiřazen v rámci jednoho serveru. Výkon serveru je navržen tak, aby byl plně dostačující k bezproblémovému provozu obou těchto částí Zabbixu.

Příkazem `apt-get install zabbix-frontend-php` je provedena instalace frontendu. Následně je nutné upravit konfigurační soubor Apache serveru, na kterém je frontend Zabbixu provozován. Následujícím příkazem je spuštěna editace konfiguračního souboru:

```
shell> nano /etc/apache2/conf-enabled/zabbix.conf
```

Zabbix pro svůj běh vyžaduje nastavení konkrétních hodnot, které jsou uvedeny níže. Posledním řádkem je nastaven čas pro Zabbix frontend dle toho, kde je fyzicky provozován.

```
php_value max_execution_time 300
php_value memory_limit 128M
php_value post_max_size 16M
php_value upload_max_filesize 2M
php_value max_input_time 300
php_value always_populate_raw_post_data -1
php_value date.timezone Europe/Prague
```

Po uložení upraveného konfiguračního souboru je proveden restart Apache serveru, za účelem aplikování provedených změn.

```
shell> /etc/init.d/apache2 restart
```

Další konfigurace frontendu probíhá již skrze okno internetového prohlížeče. Do adresního řádku je napsána IP adresa frontend serveru ve tvaru: IP adresa/zabbix (v tomto případě se jedná o 192.168.80.81/zabbix). Tato konfigurace je podrobněji popsána v kapitole 4.11 této práce.

4.10 Instalace agenta

Pro monitoring samotných serverů, na kterých je provozována databáze, Zabbix server a frontend, je využito instalace linuxového Zabbix agenta na oba servery. Níže je popsána instalace agentů a jejich konfigurace.

4.10.1 Frontend a Zabbix server

Následující příkazy zajistí nejprve instalaci agenta z balíčku a poté start Zabbix agenta.

```
shell> apt-get install zabbix-agent
shell> service zabbix-agent start
```

Dále je potřeba upravit konfigurační soubor Zabbix agenta. Příkazem `nano /etc/zabbix/zabbix_agentd.conf` je spuštěna editace konfiguračního souboru, do kterého jsou vepsány následující řádky textu zajišťující potřebné nastavení agenta. Vzhledem k tomu, že je agent nainstalován na identickém serveru jako samotný Zabbix server, je IP adresa nastavena na hodnotu lokální IP adresy.

```
Server=127.0.0.1
ServerActive=127.0.0.1
Hostname=zabbix-frontend
```

4.10.2 Databázový server

Instalace agenta na databázovém serveru je nepatrně odlišná od instalace na frontend a Zabbix serveru. Komunikaci mezi agentem na databázovém serveru a Zabbix serverem je doporučováno určitým způsobem zabezpečit. Neboť dochází k přeposílání dat mezi dvěma oddělenými servery. K zabezpečení komunikace je zvoleno šifrování PSK klíčem, jehož vygenerování a implementace je popsána níže v této kapitole.

Pro instalaci agenta je nutné nejdříve přidat úložiště s instalačními balíčky Zabbixu pomocí příkazu níže:

```
shell> wget http://repo.zabbix.com/zabbix/3.4/debian/pool/mai
n/z/zabbix-release/zabbix-release_3.4-1+stretch_all.deb
shell> dpkg -i zabbix-release_3.4-1+stretch_all.deb
```

Poté je provedena instalace agenta s použitím instalačního balíčku.

```
shell> apt-get update
shell> apt-get install zabbix-agent
```

Následující příkaz zajistí vygenerování klíče složeného z 64 náhodných písmen a číslic, které jsou zapsány do souboru `zabbix_agentd.psk`.

```
shell> sh -c "openssl rand -hex 32 >
/etc/zabbix/zabbix_agentd.psk"
```

Následujícím příkazem lze vypsat vygenerovaný klíč:

```
shell> cat /etc/zabbix/zabbix_agentd.psk
```

Výstupem je následující řetězec:

```
d52c3f448002dc9c93283d12345678b74f9241ad030a1bcb7b5defb9a093f
809
```

Následně je spuštěna editace konfiguračního souboru agenta příkazem `nano /etc/zabbix/zabbix_agentd.conf`. Na konec tohoto souboru je vepsán následující text:

```
Server=192.168.80.81
ServerActive=192.168.80.81
TLSConnect=psk
TLSAccept=psk
TLSPSKIdentity=PSK001
TLSPSKFile=/etc/zabbix/zabbix_agentd.psk
```

Tímto je agent nakonfigurován ke komunikaci se Zabbix serverem (IP adresa 192.168.80.81) a nastavení spojení skrze PSK klíč s identifikačním číslem `PSK001`. Následující příkazy zajistí spuštění procesu Zabbix agenta a jeho povolení po restartu serveru. Posledním příkazem je vypsán stav spuštěného procesu agenta.

```
shell> systemctl start zabbix-agent
shell> systemctl enable zabbix-agent
shell> systemctl status zabbix-agent
```

Tímto je dokončena instalace a konfigurace agenta pro databázový server. V další části této práce je popsáno, jak agenta přidat ve webovém rozhraní jako sledované zařízení, aby bylo umožněno zobrazovat získaná data.

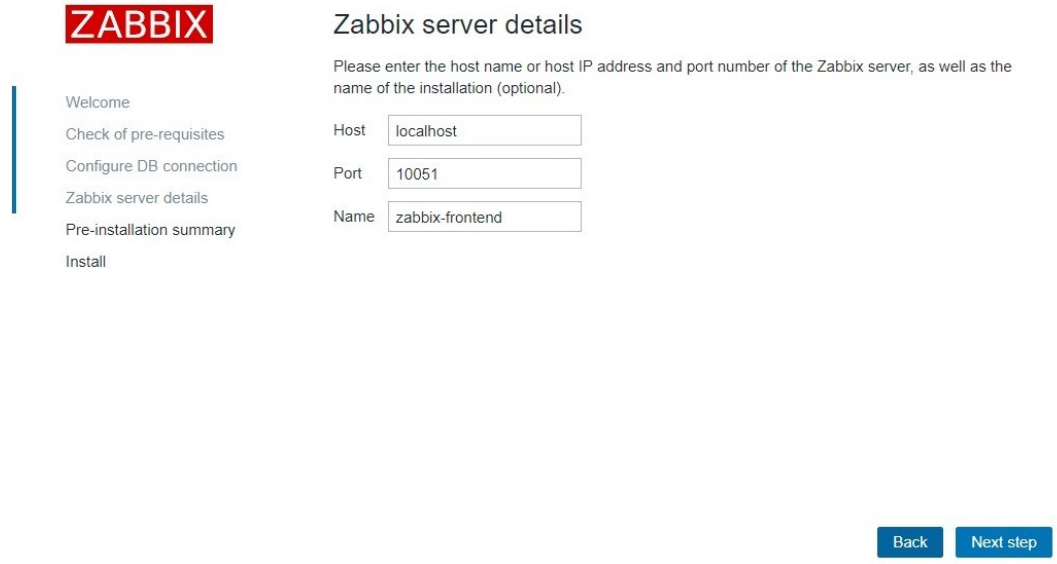
4.11 Základní konfigurace webového rozhraní

Základní konfigurace webového rozhraní probíhá zadáním IP adresy Zabbix frontendu (ve tvaru 192.168.80.81/zabbix) do adresního řádku prohlížeče. Po zadání se postupně zobrazí následující obrazovky:

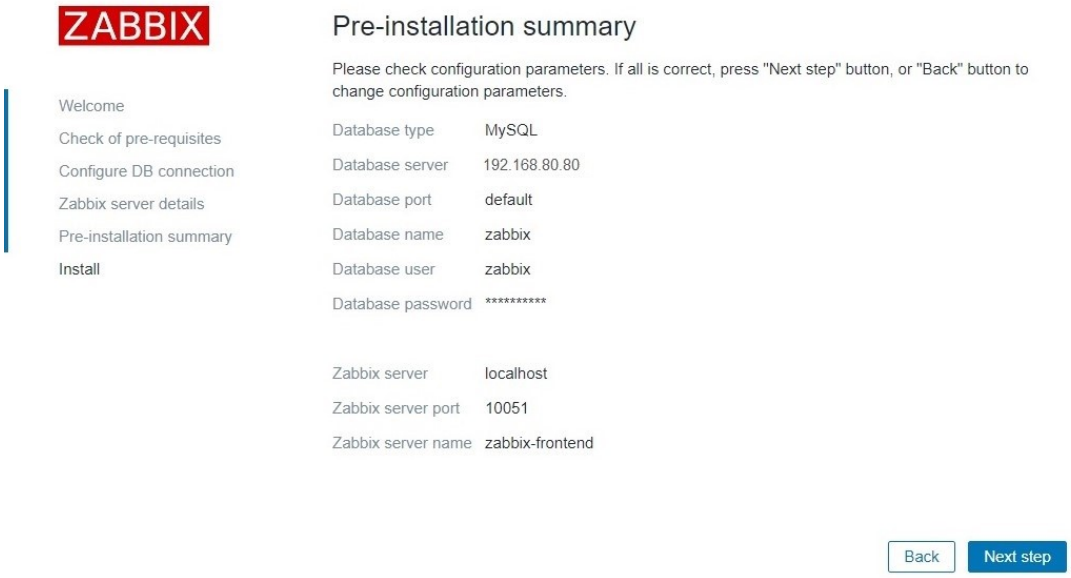
- uvítací obrazovka,
- kontrolní obrazovka nastavení frontendu (Obrázek 7),
- konfigurace připojení databáze (Obrázek 8),
- nastavení Zabbix serveru (Obrázek 9),
- shrnutí nastavení (Obrázek 10),
- informační obrazovka o dokončení instalace a vytvoření konfiguračního souboru,
- přihlašovací okno do administrace dohledového systému Zabbix.

Obrázek 7 - Kontrola nastavení frontendu (Zdroj: Autor)

Obrázek 8 - Konfigurace připojení databáze (Zdroj: Autor)



Obrázek 9 - Nastavení Zabbix serveru (Zdroj: Autor)



Obrázek 10 - Shrnutí nastavení (Zdroj: Autor)

Po instalaci jsou výchozí přihlašovací údaje nastaveny následovně:
Uživatelské jméno: Admin
Heslo: zabbix

Prvním krokem po přihlášení je změna tohoto výchozího hesla na heslo bezpečnější. Podle knihy *Perfect passwords* od autorů Burnetta a Kleimana je doporučováno zvolit heslo o minimálním počtu 8 znaků, přičemž bude obsahovat alespoň jeden zvláštní znak, alespoň jednu číslici a alespoň jedno velké písmeno. (Mark Burnett. Dave Kleimann, 2006).

Změna tohoto nastavení je provedena přes položku hlavního menu *Administration – Users*. Následně je vybrán uživatel s názvem *Admin*. Po zobrazení okna pro administraci údajů uživatele je do příslušných polí vyplněno nové heslo. Kliknutím na tlačítko *Update* je nové nastavení uloženo.

4.12 Konfigurace přihlašování uživatelů

V této kapitole je předvedeno nastavení přihlašování uživatelů do administrace dohledového systému pomocí LDAPS protokolu. Celý proces konfigurace je popsán v následujících bodech:

1. Na doménovém kontroleru (Microsoft Domain Controler) je vytvořen uživatelský účet s názvem *zabbix*. Tento účet má nastavena omezená práva a slouží pouze k přihlášení a čtení hesel ze služby MS Active Directory.
2. V systému Zabbix je vytvořen lokální uživatelský účet, jehož jméno je shodné s názvem uživatelského účtu na MS Active Directory.
3. K zajištění šifrované komunikace je vyžadován certifikát certifikační autority `FirmaCA.crt`. Jeho stažení je provedeno následujícím příkazem v terminálovém okně Zabbix serveru.

   ```
   shell> wget http://pki.firma.cz/ca/firmaCA.crt
   ```
4. Následně je certifikát umístěn do adresáře `/usr/share/ca-certificates/` a za pomoci příkazu níže je provedena editace konfiguračního souboru.

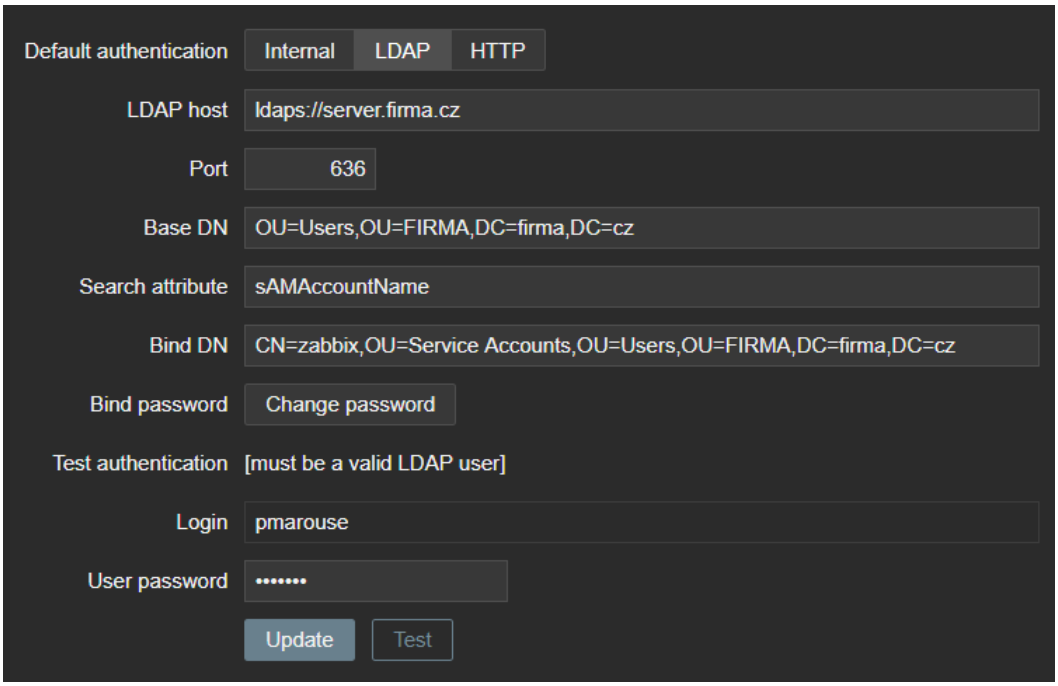
   ```
   shell> vim /etc/ca-certificates.conf.
   ```

   Na konec konfiguračního souboru je přidán název certifikátu `FirmaCA.crt`. Následně je provedena aktualizace seznamu důvěryhodných certifikačních autorit příkazem `update-ca-certificates`.
5. V tomto kroku je vytvořena žádost o certifikát `ldap.csr` a privátní klíč `ldap.key`. Tvorba žádosti je realizována příkazem uvedeným na dalším řádku.

   ```
   shell> openssl req -out ldap.csr -new -newkey rsa:2048 -
   nodes -keyout ldap.key
   ```
6. Klientský certifikát `ldap.crt` je vytvořen s využitím obsahu souboru žádosti o certifikát `ldap.csr` na webové stránce certifikační autority společnosti (ca.firma.cz/certsrv).
7. V konfiguračním souboru `ldap.conf` je postupně nastavena cesta k souboru certifikátů důvěryhodných certifikačních autorit (včetně certifikátu `FirmaCa.crt` vloženého autorem této práce), klientskému certifikátu a privátnímu klíči. Příkaz k editaci a konkrétní část konfiguračního souboru je zobrazen níže.

   ```
   shell> nano /etc/ldap/ldap.conf
   # TLS certificates (needed for GnuTLS)
   TLS_CACERT      /etc/ssl/certs/ca-certificates.crt
   LDAPTLS_CERT    /etc/ldap/ldap.crt
   LDAPTLS_KEY     /etc/ldap/ldap.key
   ```

8. Posledním krokem je konfigurace LDAPS autentizace ve webovém rozhraní Zabbixu, které je zobrazeno na Obrázek 11.

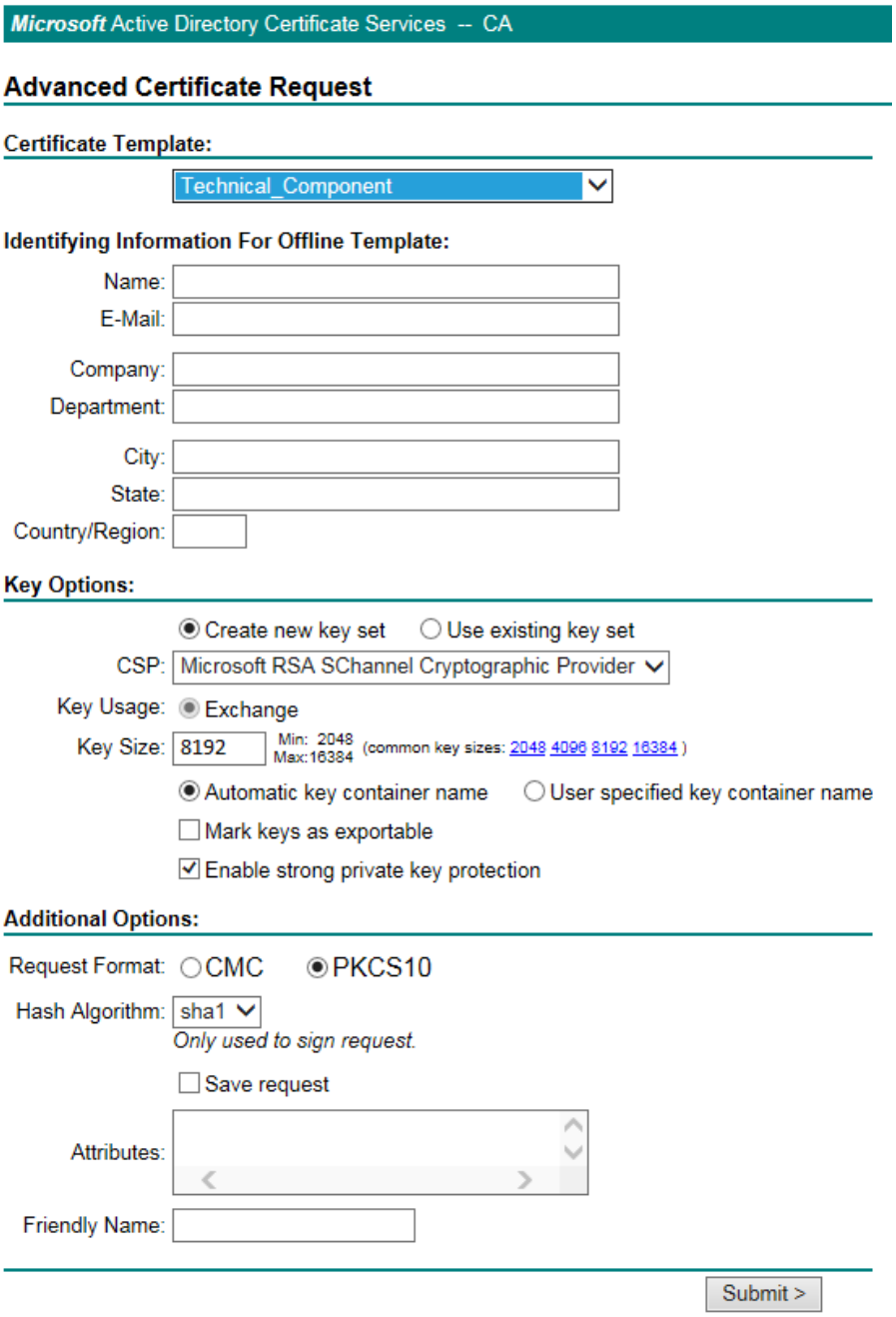


Obrázek 11 - Nastavení LDAP autentizace (Zdroj: Autor)

4.13 Konfigurace HTTPS

Pro správnou konfiguraci zabezpečení přístupu k webovému rozhraní jsou provedeny následující kroky:

- Je vytvořen záznam na DNS serveru pro IP adresu 192.168.80.81 s doménovým jménem zabbix-frontend.firma.cz.
- Je vygenerován certifikát pro zabbix-frontend.firma.cz. Vygenerování certifikátu je provedeno na webové stránce certifikační autority společnosti viz Obrázek 12.



Obrázek 12 - Tvorba certifikátu (Zdroj: Autor)

- Certifikát je nahrán na Zabbix server s využitím programu WinSCP.
- S použitím následujících dvou příkazů je certifikát rozdělen na *.key* a *.crt* soubory.

```
shell> openssl pkcs12 -in zabbix-frontend.pfx -nocerts
 -out zabbix-frontend.key
shell> openssl pkcs12 -in zabbix-frontend.pfx-clcerts -
nokeys -out zabbix-frontend.crt
```

- Certifikát i klíč jsou zkopírovány do adresáře `/etc/apache2/ssl/`
- Příkaz `a2enmod ssl` je použit k úpravě souboru `/etc/apache2/ports.conf` na následující obsah:

```
Listen 80
<IfModule ssl_module>
        Listen 443
</IfModule>
<IfModule mod_gnutls.c>
        Listen 443
</IfModule>
```

- Dále je upraven konfigurační soubor `sites-enabled` příkazem:

```
shell> nano /etc/apache2/sites-enabled/000-default.conf
```

Obsah editovaného souboru je změněn na:

```
<VirtualHost *:443>
   DocumentRoot /var/www/zabbix
   ServerName zabbix-frontend.firma.cz
   ErrorLog ${APACHE_LOG_DIR}/error.log
   CustomLog ${APACHE_LOG_DIR}/access.log combined
   SSLEngine on
   SSLCertificateFile /etc/apache2/certifikat/zabbix-
     frontend.crt
   SSLCertificateKeyFile /etc/apache2/certifikat/zabbix-
     frontend.key
   SSLCACertificateFile  /etc/ssl/certs/ca-
     certificates.crt
</VirtualHost>
```

- V posledním kroku je provedeno přesměrování z HTTP na HTTPS příkazem:

```
shell> /etc/apache2/sites-available/000-default.conf
```

Obsah souboru je změněn na následující obsah:

```
# Redirect to https
        RewriteEngine On
        RewriteCond %{HTTPS} off
```

```
RewriteRule (.*) https://%{SERVER_NAME}/$1 [R,L]
```

4.14 Konfigurace sledovaných zařízení

V této části práce jsou popsány konfigurace jednotlivých sledovaných zařízení rozdělených podle způsobu, jakým jsou monitorována. Mezi nejvíce používaný způsob monitoringu patří využití SNMP protokolu, kterým jsou monitorovány všechny důležité switch zařízení a vybrané servery společnosti. Monitoring dostupnosti je zajištěn protokolem ICMP. Mezi další podstatné konfigurace patří monitoring s využitím agenta a monitoring VMware skrze poskytované SDK.

4.14.1 Linux agent

Ke sledování samotných serverů, na kterých je nainstalován dohledový systém Zabbix, je využito Linuxových agentů. Linuxový agent je již na každém ze serverů nainstalován (viz kapitola 4.10). V následujících podkapitolách je popsána jejich konfigurace ve webovém prostředí Zabbix.

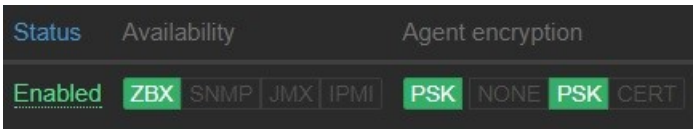
4.14.1.1 Zabbix server a frontend

Agent je spuštěn na stejném serveru společném pro Zabbix server i frontend, proto není potřeba šifrovat komunikaci mezi agentem a serverem. Agent je přidán ve webovém rozhraní přes nabídku *Configuration – Hosts*. Tlačítkem *Create host* je vytvořeno nové sledované zařízení. Následuje vyplnění základních informací, přidání hosta do skupiny *Linux servers* a zvolení předinstalované šablony pro *Linux OS*. Tímto způsobem jsou automaticky nastaveny položky (takzvané items), triggery a grafy pro námi sledované zařízení.

4.14.1.2 Databázový server

Agent pro databázový server je spuštěn na odděleném serveru od Zabbix serveru a frontendu, proto je potřeba zajistit bezpečnou komunikaci zašifrováním spojení mezi databázovým serverem a Zabbix serverem. Přidání nového sledovaného zařízení probíhá obdobným způsobem jako v předchozí kapitole s tím rozdílem, že je navíc nastaveno šifrování v sekci *Encryption* na PSK. Po uložení konfigurace a aktivování monitoringu lze po několika minutách zpozorovat zeleně označenou dostupnost a šifrování agenta na

znamení úspěšného nastavení a funkčního provozu agenta na sledovaném zařízení (viz Obrázek 13).



Obrázek 13 - Úspěšné nastavení databázového agenta (Zdroj: Autor)

4.14.2 SNMP

V následujících podkapitolách je popsána konfigurace zařízení, která jsou sledována přes protokol SNMP. Jedná se o Cisco switche a IBM servery.

4.14.2.1 Cisco switch

Na základě této práce je autorem prováděn monitoring čtyř hlavních switch zařízení Cisco ve společnosti. Vzhledem k tomu, že konfigurace každého zařízení probíhá obdobným způsobem, je v této práci popsána konfigurace pouze jednoho z nich.

Pro konfiguraci sledování je po přihlášení do Zabbix administrace vybrána z hlavní nabídky položka *Configuration – Hosts*. Tlačítkem *Create host* je vytvořeno nové zařízení. Následně je nastaven název zařízení na hodnotu *sw1c*. Pro tento typ zařízení je tlačítkem *Create group* vytvořena nová skupina s názvem *Cisco switche*, ve které se budou všechna zařízení stejného typu sdružovat. Do řádku *SNMP interfaces* je vyplněna hodnota IP adresy zařízení (192.168.20.65) a port (161).

Na záložce *Templates* jsou vybrány přednastavené šablony: *Template Module Cisco CISCO-ENVMON-MIB SNMPv2*, *Template Module Cisco CISCO-PROCESS-MIB SNMPv2* a *Template Module Generic SNMPv2*. Těmito šablonami jsou přednastaveny sledované položky, triggery, grafy a zjišťovací pravidla. Zjišťovací pravidla mají za následek automatické přidávání nových položek, triggerů a grafů na právě přidaném zařízení.

Na záložce *Macros* je nutné vytvořit nové makro pro nastavení community stringu. V tomto makru je nastaveno jméno na *{$SNMP_COMMUNITY}* a hodnota tohoto makra na *firma-net*. Tímto je zajištěno, že Zabbix bude pro komunikaci s tímto zařízením používat community string s hodnotou *firma-net*, čímž je současně povoleno získávat data přes SNMP protokol.

Kliknutím na tlačítko *Add* je přidáno nové zařízení k monitoringu. Následným kliknutím na tlačítko *Enabled* je aktivován jeho monitoring. Protože se u nově přidaného zařízení zobrazuje jen několik základních položek k monitoringu, je potřeba aktivovat automatické zjišťování, aby bylo možné sledovat větší množství položek. Na následujícím obrázku (Obrázek 14) lze vidět pravidla, která jsou k dispozici na základě zvolených šablon. V základním nastavení jsou tato pravidla deaktivována a nastavena na interval zjišťování 3600 vteřin.

Name ▲	Items	Triggers	Graphs	Hosts	Key	Interval	Type	Status
Template Module Cisco CISCO-PROCESS-MIB SNMPv2: CPU Discovery	Item prototypes 2	Trigger prototypes 2	Graph prototypes 2	Host prototypes	cpu.discovery	3600	SNMPv2 agent	Disabled
Template Module Cisco CISCO-ENVMON-MIB SNMPv2: FAN Discovery	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	fan.discovery	3600	SNMPv2 agent	Disabled
Template Module Cisco CISCO-ENVMON-MIB SNMPv2: PSU Discovery	Item prototypes 1	Trigger prototypes 1	Graph prototypes	Host prototypes	psu.discovery	3600	SNMPv2 agent	Disabled
Template Module Cisco CISCO-ENVMON-MIB SNMPv2: Temperature Discovery	Item prototypes 2	Trigger prototypes 3	Graph prototypes	Host prototypes	temperature.discovery	3600	SNMPv2 agent	Disabled

Obrázek 14 - Přehled pravidel automatického zjišťování (Zdroj: Autor)

V důsledku aktivace automatického zjištění dochází zpravidla k vyššímu zatížení sledovaného zařízení. Je proto důležité zvolit rozumný časový interval či namísto intervalu zvolit možnost naplánování spuštění automatického zjištění na určitý okamžik. V tomto konkrétním případě zvolil autor práce jako termín spuštění akce sobotu 2 hodiny po půlnoci, neboť je vytížení switche v této době zpravidla nejnižší. Nemůže se tedy stát, že by vlivem nárůstu komunikace mezi Zabbixem a sledovaným switchem došlo k ohrožení provozu.

Po provedení automatického zjištění je k dispozici 69 položek, 90 triggerů a 4 grafy. Z těchto položek si lze poté vybrat jen takové, které jsou pro samotný monitoring podstatné. Autor práce po konzultaci s odpovědnými osobami za infrastrukturu společnosti vybral k monitoringu následující položky: vytížení procesoru, teplota modulů, ICMP ping, ICMP ztráta, doba od spuštění zařízení, název zařízení a popis zařízení.

Čas od spuštění zařízení je získáván vždy jednou za 30 vteřin a je v databázi uchován vždy po dobu 14 dní. ICMP ping a ztráta je zjišťována jednou za 60 vteřin a data jsou uchovávána po dobu jednoho týdne. Hodnoty teplot jednotlivých modulů a vytížení procesoru jsou získávány v intervalu 180 vteřin a jsou uchovávány po dobu 30 dní. Název zařízení a popis je zjišťován vždy jednou za hodinu a je uchováván po dobu 14 dní.

4.14.2.2 IBM server

Ve společnosti se nachází dva servery od společnosti IBM. U těchto serverů je žádoucí sledovat jejich dostupnost a v případě nedostupnosti okamžitě informovat odpovědné zaměstnance. Vzhledem k tomu, že jsou oba servery stejného typu, je v této práci popsána konfigurace pouze jednoho z nich.

Pro konfiguraci sledování je po přihlášení do Zabbix administrace vybrána z hlavní nabídky položka *Configuration – Hosts*. Tlačítkem *Create host* je vytvořeno nové zařízení. Následně je nastaven název zařízení na hodnotu IBM-p7a. Pro tento typ zařízení je tlačítkem *Create group* vytvořena nová skupina s názvem *IBM servery*, ve které se budou všechna zařízení stejného typu sdružovat. Do řádku *SNMP interfaces* je vyplněna hodnota DNS názvu zařízení (p7-a.firma.cz).

Na záložce *Templates* je vybrána přednastavená šablona *ICMP ping*, která obsahuje položky pro sledování dostupnosti zařízení, ztrátovosti packetů a dobu odpovědi na dotaz. V návaznosti na tyto položky jsou v šabloně obsaženy také triggery, které slouží ke spuštění upozornění na vzniklou událost. Nastavení upozornění na události a spuštění akcí jsou popsány v pozdějších kapitolách této práce.

Hodnoty dostupnosti zařízení, ztrátovosti packetů a doby odpovědi na dotaz jsou zjišťovány vždy jednou za 60 vteřin, data jsou uchovávána po dobu jednoho týdne.

4.14.3 VMware

Pro konfiguraci sledování VMware je zapotřebí nejdříve upravit konfigurační soubor Zabbixu na serveru. V souboru `/etc/zabbix/zabbix-server.conf` je s použitím příkazu `nano` připsán na konec souboru řádek `StartVMwareCollectors=2`. Tímto je zajištěno spuštění procesu, který bude sbírat data z VMware. Hodnota 2 je nastavena dle specifikací a doporučení v oficiální dokumentaci Zabbixu v závislosti na počtu VMware zařízení. Po uložení změněného konfiguračního souboru Zabbixu je nutné provést restart Zabbix serveru příkazem `service zabbix-server restart`.

Pro konfiguraci ve webovém rozhraní Zabbixu je z hlavní nabídky vybrána položka *Configuration – Hosts*. Tlačítkem *Create host* je vytvořeno nové zařízení. Následně je nastaven název zařízení na hodnotu *vmware* a zvolena skupina *Discovered hosts*. Na

úvodní straně konfigurace nového zařízení je také potřeba vyplnit IP adresu, na které je VMware provozován, tedy na hodnotu *192.168.90.10*.

Po kliknutí na záložku *Templates* je vybrána šablona s názvem *Template VM VMware*. Tato šablona obsahuje pravidla automatického zjišťování pro konfiguraci VMware hypervizorů, hostitelů a clusterů. Použití této šablony a v ní obsažených pravidel ušetří uživateli značné množství práce, neboť vyhledání a přidání nových zařízení probíhá díky tomuto mechanismu automaticky. Pravidla automatického zjišťování jsou přednastavena na interval jedné hodiny, ve kterém vždy proběhne kontrola potenciálních nových zařízení.

Pro připojení k VMware je nutné Zabbixu definovat makra. Na záložce *Macros* jsou proto přidána následující makra a jejich hodnoty:

- {$URL} = https://192.168.90.10/sdk
- {$USERNAME} = zabbix
- {$PASSWORD} = FirmaHeslo.6

Jak lze vyčíst z údajů výše, pro připojení do Zabbixu je využito SDK[36] provozovaného VMware, dále pak uživatelského jména a hesla.

Po uplynutí jedné hodiny proběhne automatické zjišťování zařízení a dle přednastavené šablony je automaticky přidáno nové hostitelské zařízení s názvem *localhost.firma.cz*. Jedná se o nově nalezeného hypervizora VMware. K tomuto zařízení jsou automaticky přidány položky, jako jsou například: model procesoru, počet jader procesoru, využití procesoru, celková paměť, využitá paměť, název clusteru, počet přijatých a odeslaných bajtů a uplynulý čas od spuštění. Všechny hodnoty položek jsou uchovávány po dobu 90 dní. U položek, ve kterých dochází k častým změnám hodnot (využití procesoru, využití paměti a počtu přijatých či odeslaných bajtů) jsou hodnoty zjišťovány v intervalu jedné minuty. U položek jako jsou například celková paměť, model procesoru, počet jader procesoru jsou hodnoty získávány v intervalu jedné hodiny, neboť u nich nedochází k tak častým změnám. Časté dotazování by také mělo za následek nadměrné zatěžování sledovaného zařízení.

[36] Software development kit – soubor nástrojů pro vývoj software

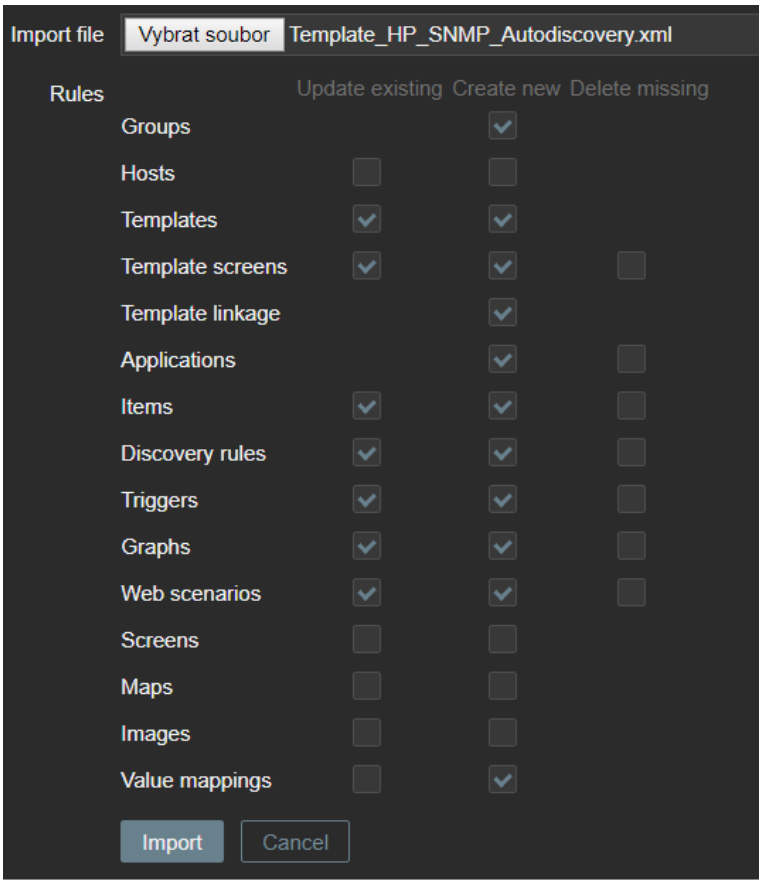
4.14.4 IBM GPFS

K monitoringu IBM GPFS je využito šablon, které nejsou obsahem základní instalace Zabbixu. K rozšíření funkčnosti Zabbixu je možné šablony stáhnout z oficiální webové stránky share.zabbix.com určené ke sdílení šablon vytvořených uživateli. Šablony lze vyhledávat podle názvu, typu zařízení a dalších kritérií. Ostatní uživatelé mohou šablony hodnotit a přidávat komentáře. V následujících dvou kapitolách je popsáno stažení, import šablony a použití k monitoringu.

4.14.4.1 Import šablon

Na webové stránce www.share.zabbix.com je do vyhledávání zadán výraz *HP server*, který zajistí vyhledání potřebné šablony. Z výsledku vyhledaných šablon je na základě kladného hodnocení a podrobného popisu autora šablony vybrána a stažena šablona s názvem HP server snmp autodiscovery (ke stažení zde: https://share.zabbix.com/cat-server-hardware/hp/hp-server-snmp-autodiscovery-template).

Ve webovém rozhraní Zabbixu je přes nabídku *Configuration – Templates – Import* vyvoláno okno k nahrání šablony zobrazené na následujícím obrázku.

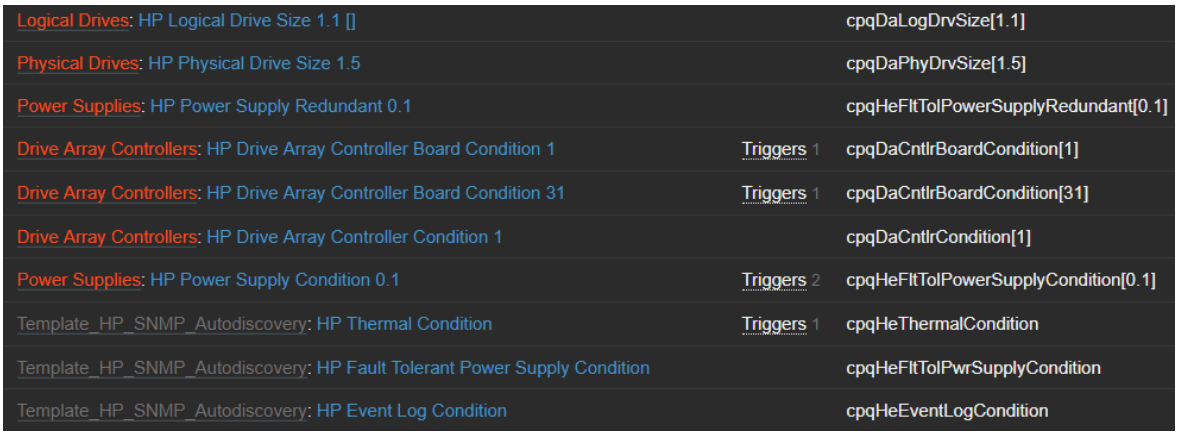


Obrázek 15 - Import šablony (Zdroj: Autor)

V tomto okně lze zvolit, které části šablony aktualizovat, které nově vytvořit a které smazat. Na Obrázek 15 je zobrazeno výchozí nastavení při importu šablony.

4.14.4.2 Konfigurace monitoringu

Konfigurace monitoringu proběhla obdobným způsobem jako při nastavování monitoringu IBM serveru, který je popsán v kapitole 4.14.2.2 této práce, a z tohoto důvodu zde nebude podrobněji popsán. Jediným rozdílem je využití nově importované šablony, která v sobě obsahuje pravidla pro automatické zjištění sledovaných položek. Na následujícím obrázku je zobrazen výběr z několika položek, které byly nalezeny.



Obrázek 16 - Položky z importované šablony (Zdroj: Autor)

4.15 Tvorba položky

Prakticky všechny položky lze sledovat s využitím předem implementovaných oficiálních šablon. V případě, kdy se nehodí využít šablonu nebo pro potřebnou položku šablona neexistuje, lze vytvořit sledovanou položku manuálně ve webovém rozhraní Zabbixu.

Zabbix například obsahuje šablonu *Template Module Interfaces Simple SNMPv2*, která automaticky vyhledá všechny dostupné porty switche a automaticky vytvoří ke každému portu několik položek. Při vypracovávání této práce bylo otstováno použití výše zmiňované šablony na 48 portovém switchi. Tento postup měl za následek automatické vygenerování téměř 3000 nových položek, což zapříčinilo nežádoucí krátkodobé vytížení sledovaného zařízení i Zabbixového serveru. Ve společnosti byl stanoven požadavek na sledování objemu příchozích a odchozích dat pouze u jednoho portu. Proto bylo od využití této šablony upuštěno a sledování položky bylo konfigurováno manuálně ve webovém rozhraní Zabbixu, jež je podrobněji popsáno v následujících odstavcích této kapitoly.

Pro konfiguraci nové položky jsou ve webovém rozhraní Zabbixu vybrány postupně položky *Configuration – Hosts – sw1c – Items – Create item.* Následně jsou vyplněny požadované informace pro nastavení položky, která bude sloužit pro sledování objemu příchozích dat z páteřního switche do switche s názvem *sw1c*. Obdobným způsobem je provedena i konfigurace položky pro sledování objemu odchozích dat ze zařízení *sw1c* do páteřního switche. Hodnoty pro konfiguraci jsou shrnuty v následující tabulce.

Název pole	Hodnota
Name	Uplink << sw-core
Type	SNMPv2 agent
Key	locIfInBitsSec-sw1c-Po100
Host interface	192.168.20.65:161
SNMP OID	1.3.6.1.4.1.9.2.2.1.1.6.298
SNMP community	Firma-cz-64823
Units	Bits/s
Update interval	120s
History storage period	90d

Tabulka 16 - Konfigurace položky (Zdroj: Autor)

Popis polí tabulky:

- Pole *Name* obsahuje název sledované položky.
- Pole *Type* obsahuje způsob, jakým je získávána hodnota položky.
- Pole *Key* obsahuje jedinečný identifikátor jedné konkrétní položky.
- *Host interface* obsahuje IP adresu sledovaného zařízení a port.
- Číslo v poli *SNMP OID* vyjadřuje cestu ke konkrétní hledané hodnotě. Lze jí získat například spuštěním příkazu *snmpwalk* s příslušnými parametry na sledovaném zařízení, prohlížením MIB tabulky pomocí specializovaného programu (například MIB Browser) či vyhledáním na webových stránkách (například http://www.oidview.com).
- *SNMP community* obsahuje předem definovanou hodnotu, která zastupuje funkci hesla umožňujícího Zabbixu získávat data ze sledovaného zařízení.
- *Units* definuje jednotky, ve kterých budou získaná data prezentována.
- *Update interval* nastavuje interval získávání nové hodnoty. V tomto případě je interval nastaven na 120 vteřin.
- *History storage period* obsahuje informaci o délce uchovávání získané hodnoty. V tomto případě je hodnota uložena 90 dní. Po uplynutí této doby je hodnota nenávratně vymazána z databáze.

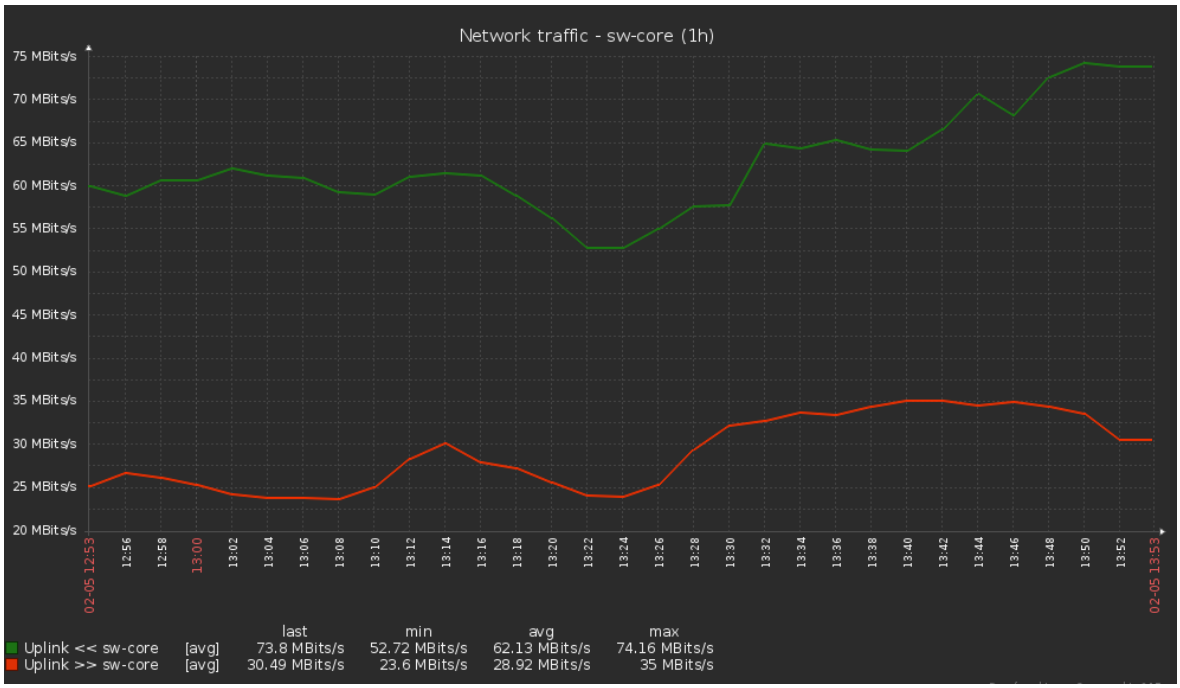
4.16 Tvorba grafu

V této kapitole je popsáno, jakým způsobem lze vytvořit nový graf. Zabbix umožňuje pro všechny položky pracujícími s číselnými hodnotami, automaticky vytvářet grafy, ve kterých jsou tyto hodnoty zobrazeny v průběhu času.

V případě, kdy je potřeba zobrazit například více položek do jednoho grafu, je možné si vytvořit nový graf na základě uživatelem specifikovaných požadavků. Ve

společnosti byl stanoven konkrétní požadavek na zobrazení hodnot příchozího a odchozího objemu dat ze switche do hlavního páteřního switche.

Pro vytvoření nového grafu jsou ve webovém rozhraní Zabbixu vybrány postupně položky *Configuration – Hosts – sw1c – Graphs – Create graph*. Následně je vyplněn název grafu, jsou nastaveny rozměry grafu, typ grafu (například koláčový, sloupcový, linkový) a poté pomocí zaškrtávacích polí zvoleno zobrazení popisků grafu, zobrazení triggerů a zobrazení času. Jako poslední jsou pomocí tlačítka *Add* vybrány položky grafu, které chceme v grafu zobrazit. V nově vytvořeném grafu autor práce vybral položky *Uplink << sw-core* a *Uplink >> sw-core*. U každé položky je možné zvolit barvu čáry, typ čáry a funkci (například průměr, maximum, minimum). Kliknutím na *Preview* lze zobrazit náhled grafu. Na Obrázek 17 je zobrazen výsledný graf síťového provozu mezi zařízením *sw1c* a páteřním switchem.



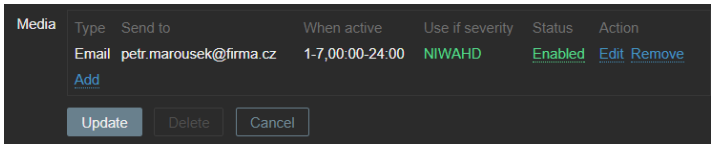
Obrázek 17 - Graf přenosu dat mezi sw1c a páteřním switchem (Zdroj: Autor)

4.17 Konfigurace upozornění

V následujících podkapitolách je popsáno nastavení upozornění na vzniklé události. Upozornění je uživatelům doručováno formou elektronické zprávy. V další části této kapitoly je popsána konfigurace triggeru a nastavení akce, které je nutné konfigurovat k zajištění funkčnosti notifikací.

4.17.1 Nastavení e-mailového upozornění

K upozornění na nově vzniklé události z dohledového systému Zabbix je ve společnosti využíváno notifikací přes e-mailové zprávy. Ke každému uživateli lze přes nabídku *Administration – Users* po vybrání konkrétního uživatele nastavit e-mailovou adresu (viz Obrázek 18). Na tuto adresu je poté v případě výskytu určité události odeslána zpráva s popisem problému v přednastaveném formátu.



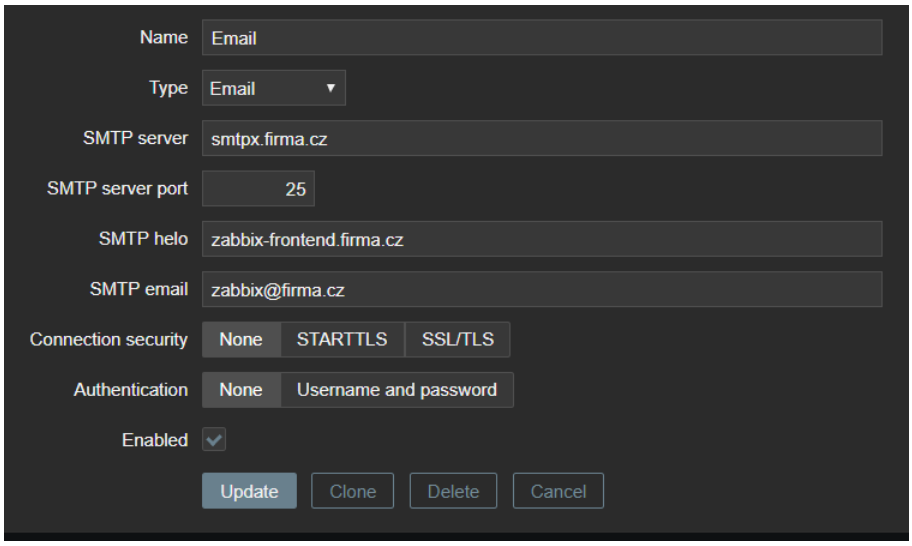
Obrázek 18 - Nastavení e-mailové adresy uživatele (Zdroj: Autor)

Na Obrázku 18 je znázorněno také nastavení času, ve kterém je povoleno odesílání elektronických zpráv uživateli. Autor práce nastavil čas notifikací na všechny dny v týdnu v jakoukoli dobu (hodnoty *1 – 7* a *00:00 – 24:00* v Obrázku 18).

Lze nastavit i závažnost události, na kterou je uživatel upozorněn. V Obrázku 18 je to znázorněno písmeny *NIWAHD*, kde každé písmeno zastupuje určitý stupeň závažnosti vzniklé události, například písmeno H = High čili vysoká závažnost.

K odesílání e-mailových notifikací je využito již existujícího e-mailového serveru ve společnosti. Aby bylo umožněno předávání e-mailových zpráv z dohledového systému Zabbix zmiňovanému SMTP serveru ve společnosti, byl na Zabbix server nainstalován program sSMTP (simple SMTP). Jedná se o MTA (message transfer agent), který zajišťuje přenos zpráv elektronické pošty.

Ve webovém rozhraní Zabbix je přes nabídku *Administration – Media types – Email* provedena konfigurace upozornění e-mailem. Nastavení je znázorněno na následujícím obrázku.

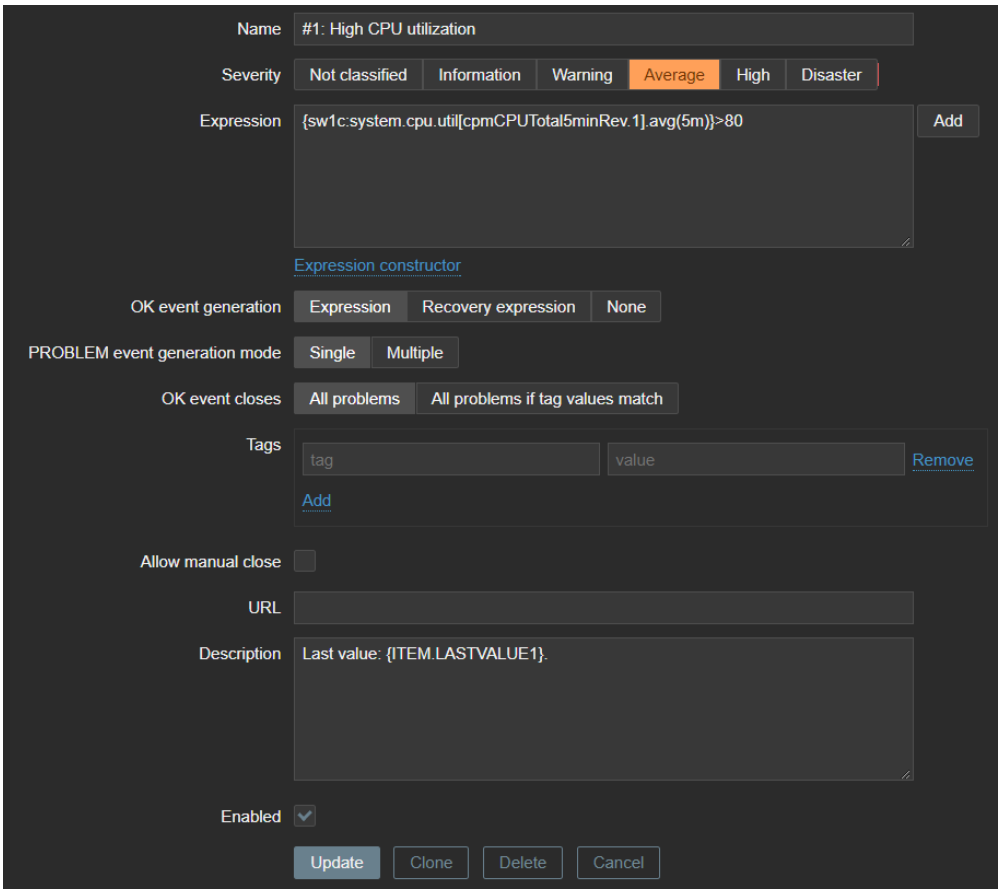


Obrázek 19 - Konfigurace upozornění e-mailem (Zdroj: Autor)

4.17.2 Konfigurace triggeru

Samotné nastavení e-mailových upozornění nedostačuje k tomu, aby uživatelé dohledového systému dostávali zprávy o vzniklých událostech. Ke sledování vzniklých událostí je nutné nastavit trigger. V této kapitole je popsáno nastavení jednoho ukázkového triggeru. Nastavení dalších triggerů probíhá obdobným způsobem a v dohledovém systému je jich obsaženo několik desítek. Popisovat každý trigger není pro tuto práci podstatné, a proto je zde popsán pouze jeden.

Konfigurace triggerů se provádí ve webovém rozhraní Zabbix přes nabídku *Configuration – Hosts – Triggers – Create trigger*. Nastavení triggeru je zobrazeno na následujícím obrázku.



Obrázek 20 - Konfigurace triggeru (Zdroj: Autor)

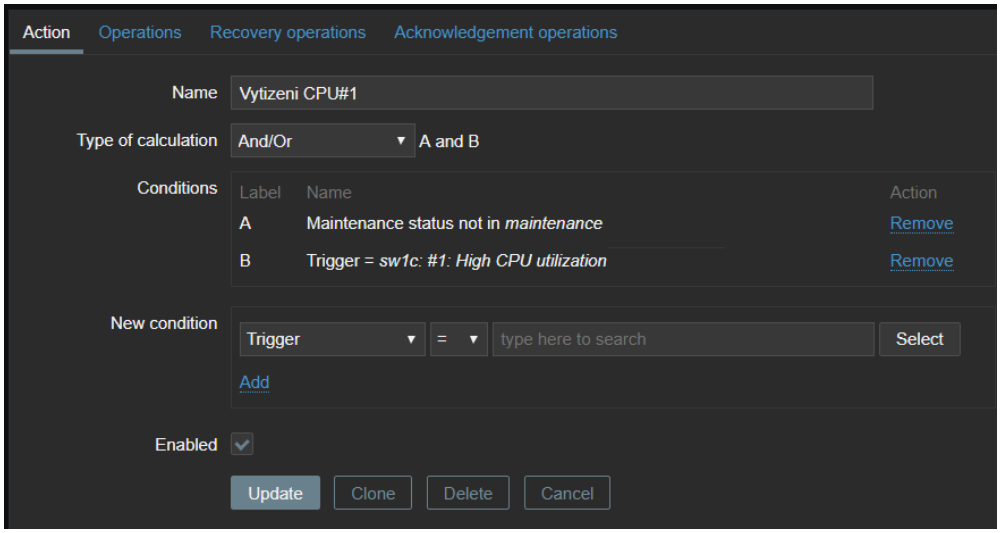
Z výše uvedeného obrázku je nejdůležitější pole s názvem *Expression*, ve kterém je vepsán výraz podle stanovených pravidel. V tomto konkrétním případě výraz vyjadřuje sledování hostitelského zařízení s názvem *sw1c*, u kterého je sledováno průměrné vytížení prvního procesoru za dobu 5 minut. V podmínce je dále stanoveno, že trigger má být spuštěn v případě hodnoty přesahující 80% vytížení. Závažnost tohoto triggeru je nastavena na hodnotu *Average* čili průměrný stupeň rizika.

V následující kapitole je popsáno přiřazení triggeru k akci takovým způsobem, aby byli uživatelé upozorněni skrze zprávu elektronické pošty.

4.17.3 Konfigurace akce

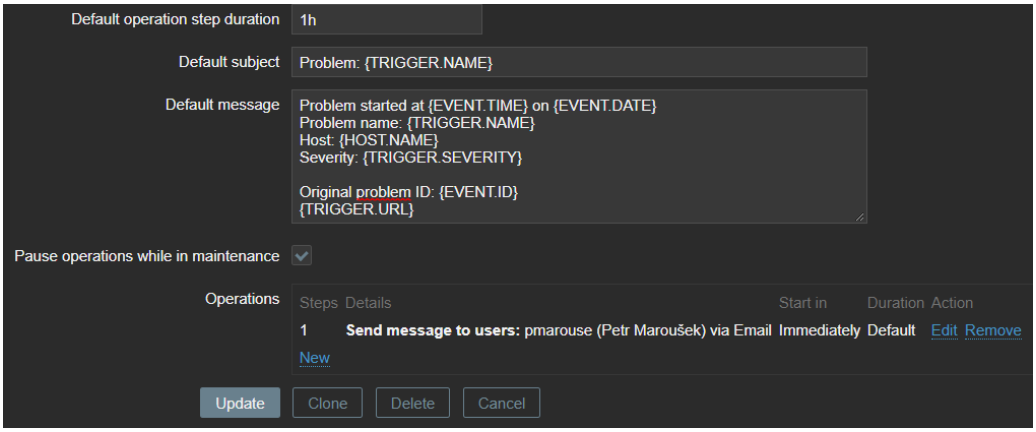
Před samotným informováním uživatele o vzniklé události je nutné nejdříve vytvořit a nastavit akci, které bude trigger přiřazen. Nastavení akcí se nachází ve webovém rozhraní Zabbixu v sekci *Configuration – Actions – Create action*. Na Obrázek 21 je zobrazeno nastavení akce s využitím vytvořeného triggeru (*1# High CPU utilization*). Lze vkládat několik triggerů najednou a výrokovou logikou nastavit, kdy má nastat vyhodnocení

podmínek. V tomto případě je nastavena podmínka *A – Maintenance status not in maintenance* (zařízení není ve stavu údržby) a podmínka *B – Trigger = sw1c: #1: High CPU utilization*, které musí platit současně.



Obrázek 21 - Konfigurace akce (Zdroj: Autor)

Pokud jsou obě tyto podmínky splněny, vykoná se akce definovaná v záložce *Operations*. Na této záložce (viz Obrázek 22) je nastaven předmět zprávy na název triggeru. Tělo zprávy obsahuje čas a datum výskytu události, název triggeru, název zařízení, závažnost události, ID triggeru a odkaz na trigger. Tlačítkem *New* je přidána operace *Send message to users*, ve které jsou vybráni všichni uživatelé, kterým bude zpráva odeslána.



Obrázek 22 - Nastavení operace (Zdroj: Autor)

Obdobným způsobem jako na záložce *Operations* je provedeno i nastavení na záložkách *Recovery operations,* které slouží k odeslání zprávy po vyřešení vzniklého

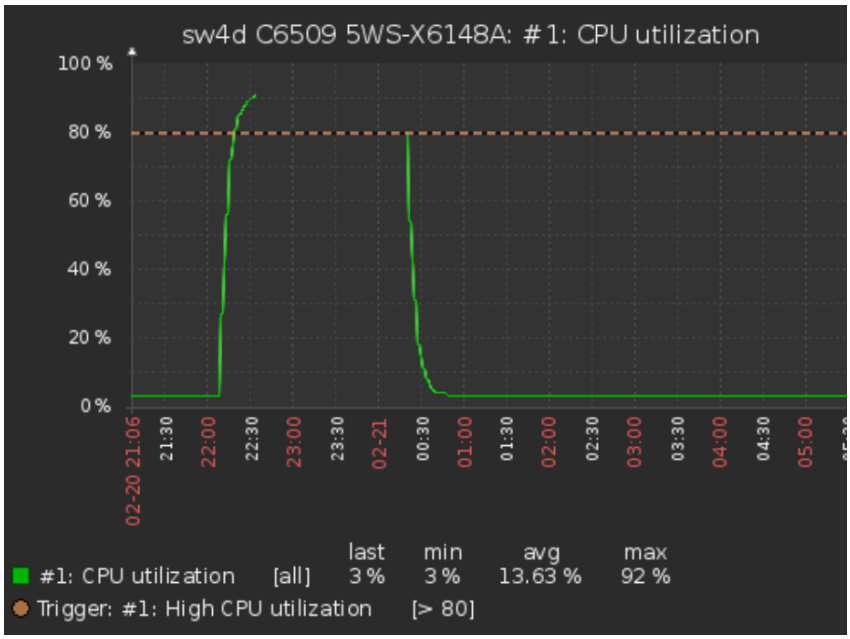
problému, a *Acknowledgement operations,* které slouží k informování vybraných uživatelů v případě, že některý z uživatelů potvrdil existenci problému.

4.18 Výstupy sledovaných zařízení

V následujících kapitolách jsou uvedeny vybrané výstupy, které lze získat z dohledového systému Zabbix. S jejich pomocí lze například získat přehled o sledovaných zařízeních či vyskytujících se problémech.

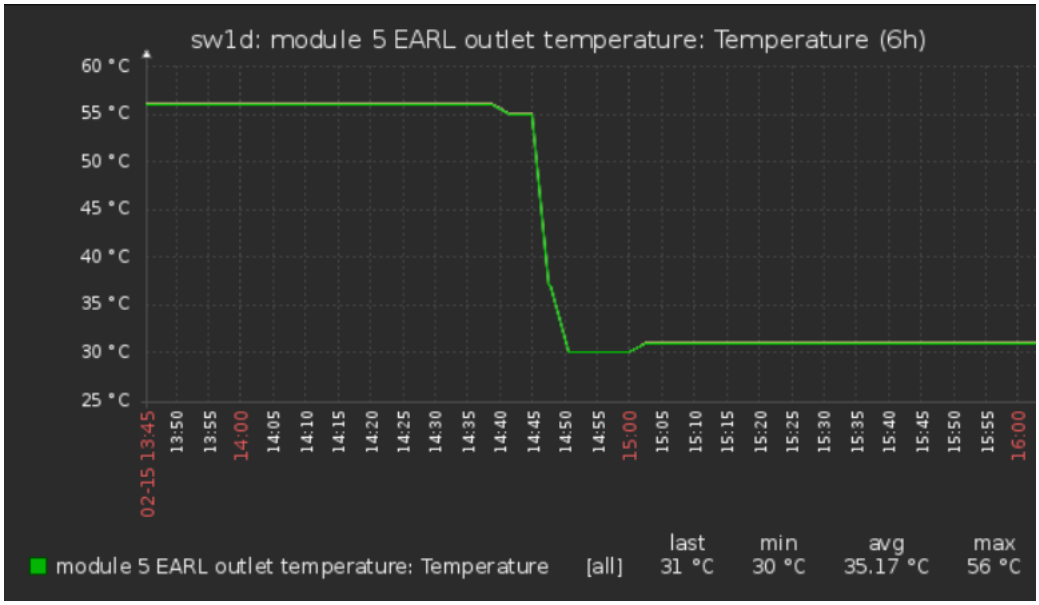
4.18.1 Grafy

Na Obrázek 23 je zobrazen graf vytížení CPU switche. Graf zobrazuje na svislé ose procenta a na vodorovné ose čas. Zelená linka vyznačuje aktuální vytížení procesoru switche v průběhu času. Přerušovanou čarou je v grafu vyobrazena hladina triggeru, který spustí akci v případě, že vytížení přesáhne hranici 80%. Pod grafem jsou zobrazeny statistické údaje pro aktuální výřez grafu a zobrazují minimální, maximální, průměrnou a poslední získanou hodnotu ze sledovaného zařízení.



Obrázek 23 - Vytížení CPU switche (Zdroj: Autor)

Na dalším obrázku je na grafu zobrazen průběh teploty jednoho z modulů switche v závislosti na čase. Podobně jako na předchozím grafu jsou pod grafem vypsány statistické údaje o minimální, maximální, průměrné a poslední získané hodnotě ze sledovaného zařízení.



Obrázek 24 - Graf teploty modulu switche (Zdroj: Autor)

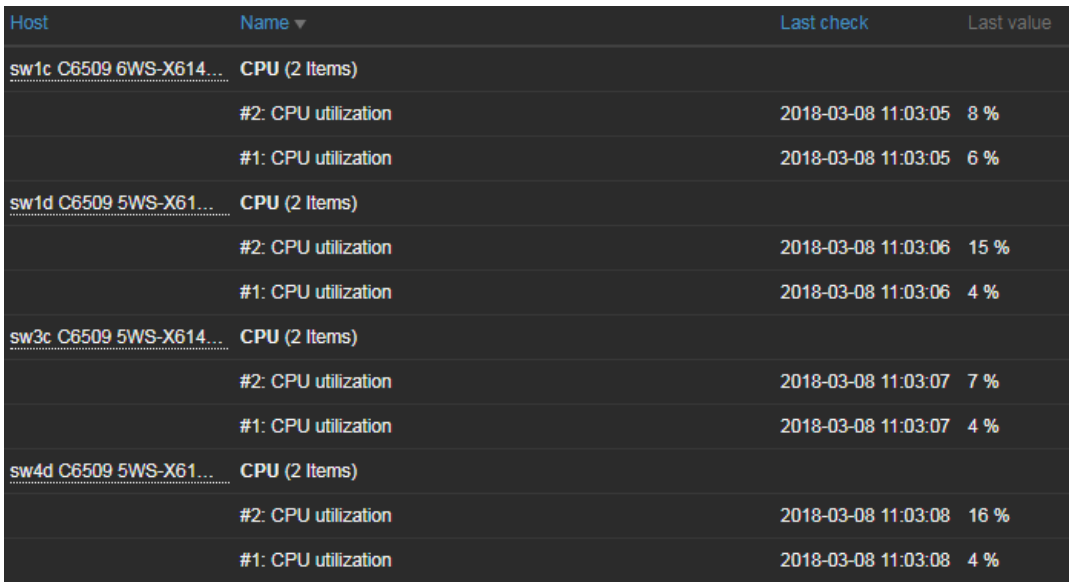
4.18.2 Reporty

Zabbix umožňuje zobrazení přehledu všech vyskytnutých problémů. Přehled je dostupný přes položku hlavního menu webového prostředí *Monitoring – Problems*. Problémy jsou primárně seřazeny podle času od nejnovějšího. Dále je v přehledu zobrazena závažnost problému, název zařízení, popis problému a čas vyřešení problému.

Time ▼		Severity	Recovery time	Status	Info	Host	Problem
2018-01-08 13:13:35		Warning	2018-01-08 13:20:35	RESOLVED		sw1c C6509 6WS-X6148A	sw1c C6509 6WS-X6148A has been restarted
2018-01-08 13:09:37		Warning	2018-01-08 13:16:07	RESOLVED		sw3c C6509 5WS-X6148A	sw3c C6509 5WS-X6148A has been restarted
2018-01-08 13:08:13		Warning	2018-01-08 13:09:13	RESOLVED		sw3c C6509 5WS-X6148A	No SNMP data collection
2018-01-08 13:07:23		Warning	2018-01-08 13:15:38	RESOLVED		sw4d C6509 5WS-X6148A	sw4d C6509 5WS-X6148A has been restarted
2018-01-08 12:56:08		High	2018-01-08 13:07:08	RESOLVED		sw4d C6509 5WS-X6148A	Unavailable by ICMP ping

Obrázek 25 - Přehled výskytu problémů (Zdroj: Autor)

4.18.3 Hodnoty sledovaných zařízení



Host	Name ▾	Last check	Last value
sw1c C6509 6WS-X614...	**CPU** (2 Items)		
	#2: CPU utilization	2018-03-08 11:03:05	8 %
	#1: CPU utilization	2018-03-08 11:03:05	6 %
sw1d C6509 5WS-X61...	**CPU** (2 Items)		
	#2: CPU utilization	2018-03-08 11:03:06	15 %
	#1: CPU utilization	2018-03-08 11:03:06	4 %
sw3c C6509 5WS-X614...	**CPU** (2 Items)		
	#2: CPU utilization	2018-03-08 11:03:07	7 %
	#1: CPU utilization	2018-03-08 11:03:07	4 %
sw4d C6509 5WS-X61...	**CPU** (2 Items)		
	#2: CPU utilization	2018-03-08 11:03:08	16 %
	#1: CPU utilization	2018-03-08 11:03:08	4 %

Obrázek 26 - Hodnoty sledovaných zařízení (Zdroj: Autor)

Na Obrázek 26 je uveden výběr ze sledovaných zařízení. Tento konkrétní výběr lze ve webovém rozhraní zobrazit přes položky hlavní nabídky *Monitoring – Latest data – Host Groups – Cisco swiche.* Tímto je zobrazen přehled o všech zařízení ze skupiny Cisco switche. V přehledu je uveden nejprve název sledovaného zařízení, poté název sledované položky, čas posledního získání dat a poslední získaná hodnota.

4.19 Aktualizace na novou verzi

V průběhu provozu Zabbixu bylo vydáno několik nových verzí, které poskytují opravy odhalených problémů v dohledovém systému. Prvotní instalace, která byla od nasazení dohledového systému Zabbix využívána, je verze Zabbix 3.4.4. Za dobu provozu bylo vydáno několik aktualizací. V době dokončování této diplomové práce se jedná o verzi 3.4.7, což je v současnosti nejnovější dostupná stabilní verze. V zájmu bezpečnosti a spolehlivosti dohledového systému je doporučováno udržovat dohledový systém aktuální. Z tohoto důvodu byla provedena aktualizace na novější verzi, která je popsána v této kapitole.

Před samotnou aktualizací je vytvořen snapshot (záloha aktuálního stavu serveru) serveru zabbix-frontend v systému vCenter, který slouží ke správě virtuálních serverů VMware. Tím je zajištěna možnost vrátit se k předchozí konfiguraci serveru v případě výskytu problémů při aktualizaci.

Pro zálohu konfiguračních souborů Zabbix serveru a php serveru apache2, na kterém je provozováno webové prostředí, jsou postupně provedeny následující příkazy:

```
shell> mkdir /opt/zabbix-backup/
shell> cp /etc/zabbix/zabbix_server.conf /opt/zabbix-backup/
shell> cp /etc/apache2/conf-enabled/zabbix.conf /opt/zabbix-
backup/
```

Vykonání předchozích příkazů provede kopírování zmiňovaných souborů do adresáře `/opt/zabbix-backup/`. Obdobně je provedena záloha php souborů a Zabbix binárních souborů s využitím následujících příkazů:

```
shell> cp -R /usr/share/zabbix/ /opt/zabbix-backup/
shell> cp -R /usr/share/doc/zabbix-* /opt/zabbix-backup/
```

Pro přípravu instalace z nových instalačních balíčků je nejdříve provedeno mazání starých balíčků s příkazem `rm –Rf`.

```
shell> rm -Rf /etc/apt/sources.list.d/zabbix.list
```

Příkazem `wget` a `dpkg` je provedeno stažení nových instalačních balíčků ze serveru Zabbixu a jejich instalace.

```
shell>
wget http://repo.zabbix.com/zabbix/3.4/debian/pool/main/z/zab
bix-release/zabbix-release_3.4-1+stretch_all.deb
shell> dpkg -i zabbix-release_3.4-1+stretch_all.deb
```

Protože se nejedná o aktualizaci s přechodem na řádově vyšší verzi, není potřeba zastavovat Zabbix server a aktualizace je provedena za běhu s použitím následujícího příkazu:

```
shell> apt-get install --only-upgrade zabbix-frontend-php
```

4.20 Instalace české lokalizace

Na virtuálním serveru *zabbix-frontend* byla od počátku provozu systému nainstalována pouze lokalizace v anglickém jazyce. Pro uživatele, kteří nedisponují znalostí anglického jazyka, je na serveru provedena instalace české lokalizace pro jejich snazší orientaci v systému. Je tak poskytnut například český formát data, peněžních údajů či podpora českých znaků.

Před instalací je vytvořen snapshot serveru *zabbix-frontend* pro případ výskytu problémů při instalaci. Následně je příkazem `nano /etc/locale.gen` spuštěna editace souboru `/etc/locale.gen`, ve kterém je přidán řádek `cs_CZ.UTF-8 UTF-8`. Následně je takto upravený soubor uložen a příkazem `locale-gen cs-CZ.utf8` je instalována česká lokalizace. Úspěšnou instalaci lze ověřit příkazem `locale –a`, který vypíše aktuálně instalované lokalizace systému.

Na závěr je následujícími příkazy proveden restart služby *apache2* a *zabbix-serveru* pro aplikování změn.

```
shell> service apache2 restart
shell> service zabbix-server restart
```

Přechod na nově instalovanou lokalizaci dohledového systému Zabbix je proveden změnou hodnoty pole *Language* u konkrétního uživatele dostupného přes záložky *Administration – Users* ve webovém rozhraní dohledového systému Zabbix.

5 Výsledky a diskuse

V této práci bylo dosaženo výsledků, jejichž dopad na společnost je diskutován v podkapitole uvedené níže. Jsou také zmíněna možná vylepšení do budoucna. Zpracování navržených vylepšení se prozatím z kapacitních důvodů nepodařilo realizovat.

5.1 Dosažené výsledky

Na základě zprovoznění dohledového systému je zjištěno několik nedostatků a rizik ve společnosti. Jako příklad jednoho z takovýchto nedostatků lze zmínit problém s vysokým zahříváním switche či nadměrné vytěžování CPU switche.

Vysoké zahřívání bylo způsobeno zanesením větracích otvorů switche prachem. Teplota dosahovala 56°C, přičemž teplota, kdy dojde ke spuštění varování je 60°C. Vysoká teplota byla zpozorována před hlášením problému, a tak se podařilo předejít možným komplikacím před jejich vznikem. Po následném vyčistění switche klesla teplota o 26°C na konečných 30°C.

Problém s vysokým vytěžováním CPU se projevoval každý den vždy v konkrétní čas a následně způsoboval nedostupnost SNMP protokolu. Na základě e-mailové notifikace z dohledového systému Zabbix bylo zjištěno nadměrné vytěžování switche, které bylo zapříčiněno spouštěním procesu monitorovacího systému Cisco Prime. Po analýze procesu bylo zjištěno, že switch obsahoval chybný obsah paměti, což způsobilo opakované vytváření mnoha log souborů a následnou nedostupnost SNMP služeb na zařízení. Problém byl vyřešen naformátováním zmiňované paměti switche, který odstranil chybný obsah.

Mezi dalšími dosaženými výsledky lze zmínit například problém, který vznikl v důsledku stále narůstajícího počtu sledovaných položek. V prvních 2 měsících provozu Zabbix obsahoval přibližně 150 sledovaných položek. Tento počet v průběhu následného provozu vzrostl až na celkový počet 1512 sledovaných položek. V dohledovém systému se začalo projevovat zvýšené vytěžování procesu obstarávající získávání nových hodnot. Proces byl stabilně vytížen na 60% a ve špičkách dosahovalo vytížení až 80%. Tento problém byl vyřešen úpravou konfiguračního souboru Zabbixu, ve kterém byl zvýšen počet běžících procesů (z 1 na 5) k získávání hodnot ze sledovaných zařízení. Tím bylo dosaženo snížení stabilního vytížení z 60% na přibližně 9%. Přičemž počet procesů lze zvýšit až na

maximální hodnotu 1000 současně běžících procesů. Zabbix tedy v tomto ohledu stále poskytuje dostatečnou rezervu pro další nárůst sledovaných položek.

Vytížení CPU virtuálního serveru, na kterém je frontend Zabbixu provozován, se pohybuje okolo 2%. Je zde tedy stále dostatek hardwarového výkonu pro další nárůst počtu sledovaných zařízení. Výhodou provozu Zabbix serveru ve virtuálním prostředí VMware je jeho možné rozšíření počtu CPU jader, paměti RAM a úložného prostoru bez nutnosti restartu serveru. U fyzického serveru by toto rozšíření nebylo možné. Jednou z výhod je také možnost vytvoření obrazu disku z provozovaného serveru a jeho následný přesun na jiný VMware cluster. Pro úpravu konfigurací serveru je také velkou výhodou možnost vytváření snapshotů, které je možné vytvořit před každou změnou konfigurace, a v případě vzniklých problémů je možné se vrátit k předchozímu stavu konfigurace bez omezení funkčnosti serveru.

5.2 Navrhované změny

V době vypracování této práce probíhal veškerý monitoring realizovaný protokolem SNMP v jeho druhé verzi, která poskytuje zabezpečení pouze ve formě community stringu. Do budoucna je plánovaný přechod na monitoring protokolem ve třetí verzi, kde je již dostupné šifrování přenášených informací. V době vypracovávání práce nebylo ve společnosti autorovi umožněno realizovat přechod na tuto verzi protokolu z důvodu nedostatečných oprávnění ke změně konfigurace switche.

Pro vylepšení přehlednosti monitoringu a rychlejší orientaci při výskytu problému je vhodné vytvořit mapy, ve kterých jsou zobrazena patra jednotlivých budov společnosti a v nich zapojená sledovaná zařízení. Využití této funkce je vhodné zejména pro získání okamžitého přehledu nebo v případě předávání dohledového systému novému správci pro jeho rychlejší orientaci. Nástroj pro tvorbu map v Zabbixu však není příliš intuitivní ani uživatelsky přívětivý. Z tohoto důvodu je společností upřednostněno zahrnutí nových kritických prvků sítě do monitoringu, který společnosti poskytuje větší výhody, před tvorbou map. V okamžiku ustálení počtu monitorovaných prvků je vhodné tvořit mapy z důvodů uvedených na začátku tohoto odstavce.

6 Závěr

V praktické části této práce bylo provedeno srovnání vybraných open source dohledových systémů. K vyhodnocení byla použita vícekriteriální analýza variant metodou AHP. Na základě výsledků byl jako nejlepší kompromisní varianta vybrán dohledový systém Zabbix, který dosáhl nejlepšího výsledku především kvůli vysokému hodnocení u kritérií VMware, kde jako jediný poskytuje funkci automatického zjišťování zařízení, a kritéria databáze, kde nabízí nejširší základnu použitelných typů databází.

Následně byla vypracována analýza současného stavu monitoringu ve společnosti, na jejímž základě byly stanoveny kritické prvky pro monitoring. V další kapitole byl určen způsob monitoringu jednotlivých prvků. U většiny sledovaných zařízení bylo k monitoringu využíváno protokolu SNMP.

Dále byly navrženy potřebné parametry pro servery na základě požadavků společnosti a požadavků stanovených v dokumentaci systému Zabbix. Samotný systém byl nainstalován na virtuálních serverech na platformě VMware. Byla provedena instalace a konfigurace virtuálních serverů, instalace Zabbix serveru s frontendem na prvním virtuálním serveru a instalace databáze MariaDB na druhém virtuálním serveru. Protože je databáze provozována na odděleném serveru od Zabbix serveru, bylo zajištěno zabezpečení vzájemné komunikace na základě implementace PSK, které poskytuje šifrování.

V práci byla dále popsána instalace linuxových agentů ke sledování Zabbix serveru a databázového serveru. Byl také konfigurován systém Zabbix a postupně přidávána sledovaná zařízení, mezi která patří například VMware, Cisco switche, IBM servery, Zabbix server a databázový server.

K potřebné konfiguraci notifikací byla provedena postupně konfigurace uživatelských účtů, triggerů, akcí a nastavení e-mailových notifikací. Autentizace uživatelů byla realizována pomocí protokolu LDAPS, kde je nyní díky certifikátu zajištěno zabezpečené přihlašování uživatelů do administrace systému Zabbix na základě doménového uživatelského účtu. Při realizaci práce byla ověřena funkčnost triggerů, akcí a notifikací. V poslední kapitole vlastní části práce je popsána aktualizace systému na aktuální verzi.

Lze konstatovat, že dohledový systém je v dnešní době již nedílnou součástí infrastruktury mnoha společností. Zvlášť v případě, kdy lze využít open source variantu, za

kterou není nutné platit žádné licenční poplatky. Výsledky, které dohledový systém poskytuje, umožňují společnosti získat přehled nad stávajícím stavem sledovaných zařízení a zabránit možným problémům, a to často ještě před jejich vznikem. Díky dohledovým systémům je tak společnosti poskytnuta určitá konkurenční výhoda a je docíleno i šetření finančních prostředků, které by jinak musely být vynaloženy na nečekané opravy porouchaných zařízení či platbu za služby servisních firem.

7 Seznam použitých zdrojů

1) ADATO, Leon, 2015. *Monitoring 101: A primer to the philosophy, theory, and fundamental concepts involved in systems monitoring* [online]. SolarWinds [cit. 2017-10-06]. Dostupné z: https://thwack.solarwinds.com/servlet/JiveServlet/downloadBody/187523-102-2-28123/1510_SWI_monitoring-101-eBook_20151211.pdf
2) BEZPALEC, Pavel, 2016. Nové trendy v elektronických komunikacích Management ICT systémů. In: *Publi.cz* [online]. České vysoké učení technické v Praze [cit. 2017-10-03]. Dostupné z: https://publi.cz/books/242/03.html
3) BOUŠKA, Petr, 2006. SNMP - Simple Network Management Protocol. In: *Samuraj-cz.com* [online]. [cit. 2017-10-06]. Dostupné z: http://www.samuraj-cz.com/clanek/snmp-simple-network-management-protocol/
4) BOUŠKA, Petr, 2009. Zařízení v síti pod kontrolou. In: *Samuraj-cz.com* [online]. [cit. 2017-10-08]. Dostupné z: http://www.samuraj-cz.com/clanek/zarizeni-v-siti-pod-kontrolou/
5) DALLE VACCHE, Andrea a Stefano KEWAN LEE, 2013. *Mastering Zabbix monitor your large IT environment efficiently with Zabbix*. Birmingham: Packt Pub. ISBN 9781783283491.
6) DALLE VACCHE, Andrea a Stefano KEWAN LEE, 2015. *Zabbix Network Monitoring Essentials*. Birmingham: Packt Pub. ISBN 9781784399764.
7) GERHARDS, R., 2009. RFC 5424: The Syslog Protocol. In: *Datatracker.ietf.org* [online]. [cit. 2017-10-18]. Dostupné z: https://datatracker.ietf.org/doc/rfc5424/?include_text=1
8) ICINGA, 2018. Icinga open source monitoring. In: *Icinga.com* [online]. [cit. 2018-03-08]. Dostupné z: https://www.icinga.com/products/icinga-2/
9) KNIGHT, Joel, 2017. Why I Enthusiastically Switched from Cacti to Zabbix for System Monitoring. In: *Packetmischief.ca* [online]. [cit. 2017-10-18]. Dostupné z: https://www.packetmischief.ca/2017/02/15/why-i-enthusiastically-switched-from-cacti-to-zabbix-for-system-monitoring/
10) KOLÍSEK, Antonín, 2013. Dohledový systém Zabbix. In: *Linuxsoft.cz* [online]. [cit. 2017-10-03].

11) KOLÍSEK, Antonín, 2013. Dohledový systém Zabbix V.: pokročilé metody dohledu II. In: *Linuxsoft.cz* [online]. [cit. 2017-10-06]. Dostupné z: http://www.linuxsoft.cz/article.php?id_article=1981
12) KUROSE, James F. a Keith W. ROSS, 2013. *Computer networking: A Top-Down Approach*. 6th ed. New Jersey: Pearson. ISBN 978-0-13-285620-1.
13) MARK BURNETT. DAVE KLEIMANN, a Technical EDITOR, 2006. *Perfect passwords: selection, protection, authentication*. [Online-Ausg.]. Rockland, Mass: Syngress Publ. ISBN 15-974-9041-5.
14) MOERCH, Martin, 2016. Alerts Must Be Actionable. In: *Zabbix.tips* [online]. [cit. 2017-11-10]. Dostupné z: https://zabbix.tips/alerts-must-be-actionable/
15) MOORE, HD, 2013. A Penetration Tester's Guide to IPMI and BMCs. In: *Rapid7* [online]. [cit. 2017-10-18]. Dostupné z: https://blog.rapid7.com/2013/07/02/a-penetration-testers-guide-to-ipmi/
16) NAGIOS ENTERPRISES, 2018. Security Considerations. In: *Assets.nagios.com* [online]. [cit. 2018-02-08]. Dostupné z: https://assets.nagios.com/downloads/nagioscore/docs/nagioscore/4/en/security.html
17) NAGIOS ENTERPRISES, LLC, 2017. What is Nagios?. In: *Nagios.org* [online]. [cit. 2017-10-18]. Dostupné z: https://www.nagios.org/about/overview/
18) ODVÁRKA, Petr, 2001. ICMP: Internet Control Message Protocol. In: *Svět sítí* [online]. [cit. 2017-10-07]. Dostupné z: http://www.svetsiti.cz/clanek.asp?cid=ICMP-Internet-Control-Message-Protocol-1012001
19) POSTEL, J., 1981. *RFC 792: INTERNET CONTROL MESSAGE PROTOCOL* [online]. [cit. 2017-10-06]. Dostupné z: https://www.ietf.org/rfc/rfc792.txt
20) SOSINSKY, Barrie, 2016. *Mistrovství - počítačové sítě*. Computer Press. ISBN 9788025139165.
21) THE CACTI GROUP, INC., 2017. What is Cacti?. In: *Cacti.net* [online]. [cit. 2017-10-18]. Dostupné z: https://www.cacti.net/what_is_cacti.php
22) THE OPENNMS GROUP, 2017. OpenNMS. In: *Opennms.org* [online]. [cit. 2017-11-23]. Dostupné z: https://www.opennms.org/en/opennms
23) UBIK, Sven, 2006. *Trendy v monitorování vysokorychlostních počítačových sítí* [online]. Cesnet [cit. 2017-10-06]. ISSN 0036-9942. Dostupné z: https://www.ist-lobster.org/publications/articles/sdel_tech.pdf
24) UYTTERGIEVEN, Patrik, 2015. *Zabbix cookbook*. Birmingham: Packt Publishing Limited. ISBN 9781784397586.

25) VLADISHEV, Alexander, 2017. CVE-2016-10134. In: *CVE details* [online]. [cit. 2018-03-05]. Dostupné z: https://www.cvedetails.com/cve/CVE-2016-10134/
26) ZABBIX LLC, 2017. What is Zabbix. In: *Zabbix.com* [online]. [cit. 2017-10-18]. Dostupné z: https://www.zabbix.com/product
27) ZABBIX SIA, 2017. Zabbix Documentation 3.4. In: *Zabbix.com* [online]. [cit. 2017-10-09]. Dostupné z: https://www.zabbix.com/documentation/3.4/manual/config/items/itemtypes/zabbix_agent
28) ZABBIX SIA, 2018. Zabbix Documentation 3.4: Best practices for secure Zabbix setup. In: *Zabbix.com* [online]. [cit. 2018-02-08]. Dostupné z: https://www.zabbix.com/documentation/3.4/manual/installation/requirements/best_practices