Mendelova univerzita v Brně
Provozně ekonomická fakulta

Monitoring počítačové sítě

Bakalářská práce

Vedoucí práce:
Ing. Martin Pokorný, Ph.D.

Tomáš Papež

Brno 2014

Poděkování patří hlavně vedoucímu práce Ing. Martinu Pokornému, Ph.D. za odborné vedení a velmi přínosné rady, kterými mě vedl. Dále bych chtěl poděkovat rodině za její podporu za všech situací.

Čestné prohlášení

Prohlašuji, že jsem tuto práci: **Monitoring počítačové sítě**
vypracoval samostatně a veškeré použité prameny a informace jsou uvedeny v seznamu použité literatury. Souhlasím, aby moje práce byla zveřejněna v souladu s § 47b zákona č. 111/1998 Sb., o vysokých školách ve znění pozdějších předpisů, a v souladu s platnou *Směrnicí o zveřejňování vysokoškolských závěrečných prací.*
Jsem si vědom, že se na moji práci vztahuje zákon č. 121/2000 Sb., autorský zákon, a že Mendelova univerzita v Brně má právo na uzavření licenční smlouvy a užití této práce jako školního díla podle § 60 odst. 1 Autorského zákona.
Dále se zavazuji, že před sepsáním licenční smlouvy o využití díla jinou osobou (subjektem) si vyžádám písemné stanovisko univerzity o tom, že předmětná licenční smlouva není v rozporu s oprávněnými zájmy univerzity, a zavazuji se uhradit případný příspěvek na úhradu nákladů spojených se vznikem díla, a to až do jejich skutečné výše.

V Brně dne 3. ledna 2014 .....

Abstract

Papež, T. Computer network monitoring. Bachelor thesis. Brno, 2014.

This bachelor thesis deals with computer network monitoring, specifically monitoring from the viwepoint of availability. The theoretical part of the thesis deals with the introduction into the technological apparatus, which provides the theoretical basis for the bachelor thesis. The next point is the analysis of existing work, which specialize in the monitoring network. The practical part consists of the comprehensive analysis of user requirements Then it was selected several open source tools, in testing. They are discussed in the following chapter. Economic evaluation focuses on the tool Zabbix that was selected by testing as appropriate.

Keywords:
Computer network monitoring, Zabbix, Zenoss, Nagios, Munin, Cacti

Abstrakt

Papež, T. Monitoring počítačové sítě. Bakalářská práce. Brno, 2014.

Tato bakalářská práce se zabývá problematikou monitoringu počítačové sítě, konkrétně monitorováním z pohledu dostupnosti. První kapitola se věnuje základnímu popisu technologického aparátu, který poskytuje teoretickou základnu pro bakalářskou práci. Dále je prostor věnován analýze stávajících prací, které se zabývají tematikou monitoringu počítačové sítě. Praktická část je tvořena komplexní analýzou požadavků uživatelů, dále v testování bylo vybráno několik open source nástrojů, které jsou rozebrány v následující kapitole. Ekonomické zhodnocení se zaměřuje na nástroj Zabbix, který byl dle testování vybrán jako nejvhodnější.

Klíčová slova:
Monitoring počítačové sítě, Zabbix, Zenoss, Nagios, Munin, Cacti

Obsah

1 Úvod **6**

2 Cíl práce **7**

3 Popis technologického aparátu **8**
3.1 Jak monitorovat? . . . 9
S agentem . . . 9
Bez agenta . . . 9
3.2 SNMP . . . 10
MIB a OID . . . 10
Typy SNMP zpráv . . . 10
Verze SNMP . . . 11
3.3 NetFlow . . . 12
Proč používat NetFlow? . . . 12
Co je to IP tok? . . . 13
Architektura . . . 13

4 Analýza stávajicích prací **15**
DP – Bc. Zdeněk Habrman – Monitorování aktivních prvků počítačové sítě . . . 15
BP – Vít Pleskot – Dohledové systémy pro počítačové sítě . . . 15
DP – Kocourková Lucie – Udržitelnost a rozvoj monitorovacích systémů v komerční společnosti . . . 15
BP – Szabo Gábor – Monitoring lokální sítě prostřednictvím Nagiosu 15
BP – Dvořan Jan – Návrh a implementace monitoringu síťových linek na platformě MikroTik . . . 15
BP – Szaniszlo Tomáš – Správa heterogenních síťových prvků . . . 16
Článek – Bouška Petr – Začínáme s monitoringem sítě . . . 16

5 Analýza uživatelských požadavků **17**
5.1 Vlastní požadavky . . . 17
Sledované prvky . . . 17
Finanční a personální požadavky . . . 17
Provozní požadavky . . . 18
Vlasní monitorovací server . . . 18

6 Testování **19**
6.1 Testovací topologie . . . 19
6.2 Metodika . . . 20
Testované situace . . . 21

7 Testy jednotlivých nástrojů **23**
7.1 Zabbix 23
7.2 Zenoss 30
7.3 Nagios 33
7.4 Cacti 39
7.5 The Dude 42

8 Ekonomické zhodnocení **48**
8.1 Náklady 48
8.2 Reálný přínos 48

9 Závěr **51**

10 Literatura **52**

1 Úvod

Monitoring počítačových sítí je dnes velmi důležitou součástí většiny firem. Díky monitorování počítačové sítě se její správce dozví o problému dříve než nastane, případně v krátkém časovém intervalu. Tato skutečnost umožní řešit konkrétní problémy, namísto né vždy triviálního hledání, co nefunguje. Druhá již zmíněná výhoda je předcházení problémům, díky analýze starších dat, kterých nám monitorovaná síť poskytuje nespočet, nejčastěji ve formě grafů či vývojových diagramů lze jednoduše odhalit slabá místa, jako je nedostatečná výkonová kapacita, nestabilní zařízení/služby. Lze tak mnohem efektivněji plánovat rozvoj infrastruktury jak z krátkodobého tak dlouhodobého hlediska a předejít budoucím problémům s náhlým nedostatkem kapacit. Díky těmto skutečnostem lze tedy uspořit finanční prostředky, o které by společnost přišla na ušlém zisku, který způsobila nedostupnost poskytovaných služeb, nebo pokutách za garance dostupnosti těchto služeb třetím stranám.

Monitorovat se dá téměř cokoliv, avšak tato práce se bude zabývat monitorováním z pohledu dostupnosti. V tomto směru se využívá většinou několik standardizovaných protokolů, které výrobci implementují přímo do svých zařízení a rozebrány budou dále v práci. Hlavní roli tedy hrají veškeré síťové prvky - směrovače, přepínače, servery, datová úložiště, …

Dostupnost je dle SolarWinds (2012) schopnost sítě reagovat na požadavky o přístup k síti. Síťová dostupnost se může snížit, pokud je některé zařízení v síti využíváno více, než se předpokládalo a nemá již prostředky ke zpracování požadavků. Dostupnost je obvykle součástí SLA (*Service-Level Agreement*) stanovených pro firemní IT.

2 Cíl práce

Cílem této práce bylo seznámení se s monitorováním počítačové sítě. Na základě analýzy uživatelských požadavků byla následně vytvořena případová studie, ve které bylo otestováno několik nástrojů. Nakonec bylo vytvořeno ekonomické zhodnocení nasazení monitorovacího systému.

3 Popis technologického aparátu

Internet protocol Je definován v RFC 791(1981) a pracuje na třetí vrstvě ISO/OSI modelu, který dále rozebírá Bouška (2007). Tento protokol se stará zejména o přenos IP datagramů, přes mezilehlé uzly až k jejich konkrétnímu cíli. Poskytuje hierarchický systém adresace stanic v globální síti, který přispívá k technickému zjednodušení a praktické realizovatelnosti směrování v globální síti, umožňuje zajistit případnou segmentaci, rozdělení dlouhých paketů do kratších celků, pokud je zapotřebí je přenést lokální sítí, která nepodporuje dostatečně dlouhý datový rámec na spojové vrstvě tak, aby se do něj celý původní paket vešel bez nutnosti jeho rozdělení a je navržen tak, aby bylo možné v procesu směrování v síti provádět průběžnou kontrolu neporušenosti IP záhlaví, které obsahuje celou řadu důležitých řídících parametrů. IP protokol nekontroluje bezchybnost uživatelského datového pole, toto ponechává protokolům na vyšších vrstvách. (Boháč, 2013)

TCP UDP Tyto protokoly jsou definovány v RFC 793(1981) a RFC 768(1980). Oba protokoly předpokládají na nižší vrstvě protokol IP. UDP zprávy zajistí zasílání zpráv mezi programy s minimálními nároky na přenosovou kapacitu a je zaměřeno na rychlou komunikaci, takže doručení či duplicita nejsou garantovány. Pokud aplikace požaduje tyto vlastnosti, využívá se protokol TCP.

ICMP Jak uvádí Microsoft (2012) protokol ICMP (*Internet Control Message Protocol*) je diagnostický nástroj, který je považován za povinnou součást jakékoli použití protokolu IP. Protokol ICMP může vygenerovat několik typů zpráv, které je dobré znát pro lepší diagnostiku potíží se sítí. Echo Request a Echo Reply se nejčastěji používá k testování připojení pomocí protokolu IP, všeobecně jsou tyto zprávy známé jako PING. Všechny typy zpráv a ostatní detaily protokolu jsou definovány v RFC 792(1981).

Sflow Sflow je otevřený standard a dle sFlow.org(2013) jde o proktokol na sledování provozu v rychlých sítích a byl vyvynut jako náhrada za NetFlow, které je vyvynuto společností CiscoSystems.

SSH Secure shell je definován v RFC 4253.(2006) Tento protokol zajišťuje bezpečné přihlášení ke vzdálenému počítači přez nezabezpečenou síť, typicky internet.

WMI Windows Management Instrumentation je implementace Web-Based Enterprise Management[1],což je iniciativa zaměřena na vývoj standardní technologie pro přístup k informacím ze systému v podnikovém prostředí.(Microsoft, 2013) WMI poskytuje jednotné rozhraní pro všechny místní a vzdálené aplikace, nebo skripty, které získávají údaje pro řízení z počítačového systému či sítě.(Microsoft, 2013)

HTTP dle RFC 2611[2] pracuje tento protokol na aplikační vrstvě. Jde o základní bezstavový protokol, který může být využit nad rámec použití hypertextu.

Hardware CPU (*Central Procesor Unit*) je dle InetDaemon Enterprises(2013) jednou ze základních součástí počítače. Při monitoringu nás zajímá jeho vytížení v % a zatížení (*load*), přičemž %CPU (ideálně pro jednotlivá jádra) řek-

[1] http://dmtf.org/standards/wbem
[2] http://tools.ietf.org/html/rfc2611

nou kolik procent času se strávilo skutečnou prací. Load vypovídá o tom, kolik bylo procesů vyžadujících procesorový čas.(Klement, 2009)

HDD (*Hard Disk*), neboli pevný disk je dle InetDaemon Enterprises(2013) zařízení sloužící k ukládání dl informací. U monitorování využíváme údaje o jeho zaplnění v %, přičemž je důležité nastavit správné limity dle velikosti disku. Druhý měřený ukazatel je počet IOPS (*Input/Output Operations Per Second*), který je u každého disku jiný a je potřeba vědět vlasní limit.(Wikipedie, 2013)

RAM Random Access Memory slouží k ukládání informací pro data a programy, se kterými se právě pracuje a která jsou právě potřeba.(InetDaemon Enterprises, 2013) Výhodou oproti přímému čtení z HDD je rychlost čtení a krátká přístupová doba. U monitoringu pamětí se podobně jako u disku díváme hlavně na jejich využití v %, vůči jejich absolutní velikosti.

Aplikační služby V práci se monitoruje několik služeb:

- Webserver - Apache/IIS
- Databáze - MySQL/MSSQL
- Mailserver - Postfix/Exchange
- VPN - host to site - OpenVPN / site to site IPSec
- Active Directory - adresářová služba
- DHCP - ISC-DHCP/MS DHCP
- DNS - BIND9/MS DNS

3.1 Jak monitorovat?

S agentem

Typicky program, běžící na straně monitorovaného zařízení. Jak uvádí Bouška(2009), agent musí být vytvořený pro daný operační systém, musíme mít možnost instalovat aplikace na server, čímž nám přibyde další služba, která může způsobovat nečekané potíže se stabilitou celého systému, avšak získáme mocný nástroj pro jednoduché získávání informací ze serveru.

Bez agenta

Většinou se používá základních požadavků různých protokolů k ověření jeho funkčnosti. V případě www serveru je to například HTTP[3] požadavek na server, který v případě, že vrátí kód 200[4], můžeme předpokládat, že je vše v pořádku. Na podobném principu funguje i kontrola ostatních služeb.

[3]http://www.w3.org/Protocols/rfc2616/rfc2616.html
[4]http://www.w3.org/Protocols/rfc2616/rfc2616-sec10.html

3.2 SNMP

Jak píše Kretchmar(2004) ve své knize, tak SNMP (*Simple Network Management Protokol*), je relativně jednoduchý protokol, sloužící ke správě zařízení v síti.

Leskiw(2012) uvádí, že nám SNMP poskytne:

- Schopnost číst či zapisovat do zařízení.
- Shromažďovat informace o využití šířky pásma.
- Sběr chybových zpráv a jejich ukládaní do logů.
- Monitorování využití prostředků – CPU,RAM,DISK, TEPLOTA,…
- Aktivní dohled – dostupnost stanic.
- Pasivní dohled – zařízení samo pošle oznámení o chybě.

Dle RFC1157(1990) a Bouška(2006), SNMP vyžaduje při komunikaci dvě strany a to správce (manager) a agenta. SNMP pracuje ve dvou režimech činnosti:

- Správce posílá dotazy agentovi a přijímá odpovědi – využívá se k prohlížení stavu, analýze a správě síťových zařízení. Správce může sbírat data z několika agentů, většinou tyto data následně vizualizuje, do grafické podoby, čímž se zvýší celková přehlednost v případě, že obstarává desítky či stovky agentů.
- Agent - jde v podstatě o program, který běží na straně monitorovaného zařízení a shromažďuje informace o zařízení, které poskytuje správci a zasílá oznámení (trapy) na adresu správce v případě překročení některé hraniční hodnoty.

Dle Maura a Schmidta(2005) pracuje SNMP na protokolu UDP, přičemž na portu 161 jsou přijímány a odesílány žádosti a port 162 slouží pro odesílání trapů, které jsou rozebrány dále v práci.

MIB a OID

Každý agent musí obsahovat vlastní databázi MIB (*Management Information Base*), což je stromová struktura, jejíž každý uzel je jedinečné číslo, označované jako OID, které je dále vyžadováno protokolem SNMP. Textové názvy těchto uzlů jsou jak uvádí Kretchamar(2004) určeny pouze pro zvýšení čitelnosti, takže název .iso.org.dod.internet.mgmt.mib-2.system.syUpTime bude v paketu vypadat jako .1.3.6.1.2.1.1.3 a vrací tedy hodnotu informující o délce běhu systému v sekundách.

Typy SNMP zpráv

Podle (ManageEngine, 2012) a (Microsoft, 2012) rozlišujeme několik typů základních SNMP příkazů, jejichž směr je uveden na obrázku **??**.

- GET: jedná se o základní požadavek manažera na určité OID, na který agent odpovídá zprávou RESPONSE s hodnotou OID.
- GET-NEXT: rozšířený typ požadavku, díky kterému lze prohlížet celý strom OID. Využívá se pro vypisování dynamických tabulek.
- SET: v případě, že má agent práva zápisu, může pomocí této zprávy aktualizovat hodnotu, např. zapnout rozhraní.
- GETBULK: využívá se v případě, kdy manažer vyžaduje několik OID najednou, aby byl minimalizován počet výměn protokolu.
- TRAP: zpráva, kterou agent zasílá na adresu managera v případě překročení některé hodnoty či chyby.
- INFORM: má stejnou funkci jako trap, ale obsahuje navíc ověřovací mechanismus, že manager zprávu přijal.

Verze SNMP

SNMP má v současné době tři verze. První SNMPv1 je definováno v RFC 1157(1990), kde je popsána celá architektura SNMP a typy zpráv. V první verzi neexistovalo žádné zabezpečení, jediný ověřovací mechanismus představovaly community string(*CS*), což je obdoba skupin, měly právo číst, zapisovat nebo obojí. Název této CS nebyl nijak šifrovaný, pokud se útočník dostal tak blízko, že byl schopen odchytávat pakety, již pro něj nebyl problém dle názvu CS zjistit její práva a napadnout zařízení, pokud nemělo implementovaný dodatečný ochranný mechanismus.

SNMPv2 je popsáno v RFC 1901 – RFC 1908 (*community based*) a následně v RFC 1909–RFC 1910 (*user based*), odstraňovalo hlavní nedostatky první verze, avšak dle Bigelowa(2004) nebyla dobře přijata množstvím softwarových a hardwarových dodavatelů, díky složité implementovatelnosti zabezpečení. Poté byla tato verze upravena komunitou na SNMPv2c, jež upravila zabezpečení stejně jako ve verzi jedna.

SNMPv3 je popsáno v RFC 3411 – RFC 3418. Hlavní výhodou této verze je přidání zabezpečovacích funkcí. Konkrétně jde dle Burdy(2009) o autentizaci, takže se správce zařízení přihlašuje pomocí svého jména a hesla, které je šifrované v hash podobě, pomocí mechanismu MD5, popřípadě novějšího SHA1 a veškerá komunikace je šifrována protokolem DES, který byl však vyvynut v 70. letech a dnes je snadno prolomitelný. Poslední významnou novinkou bylo přidání ACL(*Access Control List*), takže můžeme různým správcům nastavit jiná oprávnění přístupu. Možnosti zabezpečení lze kombinovat, takže můžeme nastavit přístup pouze pomocí jména a hesla, popřípadě pouze šifrovat komunikaci. Pokud jsou nastaveny všechny zabezpečovací prvky a útočníkovi se povede odchytit pakety, tak není na rozdíl od verze jedna a dvě schopen získat přístup k zařízení.

3.3 NetFlow

Jak se píše na Wikipedii(2012) je NetFlow otevřený protokol vyvinutý společností Cisco Systems, určený původně jako doplňková služba k Cisco směrovačům, přičemž v dnešní době tento protokol využívá většina síťových zařízení. Dle Cisco Systems(2007) se NetFlow obvykle používá na centrálních místech sítě, kterými protéká největší množství dat, typicky směrovač na perimetru, popřípadě v novějších architekturách sítě se používají pasivní NetFlow sondy. Veškerá tyto zařízení odesílají informace o tzv. IP tocích (IP Flow), které mohou být prohlíženy přímo na zařízení, popřípadě odesílána na vzdálený server. Nároky tohoto protokolu jsou zhruba 1-5% šířky pásma v přepínaných sítích, což znamená, že v sítích s rychlostí 100Mb a reálným vytížením 70Mb protokol spotřebuje zhruba 0.7-3.5Mb, čímž celkové vytížení vzroste a je třeba s tím při plánování počítat.

Proč používat NetFlow?

Díky NetFlow můžeme získat podrobné informace o provozu na naší síti, nejen využití linky, jak je tomu v případě využití samotného SNMP protokolu. Dle Cisco Systems(2007) usnadňuje řešení běžných problémů:

- Nové aplikace v síti a jejich dopad – identifikuje dopad na zatížení sítě novou technologií, jako je VoIP, VPN,...
- Snížení provozu ve špičce – v případě již nedostatečné kapacity linky můžeme díky NetFlow analyzovat toky a zjistit, která aplikace zbytečně využívají v tuto dobu linku a naplánovat jejich spouštění mimo špičku, popřípadě tyto aplikace úplně zakázat a vyhnout se tak nepotřebným rozšířením sítě, díky zbytečným aplikacím.
- Najít slabá místa v síti – při dlouhodobém sběru informací o tocích můžeme následně určit krizová místa v síti a tyto místa vylepšit, aby nedošlo k zahlcení.
- Bezpečnost a detekce anomálií – analýzu toků používá k detekci neoprávněného provozu a diagnózám sítě většina SIEM (*Security Information and Event Management*) řešení, jako je Cisco-MARS[5], Alien Vault OSSIM[6], Cyberroam iView[7].
- Kontrola Qos parametrů – ověření, zda byla přidělaná správná šířka pásma určité třídě (COS).

[5]http://www.cisco.com/en/US/products/ps6241/
[6]http://www.alienvault.com/open-threat-exchange
[7]http://www.cyberoam-iview.org/

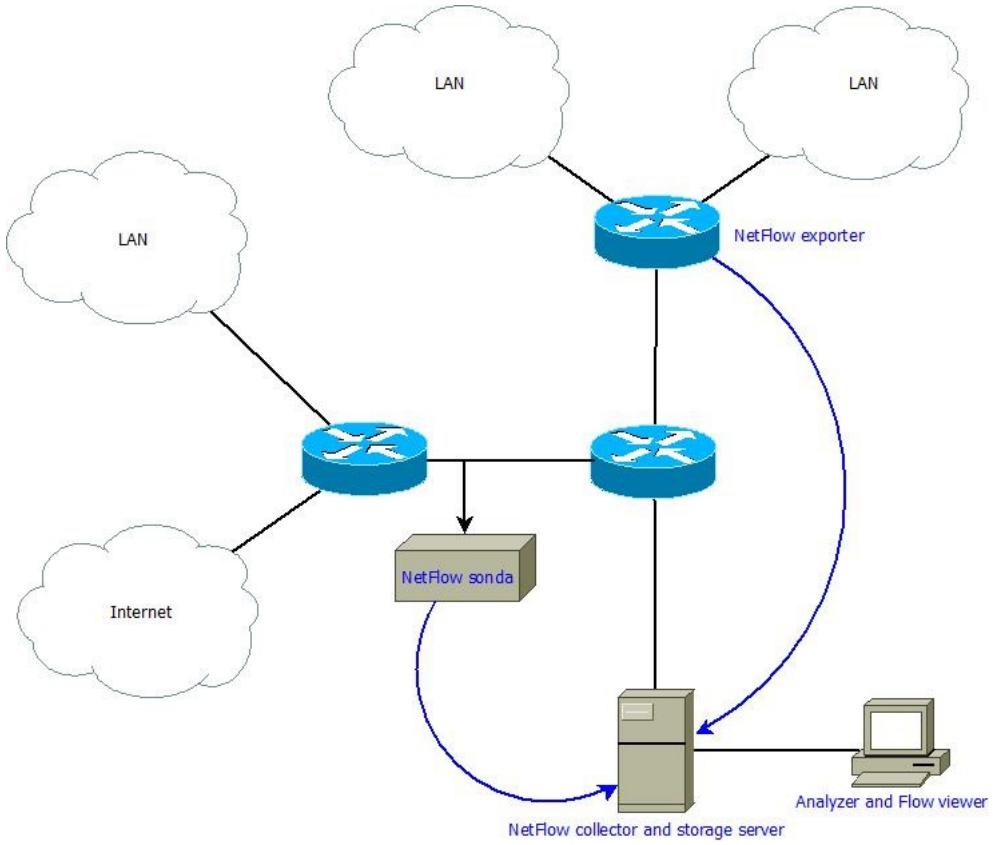
Co je to IP tok?

IP tok je vlastně seskupení jedinečných paketů, které jsou zkoumány na svou jedinečnost. Jak uvádí Cisco Systems(2007) je to většinou 5-7 atributů, přičemž v případě, že se některé údaje shodují, tak jsou tyto pakety seskupeny do proudů a pakety a bity se sčítají. Zkoumané atributy:

- Zdrojová IP adresa
- Cílová IP adresa
- Zdrojový port
- Cílový port
- Typ L3 protokolu
- Class of service (IEE P802.1p)
- Rozhraní

Architektura

Hlavním prvkem architektury je NetFlow exporter, který odesílá data na server, na kterém se data ukládají a následně se analyzují a vizualizují. Exportérem může být samostatný směrovač, popřípadě sonda. Ve většině případů se používají externí sondy, protože dle Wikipedie(2012) ubírá výpočet statistik směrovačům výkon a kvůli tomu používají vzorkování, což znamená, že pro výpočet se používá pouze každý n-tý paket. Díky vzorkování tedy ztrácíme přesnost a hlavně můžeme přehlédnut některé bezpečnostní hrozby. Naproti tomu sondy jsou pasivní zařízení, takže vůbec nezasahují do dat a na collector odesílají data přes dedikovanou linku, čímž se stávají v síti téměř neviditelné a jsou proto obtížně napadnutelné.



Obrázek 1: Architektura Netflow

4 Analýza stávajicích prací

DP – Bc. Zdeněk Habrman – Monitorování aktivních prvků počítačové sítě

Habrman(2013) se ve své práci zabývá možnostmi a principy dohledových systémů. Analyzuje systémy používané v dnešní době a to hlavně ty, které primárně využívaj protokoly SNMP a ICMP. Nasazuje systém Cacti pod operačním systémem MS Windows Server 2008 R2 na univerzitní počítačovou síť.

BP – Vít Pleskot – Dohledové systémy pro počítačové sítě

Pleskot(2013) v bakalářská práci analyzuje možnosti monitoringu a managementu počítačových sítí. V práci jsou představeny principy monitoringu i open source monitorovací systémy. Výsledkem teoretické části je zvolení vhodného monitorovacího systému na základě několika základních kritérií, jako je podpora webového rozhraní, SNMP a emailových varování. Následně je v reálném prostředí vybrané společnosti nakonfigurován nástroj OpsView[8].

DP – Kocourková Lucie – Udržitelnost a rozvoj monitorovacích systémů v komerční společnosti

Kocourková(2012) měla ve své práci popsat existující open source nástroje pro monitorování síťových systémů a posoudit je z pohledu udržitelnosti a jejich budoucího rozvoje. Při porovnání se zaměřila na nekomerční nástroje typu Nagios, Zabbix apod. Následně tyto nástroje porovnala s interně vyvinutým systémem pro monitorování ve vybrané softwarové společnosti. Na základě kritérií společnosti provedla analýzu, ze které vzešlo, že interne vyvinutý systém je pro danou společnost dostačující, přičemž navrhla pouze pár vylepšení tohoto systému, které vzešly z provedeného porovnání. V poslední části práce se věnuje projektové dokumentaci nasazení monitorovacího nástroje.

BP – Szabo Gábor – Monitoring lokální sítě prostřednictvím Nagiosu

Szabo(2013) ve své práci nejprve popisuje obecné principy managementu sítě a následně se zaobírá pouze systémem Nagios, pro který byl následně vytvořen modul složící ke správě tiskárny. Výsledný modul je implementován do reálného provozu a následně je ještě diskutováno několik dalších pluginů pro Nagios.

BP – Dvořan Jan – Návrh a implementace monitoringu síťových linek na platformě MikroTik

Dvořan(2012) se zabývá návrhem řešení a následné implementaci monitoringu rychlosti využití pronajatých linek koncovými zákazníky poskytovatele internetu, kdy

[8]"http://www.opsview.com/?currency=USD

každý zákazník se připojuje k síti pomocí protokolu PPPoE k zařízení na platformě MikroTik. V této práci je uvedeno shrnutí stávajících produktů na trhu a postup při návrhu a vytvoření nové webové aplikace, která monitoruje jednotlivé klienty sítě a poskytuje jim možnost zobrazení grafu historie rychlosti jejich síťové linky. Vytvořená aplikace sbírá data pomocí protokolu SNMP ze zařízení MikroTik a následně využívá program RRD Tool pro uchování získaných dat a tvorbu grafů reprezentující historii využití pronajaté linky.

BP – Szaniszlo Tomáš – Správa heterogenních síťových prvků

Szaniszlo(2012) řeší správu a dohledem síťových prvků v univerzitní sítě Masarykovy Univerzity. Práce detailně rozebírá možnosti řešení v této oblasti a popisuje běžně používané protokoly a nástroje. Dále je zde popsána implemetace systému správy síťovýh zařízení, nasaná studentem v jazyce Perl na FI MU, přičemž je následně tento systém vylepšen o některé fuknce.

Článek – Bouška Petr – Začínáme s monitoringem sítě

Bouška(2009)se ve svém článku zaměřil na základy ohledně monitorování sítě. Jeho článek ve stručnosti ukazuje možnosti monitorování. Detailněji rozebírá technologie, oblasti a výstupy monitoringu.

5 Analýza uživatelských požadavků

Vytvoření kvalitního dohledového systému musí předcházet důkladná analýza požadavků. Prostor pro vypracování praktické části této práce jsem dostal v komerční společnosti zabývající se provozem vlastních serverů, promírně určených na webhosting náročných aplikací a sekundární činnost je outsorcing správy IT pro menší a střední společnosti, takže řešení a požadavky jsou navrženy na základě reálných požadavků.

5.1 Vlastní požadavky

Sledované prvky

Hlavními prvky, které je potřeba monitorovat jsou:

- HW a virtuálními stroje na platformě Linux. Základem je sledovat využití zdrojů na serverech a síťovch prvcích a stav procesů pro základní služby - webserver (*Apache*), mailserver (*Postfix*), databázový server (*PostgreSQL*). Veškeré tyto služby je potřeba sledovat minimálně na úrovni stavu služby, ideálně však i odpovědi na dotazy HTTP či SQL. Neméně důležité je též dohled nad hlavním směrovačem MikroTik.
- HW a virtuální stroje na platformě Microsoft.Windows Server 2008 až 2012 R2. Mezi základní potřeby dohledu patří opět stav serveru z hlediska dostupnosti a následně stav jeho služeb, což představuje hlavně MS Exchange 2010 do co největší hloubky, VPN - site to site a host to site. Z ostatních služeb poskytovaných přímo rolemi OS je potřeba hlídat funkčnost AD (*Active Directory*), RRAS(*Routing and Remote Acess*), DHCP,DNS.
- Síťové prvky a dohled na jejich funkčnost. Typicky směrovač, nebo přepínač hlídaný z hlediska dostupnosti a zdrojů, zejména využití šířky pásma, CPU a RAM.

Pro dohled na servery, na platformě Linux i Windows by mělo být ideálně využito agenta, u zařízení bez možnosti instalace agenta bude využit protokol SNMP, případně NetFlow a u některých pouze PING, nebo ostatní základní protokoly. Dle zadání nebude řešené monitorování SYSLOG [9] informací, jelikož tuto fuknci řeší samostatný nástroj.

Finanční a personální požadavky

Vedení společnosti rozhodlo, že do monitorování chce vložit co nejméně prostředků, proto bude zvolena cesta open-source. Na nasazení nástroje je vyhrazeno celkem

[9]"http://tools.ietf.org/html/rfc5424

20 pracovních dní. V první fázi bude proveden návrh, který bude zahrnovat seznam všech zařízení a budou zde vyznačeny ty, které se budou monitorovat v prvním kroku a ty, které se přidají až po důkladném otestování, přičemž jako první budou vybrány méně kritické prvky, aby případná nestabilita neohrozila celkový provoz. Zároveň v první fázi bude u každého serveru určeno které služby a vlastnosti prvků se budou monitorovat. Ve druhé fázi bude na zakoupený virtuální server nainstalovaný zvolený nástroj a proběhne konfigurace prvků. V poslední fázi bude nástroj několikrát denně sledovaný z hlediska komunikace, funkčnosti, nároků na HW a stability. V poslední fázi po odladění chyb, které by mohly vzniknout bude monitorování nasazeno i na kritické prvky, přičemž tato fáze neproběhne dříve, než za 90 dní od spuštění testovacího provozu, avšak měla by být hotova nejpozději do šesti měsíců.

Provozní požadavky

Zvolený nástroj musí poskytovat přehledné grafické rozhraní skrze webový prohlížeč a to 24/7. Dále musí mít jednoduché přidání či odebrání monitorovaných zařízení, snadnou modifikaci již monitorovaných systémů. V případě problémů musí nástroj odeslat email se stručným popisem do tiketovacího systému. Tento popis by měl obsahovat informaci o tom, na jakém zařízení je problém, od kdy se problém vyskytuje, jeho prioritu a v čem problém spočívá. Takovýto email by tedy měl vypadat například: 10.12.2013 10:00:00 Server1, kritický problém, služba DHCP není spuštěna.

Zvolený nástroj musí uchovávat záznamy minimálně po dobu šesti měsíců pro budoucí analýzu. Z těchto údajů musí jednoduše tvořit přehledné výstupy v grafické i textové podobě.

Vlasní monitorovací server

Pro účely monitoringu bude zakoupen virtuální server disponující dvěmi jádry procesoru Intel Xeon na frekvenci 2.4 GHz, 2 GB RAM, diskem s variabilní kapacitou 100GB, přičemž 15GB bude učeno pro operační systém, v tomto případě Linux Debian a připojen do internetu bude pomocí neagregované 100 Mbps linky.

6 Testování

V rozsahu bakalářské práce není možné pojmou všechny monitorovací nástroje, o nejlepších se zmiňuje například Siliconindia(2012), TheGeekStugg(2012), případně téměř všechny, včetně výčtu základních vlastností jsou shrnuty na Wikipedii(2012). Při výběru monitorovacích nástrojů vycházela práce z průzkumu mezi IT profesionály s několikaletou praxí. V průzkumu byli zastoupeny všechny primární pozice a to správce serverů na různých plaformách, správce sítě ve středně velkých společnostech a síťový specialisté. Anketa byla prováděna osobním dotazováním, případně pomocí sdílení mezi kolegy, díky čemuž jsou výsledky relevantnější, jak podobný průzkum prováděný na internetu, kde by měl možnost do něj zasáhnout téměř kdokoliv. Kompletní sadu otázek včetně odpovědí je v příloze č.1.

Z tohoto průzkumu vzešlo několik nejpoužívanějších open-source nástrojů, přičemž pro porovnání bylo vybráno pět nejčastějších z nich, které splňují veškeré požadavky, které vzešly z analýzy. Jako šestý nástroj jsem chtěl zvolit Centreon, jelikož je to v dnešní době nejprogresivnější nástroj, ale jeho jádro je v podstatě Nagios, proto jsem od samotného testování upustil, avšak jeho pozice se zvyšuke a byla by škoda ho alepoň nezmínit.

Vybráno tedy bylo následujících 5 open source nástrojů:

- Zabbix
- Nagios
- The Dude
- Cacti
- Zenoss

6.1 Testovací topologie

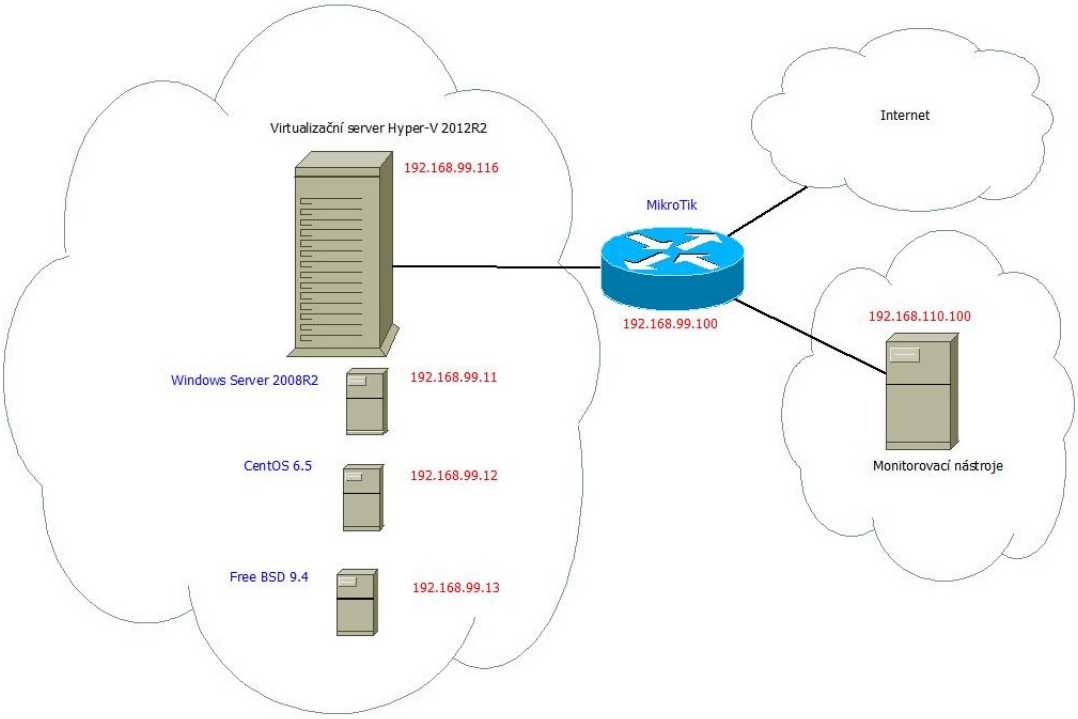
Na základě výše provedené analýzy byl vytvořen zjednodušený model topologie, který obsahuje veškeré prvky z analýzy a to operační systém Linux/Windows a jako směrovač byl použit MikroTik.

K testování byl využit základní server, který disponuje čtyřjádrovým procesorem Intel Xeon s taktem 2.4 GHz na jádro, 8 GB RAM a 500 GB S-ATA + 76 GB SAS disky, s nainstalovaným MS Hyper-V 2012 R2. Díky využití virtualizace nebylo potřeba do topologie zapojovat několik fyzických serverů, ale postačil pouze jeden a ostatní stroje běžely jako virtual machine, více o virtualizaci s Hyper-V píše třeba Microsoft(2013). Výkon serveru je dostatečný a využité prostředky se nedotkly výkonostního limitu serveru, takže zkreslení výsledků lze téměř vyloučit.

Na každý virtuální stroj VM1,VM2 bylo použito 512MB RAM + 1CPU bez garance zdrojů, na VM3 bylo použito 4GB RAM + 2 CPU s garancí 2 GB RAM a 1500 MHz CPU na každé jádro.

Jako hlavní router byl použit MikroTik RB951G 2HnD, který poskytoval NetFlow a další informace pomocí SNMP, výhodou tohoto zařízení je, že poskytuje téměř všechny údaje, jak nejvyšší model.

Na prvním obrázku nalezneme kompletní adresaci celé topologie včetně operačního systému. Obrázek též určuje základní rozdělení topologie na síť s adresací 192.168.99.0/24 ve které se nachází všechny monitorované prvky, včetně Hyper-V serveru. V síti 192.168.110.0/24 se nachází pouze monitorovací nástroj, příčemž internet je připojen pouze pro účely stažení balíčků a při samotných testech nemá žádný vliv.

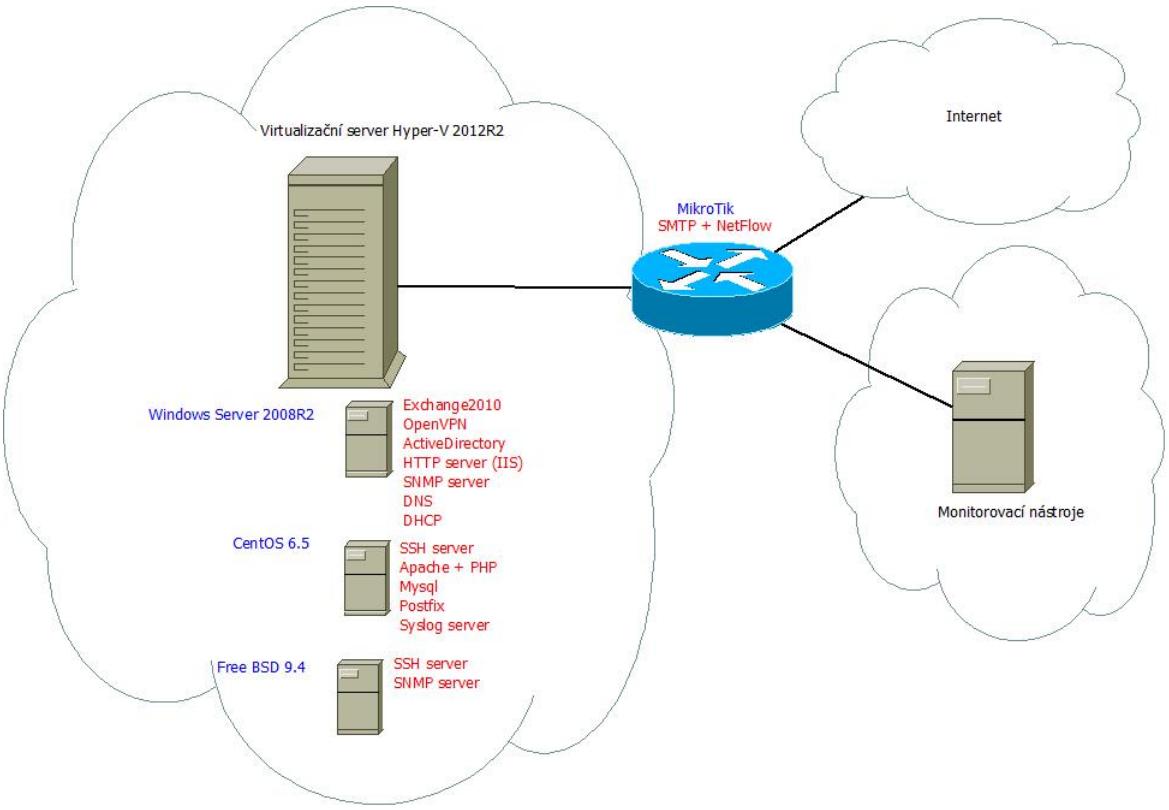


Obrázek 2: Testovací topologie - adresace

Druhý obrázek obsahuje pevně dané služby, které určité servery poskytují. Server nazvaný Monitorovací nástroje je využit vždy jen na samotný nástroj a jeho parametry jsou shodné se serverem, který bude použit v ostré verzi (*2 jádra CPU s frekvencí 2.4 GHz, 2 GB RAM a 76GB SAS disky*), hlavní rozdíl je, že není využita 100 Mbps linka, ale 1000 Mbps.

6.2 Metodika

V této kapitole se budeme zabývat testováním všech nástrojů a jejich reakcí na přesně dané události z předchozí kapitoly. Na základě těchto testů bude následně vybrán nejvíce vyhovující nástroj pro reálné nasazení v analyzované produkční síti.



Obrázek 3: Testovací topologie - rozložení služeb

Testované situace

Časový scénář výpadků: Při simulaci těchto výpadků bude nastaven všude stejný systémový čas a bude se měřit rychlost reakce na události. Pro veškeré nástroje bude nastavena stejná doba pro pravidelné dotazování na 10 sekund při dohledu na dostupnost a kritické služby a 60s pro ostatní parametry.

Dále se při hodnocení zaměříme na samotnou práci s nástrojem a to:

- Náročnost instalace a přidání nového zařízení.
- Podpora - komunitní i vývojářská.
- Možnosti rozšíření.
- Multiplatformnost nástoroje a monitorovaných zařízení.
- Stabilita a využití prostředků serveru.

Tabulka 1: Testované situace

Číslo výpadku	**Typ výpadku**	**Jak bude simulovaný**
1	Výpadek služby na serveru	služba bude ručně zastavena
2	Restart služby	vypnutí a zapnutí služby s výpadkem 30s
3	Přeplnění disku	vytvoření prázdných souborů
4	Kritické využití CPU	spuštění zátěžového testu
5	Kritická teplota	nastavení nižší hranice, dle aktuální teploty CPU bez zátěže přidání 5°C zatížení CPU testem, aby se teplota zvýšila, avšak CPU nebyl poškozen
6	Kritické využití šířky pásma do internetu	nastavení nižší hranice a spuštění několika přenosů najednou
7	Webserver - WEB hi response	přetížení serveru
8	Dlouhá fronta emailů	vytvoření několika tisíc emailů pomocí Mail Bomber 1.0[10].
9	Výpadek směrovače	Odpojení přepínače od zdroje elektrické energie
10	Výpadky spojení	Opakované zakázání a povolení interface na zařízení
11	Výpadek spoje na server	zakázání interface na směrovači MikroTik
12	Náhlé vypnutí serveru	standartní vypnutí systému
13	Restart serveru	restart serveru uživatelem

	Čas		Typ výpadku	Poznámka
1)	0:00:00		Výpadek služby na serveru	
2)	0:07:00		Restart služby	
3)	0:14:00		Přeplnění disku	
4)	0:21:00		Kritické využití CPU	
5)	0:28:00		Kritická teplota	
6)	0:35:00		Kritické využití šířky pásma do internetu	
7)	0:42:00		Webserver - WEB hi response	
8)	0:49:00		Dlouhá fronta emailů	
9)	0:56:00		Výpadek směrovače	
10)	1:03:00		Výpadky spojení	
11)	1:10:00		Výpadek spoje na server	
12)	1:17:00		Náhlé vypnutí serveru	
13)	1:24:00		Restart serveru	

Obrázek 4: Základní časová osa

7 Testy jednotlivých nástrojů

V této kapitole budou postupně popsány všechny monitorovací nástroje a jejich chování ve výše uvedených situacích. Veškeré prvky jsou vždy ve stejném základním nastavení, mění se pouze některé parametry, které jsou nutné pro chod různých monitorovacích technik, typicky nastavení SNMP či instalace různých agentů.

7.1 Zabbix

Celý systém je tvořen serverem, agentem, webovým rozhraním pro správu. Veškerá data jsou uložena v databázi, konkrétně lze použít MySQL, Oracle, PostgreSQL,SQLite, IBM BD2. Výběr databáze a celkový návrh hardware pro Zabbix záleží na množství sledovaných zařízení, počtu testů a délce historie, kterou chceme uchovávat. Pro střední podniky dostačuje jakýkoliv dvoujádrový procesor a více jak 1GB RAM. Z hlediska operačního systému je možné provozovat Zabbix na některém z unixových systému nebo libovolné distribuci GNU/Linux, případně lze Zabbix provozovat i na HP-UX, Mac OS X či Solarisu. Zabbix documentation(2013)

Dostupnost služeb

Možnost sledování běžících služeb (HTTP, SMTP, DNS) nejen z hlediska stavu procesu, ale také dostupnosti daného portu na kterém služba naslouchá. V případě výpadku je Zabbix pomocí agenta schopný spustit skripty uložené na monitorovaném stroji a obnovit tak provoz vypadlých služeb. Tyto skripty si každý musí dle svých potřeb napsat, avšak pro obyčejný restart služby na Windows postačí spustit příkaz *"net start/stop název služby"*, problém je, že musíme pro každou službu mít vytvořený nový skript, jelikož Zabbix neumí předat zpět název služby. Jediné na co si musíme dát pozor je, pod jakými oprávněními služba běží tzn. jaké může spouštět příkazy, aby v případě narušení bezpečnosti nebyl útočník skrz Zabbix schopný ovládnout celý systém.

Přehled

Zobrazuje aktuální souhrnné informace o stavu dohlížených hostů, skupin, varování o nedostupnosti a stav událostí. Zobrazované data jsou uživatelsky nastavitelné stejně jako doba aktualizace zobrazovaných dat.

Na obrázku je vidět základní přehled Zabbixu, kde se z boxu pod číslem 1 dá vyčíst údaje o počtu monitorovaných zařízení, jejich položek (*items*), aktivních spoštěčů (*triggers*), uživatelů a počtu nových hodnot za sekundu. V boxu 2 a 3 lze zjistit údaje o vytvořených skupinách, přičemž ve dvojce je rozlišení dle důležitosti a ve trojce pouze podle toho, zda má daná skupina problém či nikoliv. Z uvedeného obrázku lze tedy vyčíst, že máme 5 monitorovaných zařízení, na kterých sledujeme 287 hodnot a každou sekundu nám příjde 3.98 hodnot. Z těchto všech hodnot je na Linux serveru jedna extrrémně kritická chyba, na Windows jsou to 4 s vysokou prioritou a Zabbix server hlásí jedno varování.



Status of Zabbix

Parameter	Value	Details
Zabbix server is running	Yes	localhost:10051
Number of hosts (monitored/not monitored/templates)	31	5 / 0 / 26
Number of items (monitored/disabled/not supported)	317	287 / 0 / 30
Number of triggers (enabled/disabled)[problem/unknown/ok]	128	128 / 0 [6 / 0 / 122]
Number of users (online)	2	2
Required server performance, new values per second	3.93	-

Updated: 04:57:52

System status

Host group	Disaster	High	Average	Warning	Information	Not classified
Linux servers	1	0	0	0	0	0
Storage	0	0	0	0	0	0
Windows servers	0	4	0	0	0	0
Zabbix servers	0	0	0	1	0	0

Updated: 04:57:52

Host status

Host group	Without problems	With problems	Total
Linux servers	1	1	2
Storage	1	0	1
Windows servers	0	1	1
Zabbix servers	0	1	1

Updated: 04:57:52

Obrázek 5: Zabbix dashboard

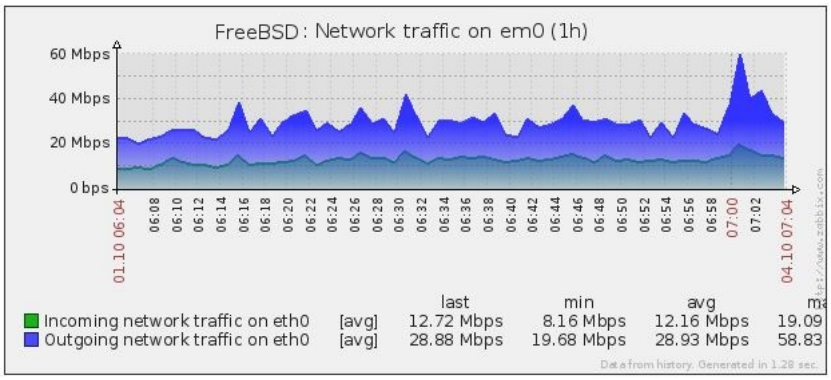
Grafy

Grafy v Zabbixu jsou buď již zakomponované v šabloně, případně si můžeme jakýkoliv vytvořit. K tomu je potřeba vědět, jaký typ grafu chceme, jelikož Zabbix podporuje koláčové a průběhové grafy s, nebo bez výplně. Následně stačí graf vytvořit v nastavení hosta a záložce Graph. Zde zvolíme hodnoty na ose X a Y, kterých může být i několik, případně zvolíme další doplňující vlastnosti grafu a dáme uložit.

Instalace

Základem bylo již předinstalované řešení tzv. "appliance", která je i dle popisu určená ke zhodnocení. Stažena byla verze ve formátu .OVF, který umožňuje jednodušší a bezchybné nasazení virtuálních strojů v rámci více virtualizačních platforem.(DMTF, 2013) Následně byl tento soubor importován na Hyper-V a před prvním spuštěním byly změněny parametry VM tak, aby měla k dispozici stejné výchozí podmínky, takže samotná instalace trvala pouze několik minut, jelikož ihned po spuštění je vše připraveno. Pro reálné nasazení je však doporučeno využít instalaci, která je popsána v dokumentaci[11] a je možné ji provést pro většinu linuxových

[11] "https://www.zabbix.com/documentation/2.0/manual/installation/install$_{f}rom_{p}ackages$



Obrázek 6: Zabbix graf - příchozí/odchozí komunikace na routeru MikroTik

systémů ze základních repozitářů, případně z dodatečných jako je např.: EPEL[12]. Ve druhém případě lze postupovat též dle dokumentace a využít zdrojových kódů a provést kompilaci přímo. U obou metod lze postupovat přesně dle dokumentace, což znamená prvně nainstalovat prereqizity, dle návodu a následně samotný Zabbix.Celá operace trvá přibližně 30 až 60 minut a zvládne ji i začínající správce.

Konfigurace - server

Po prvním spuštění Zabbixu po instalaci Vás vyzve k zadání základních údajů o uživateli, připojení na databázi a otestuje veškeré prereqizity. Veškeré kroky k dokončení instalace jsou opět uvedeny v dokumentaci dostatečně podrobně. Po dokončení konfigurace, případně po spuštění VM se na úvodní obrazovce přihlásíte a je vidět základní přehled - dashboard, který již obsahuje jednu položku a to samotný Zabbix server, na kterém si lze dobře prozkoumat hodně souvislostí a nastavení, které v Zabbixu jsou. Samotné nastavení nástroje pomocí GUI (*Graphical User Interface*) je minimální, jsou zde možnosti na zapnutí a konfiguraci proxy[13], změnit vzhled a ovlivnit pár možností ohledně zobrazení.

Konfigurace - host

Ve většině případů se budeme snažit nasadit agenta, který dokáže mnohem více, než samotné SNMP a je mnohem jednoduší na konfiguraci. Před přidáním hosta se prvně podíváme na internet na dostupné šablony (*template*) například na oficiálí wiki[14], pokud žádný vhodný nenalezneme, musíme si vytvořit vlastní. Použitý šablon

[12]"http://fedoraproject.org/wiki/EPEL

[13]"http://windows.microsoft.com/cs-cz/windows-vista/what-is-a-proxy-server

[14]"https://www.zabbix.com/wiki/templates/start

nám velmi usnadní konfigraci, jelikož se jedná v podstatě o dokment obsahující již vyplněné hodnoty OID, případně názvy procesů, vytvořené různé spoštěče a grafy. Pokud jsme tedy nenašli vhodnou šablonu, vytvoříme si vlastní v části konfigurace a template. Před samotným vytvořením si však musíme již zjistit infromace o zařízení a to minimálně:

- Název, který postihne vše co v šabloně bude.
- OID.
- Jak často se bude server dotazovat na hodnotu.
- Jak dlouho budeme uchovávat historii údajů.
- Jednotky, které získáváme a na co je budeme přepočítávat.
- Jaké budou základní hraniční hodnoty pro varování.

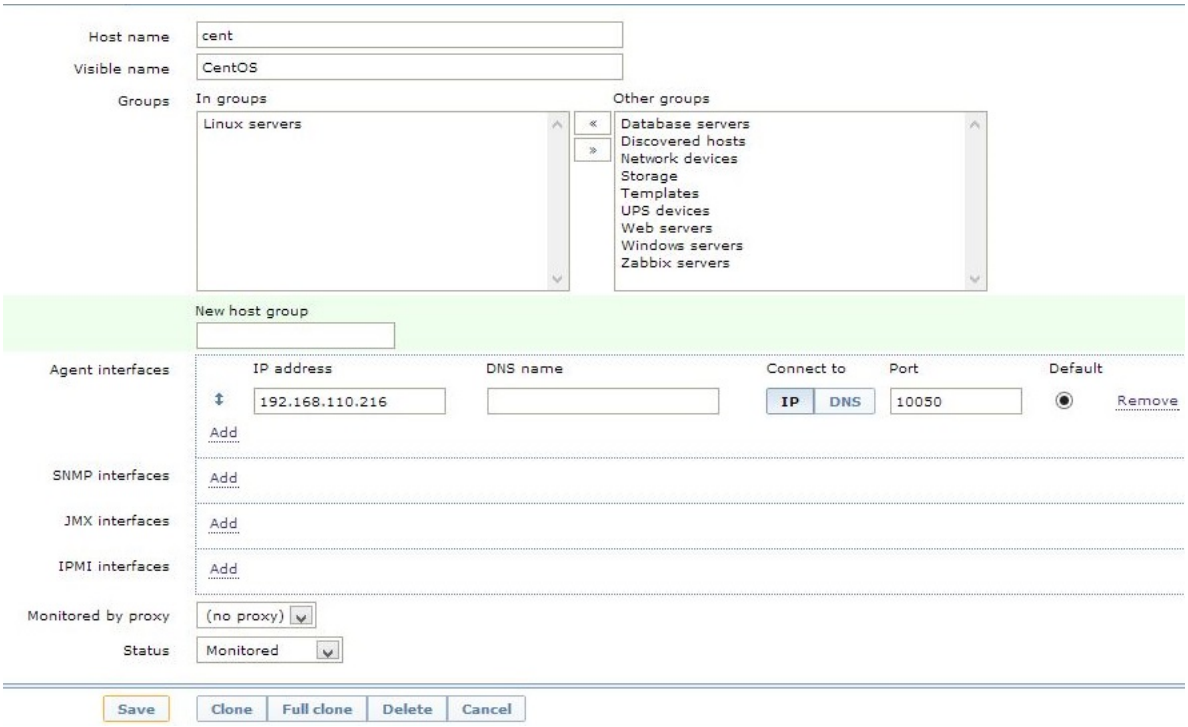
Při vytváření vyplníme tedy položku items, kde specifikujeme samotné OID a základní vlastnosti jako jsou jednotky, případně jejich přepočty na lepší hodnoty, interval v jakém se má položka zjišťovat a jak dlouho a v jakém formátu ji ukládat. Následně je důležité vytvořit si Trigger, který řeší změny hodnot a určí závažnost problému. Základní verze může vypadat například takto: (Template SNMP Synology:Disk1$_Temp.last(0)) > 50, covyuvItemDis1_Temp($specifikovaný v šabloně)porovnvjehoho$

Nastavení hosta krok po kroku. Opět je potřeba si předem specifikovat několik vlastností:

- Název, který se bude zobrazovat a interní hostname pro Zabbix.
- IP adresu, případně DNS jméno.
- Jak často se bude server dotazovat na hodnotu.
- Skupinu pod jakou patří.
- Šablony, které budeme chtít využít.
- Jaké budou základní hraniční hodnoty pro varování.

V menu konfigurace zvolíme HOST a přidat, otevře se obrazovka, kde provedeme základní nastavení. Zvolíme si HOSTNAME, na obrázku označené číslem 1, musí být unikátní a je důležité pro nastavení agenta, dále VISIBLE NAME označený 2, přiřazení do skupiny v obrázku 3 vlevo jsou zvolené, vpravo dostupné. Dále zvolíme druh monitorování, na obrázku 4, zvolen Agent, přičemž specifikujeme IP adresu či DNS jméno a port, který je v základu 10050, dále je možné ještě nastavit proxy server Zabbixu.

V následujícím kroku přiřadíme novému hostovi šablony, které se mají využívat, můžeme použít i více. V následujících záložkách lze ještě nastavit IPMI, makra a Host inventory, což je vlastně kompletní popis.



Obrázek 7: Přidání hosta a základní nastavení

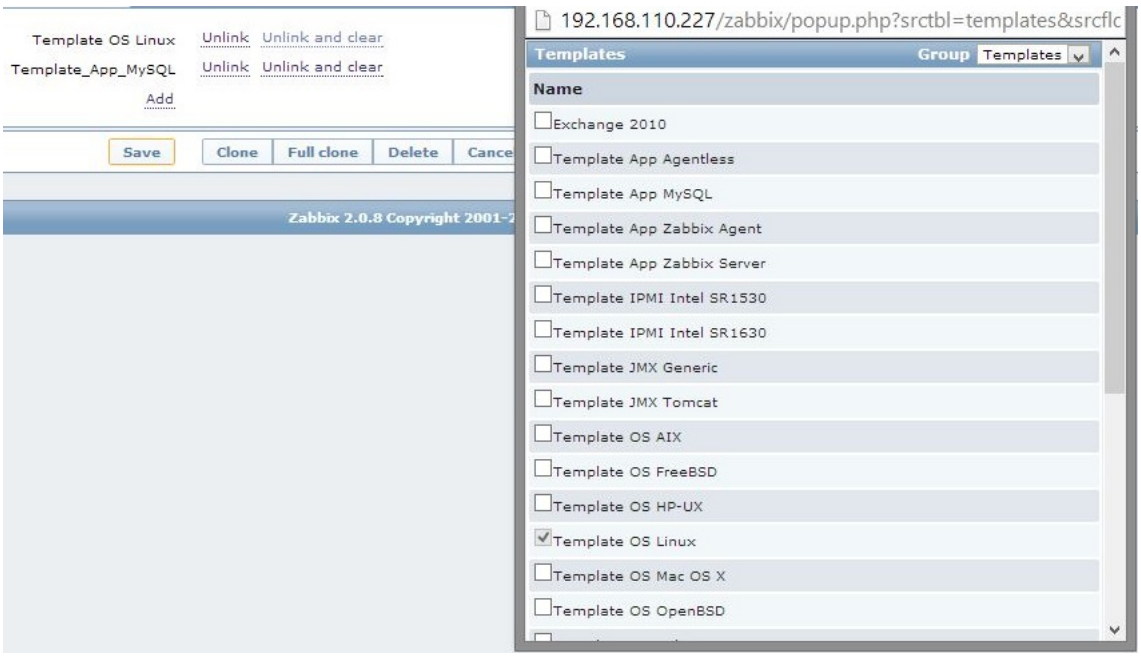
Nastavení agenta se provádí v jeho konfiguračním soubor, základní parametry mohou vypadat například takto:

Server=192.168.110.227 – IP adresa Zabbix serveru
Hostname=cent – hostname použité při přidávání na serveru!
StartAgents=5 – počet instancí Zabbix agentd
DebugLevel=3 – množství logovaných informací
LogFile=c:/Zabbix/zabbixagentd.log – cesta k logovacímu souboru (pro Win)
Timeout=3 – maximální čas zpracování úlohy
EnableRemoteCommands=1 – povolení vzdálených příkazů.

Provoz

Samotný provoz je dále bezúdržbový. Zásadní problém na který jsem však během testování narazilo byly pády databáze při testování s nižšími garantovanými prostředky, při použití 512MB RAM a 1GHz CPU hlásil Zabbix nedostupnost a chyby v databázi a pomohl vždy až restart celého serveru, proto je lepší i při menším počtu monitorovaných zařízení využít silnější server.

Výsledky testování



Obrázek 8: Přiřazení templates k novému hostovi

Dostupnost vždy měřil PING, který měl však problémy a při nastavení krátkého intervalu hlásil nedostupnost i v případě, že server byl dostupný, přičemž důvod byl dlouhý response time ve stovkách ms ze Zabbixu, avšak přímo ze základního systému, pod kterým Zabbix běžel byl do 2ms. Pomocí SNMP se několikrát stalo, že Zabbix vůbec události nezaznamenal, případě s větším zpožděním oproti agentovi. Krátkodobé výpadky nezaznamenal ani jeden způsob, příčinu bych viděl v tom, že dotaz vždy přišel ve chvíli, kdy služba běžela a díky tomu se na to nepřišlo. Pro sledování dostupnosti webových stránek byl využit monitoring webu zabudovaný v Zabbixu, který vypisuje rychlost stahování, odezvy a poslední status code.

Z hlediska náročnosti instalace a prvotní konfigurace a přidání klienta patří Zabbix k nejlehčím nástrojům a díky velké komunitě a aktivnímu vývoji se téměř vždy podaří problém vyřešit. Zabbix nepodporuje žádné externí moduly, avšak nabízí dostatek možností pro monitoring, případně lze dopsat vlastní skripty. Server běží pouze pod OS Linux, agenty má pro většinu velkých linuxových distribucí a univerzálního pro Windows. Z hlediska stability se jedná o podařený nástroj, který pokud dostane HW odpovídající požadavkům, tak poběží bez výpadků.

Díky velké rozšířenosti Zabbixu se dají z internetu stáhnout templates na všechna požadovaná zařízení a služby, které dokáží pomocí agenta systém monitorovat do hloubky a předávat tak detailní informace o chodu. V případě, že je potřeba sledovat pouze běh služby, lze jednoduše využít některého template a udělat si jeho kopii a změnit identifikátor služby, v případě využití SNMP stačí zjistit OID.

	Čas		Typ výpadku	Poznámka
1)	0:00:00		Výpadek služby na serveru	
	0:00:45		Agent	Agent reaguje téměř ihned
2)	0:07:00		Restart služby	
	0:07:59		Agent	Opět rychlá reakce
3)	0:14:00		Přeplnění disku	
	0:18:35		Agent	Pomalejší reakce, ale v limitu
4)	0:21:00		Kritické využití CPU	
	0:26:35		Agent	Celkem pomalá reakce - problém
5)	0:28:00		Kritická teplota	
	0:28:55		Agent	
6)	0:35:00		Kritické využití šířky pásma do internetu	
	0:38:50		Agent	
7)	0:42:00		Webserver - WEB hi response	
	0:47:05		Vestavěná utilita	
8)	0:49:00		Dlouhá fronta emailů	
	0:56:30		Agent	Zjistil výborně
9)	0:56:00		Výpadek směrovače	
	1:01:15		Ping	
10)	1:03:00		Výpadky spojení	
			Ping	Krátkodobé nepoznal
11)	1:10:00		Výpadek spoje na server	
	1:13:45		Ping	Reakce 5min unreachable
12)	1:17:00		Náhlé vypnutí serveru	
	1:21:30		Ping	Stejně jako výše
13)	1:24:00		Restart serveru	
	1:33:00		Agent	Po naběhnutí zpráva

Obrázek 9: Timeline výpadků

Zásadní nevýhodou Zabbixu je, že nepodporuje NetFlow, pro jeho využití musíme skriptem doimplemetovat, nebo využít jiný vhodnější nástroj.

Jelikož byl Zabbix nakonec nasazen i v produkčním prostředí, tak z testování se podařilo získat několik cenných infromací, za zhruba roční provoz bylo nasbíráno:

- Více jak pět miliónů záznamů v databázi.
- Velikost databáze 16 GB.
- Počet monitorovaných zařízení 71.
- Počet monitorovaných hodnot 3041.
- Počet spoštěčů různých událostí 870.
- Počet nových hodnot za sekundu 33.89.

7.2 Zenoss

Zenoss Core poskytuje webové rozhraní, které umožňuje systémovým administrátorům monitorovat dostupnost, konfiguraci, výkon a události na různých zařízeních. Veškerá data jsou ukládána do MySql databáze a primárně podporované systémy jsou RHEL a CenOS, avšak komunitou je podporováno i Ubuntu a Debian.

Dostupnost služeb

Možnost sledování běžících služeb zvládá Zenoss již po prvotním kotantu s hostem, avšak né vždy dokáže dále zjišťovat jejich stav, tato možnost se musí znovu nastavit. Dále automaticky přidá seznam všech otevřených portů, indentifikaci systému, veškerá síťová rozhraní, pevné disky a jejich využití a využití procesoru, případně ostatní informace, které vyčte pomocí SNMP protokolu.

Přehled

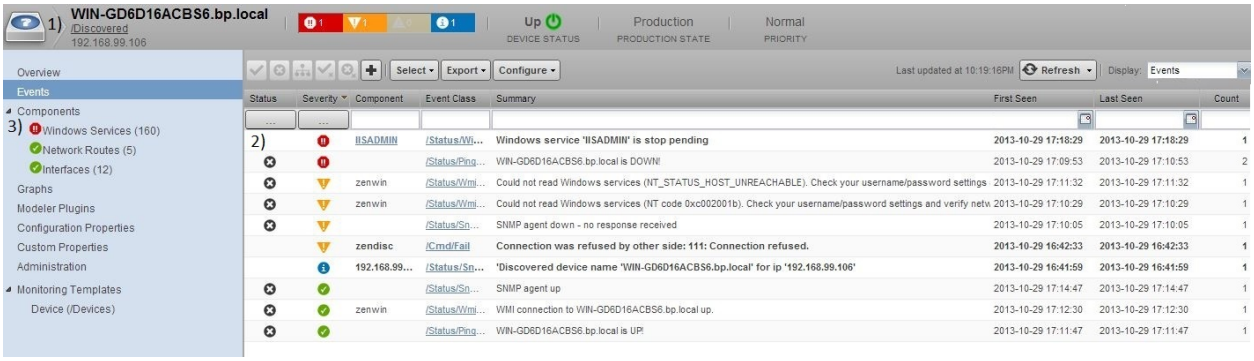
Základní přehled v Zenossu zobrazuje informace o zařízeních. Na obrázku vedle čísla jedna je vidět, zařízení, které bylo automaticky vyhledáno, včetně názvu, který nebyl dále upraven. Směrem doprava na řádku je vidět počet událostí, které se k tomuto zařízení vážou, jsou zde 2 kritická varování a jedno důležité. Zobrazované data jsou uživatelsky nastavitelné stejně jako doba aktualizace zobrazovaných dat..



Obrázek 10: Zenoss dashboard

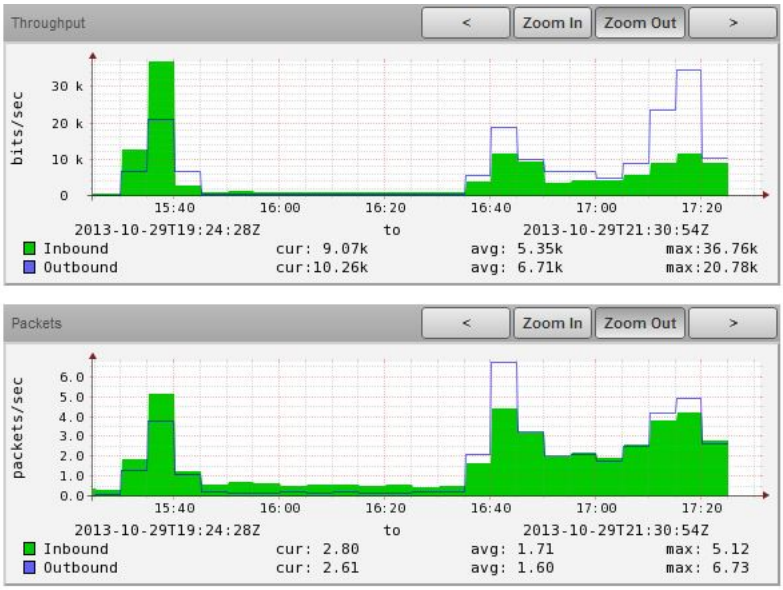
V Zenossu je pěkně udělán vždy přehled ke každému hostovi, který je vidět na následujícím obrázku. Na horní liště vedle jedničky je zobrazen název zařízení, jeho skupina a IP adresa. Směrem doprava je vidět stejný přehled jako v základním přehledu, stav zařízení, v tomto případě produkční, což znamená, že veškeré události jso logované a nakonec priorita zařízení. Pokud rozklikneme položku Events, tak pod číslem 2 je zde vidět kompletní historie událostí k zařízení a počet jejich výskytů. Vedle čísla tři jsou vidět komponenty zařízení, zde je možné zobrazit služby, routy a stav rozhraní.

Grafy



Obrázek 11: Zenoss přehled o hostovi

Grafy v Zenossu jsou přehledné a umožňují zobrazit volitelnou časovou osu do historie. Většina grafů je již obsažena v šablonách, takže je není potřeba vytvářet, avšak v případě, že je potřeba,kompletní postup je v dokumentaci[15].



Obrázek 12: Zenoss - příchozí/odchozí traffic v bps/pps

Instalace

Pro účely testování byl opět využit předpřipravený obraz, stažitelný přímo z ofociálních stránek produktu a je založený na distribuci CentOS. Pro produkční využití je opět doporučené provédst celou instalaci[16] přímo na serveru. Stažený stoj

[15]"http://community.zenoss.org/docs/DOC-12103

[16]"http://wiki.zenoss.org/Install*zenoss*

byl mírně upraven, aby poskytnutý výkon odpovídal námi stanoveným kritériím, jinak nebylo potřeba dělat žádné další úpravy a vše fungovalo dle očekávání. V případě instalace čistého Zenossu na jednu z podporovaných distribucí je možné využít připravený autodeploy, který po nastavení požadovaných balíčků stáhne a nainstaluje Zenoss dle Vašich přání, je pouze nutné mít připravený Apache a MySql databázi.

Konfigurace - server

Obdobně jako u Zabbixu je možné přez webové rozhraní provádět pouze základní operace. Ihned po spuštění se nástroj zeptá na automatické vyhledání zařízení v síti (*autodiscover*) a v případě, že využíváte Zenoss v lokální síti, velmi to usnadní práci.

Konfigurace - host

Opět jak u Zabbixu se snažme co nejvíce využít již hotové šablony, jelikož hledání zabere zlomek času, jak vytvoření nového template.

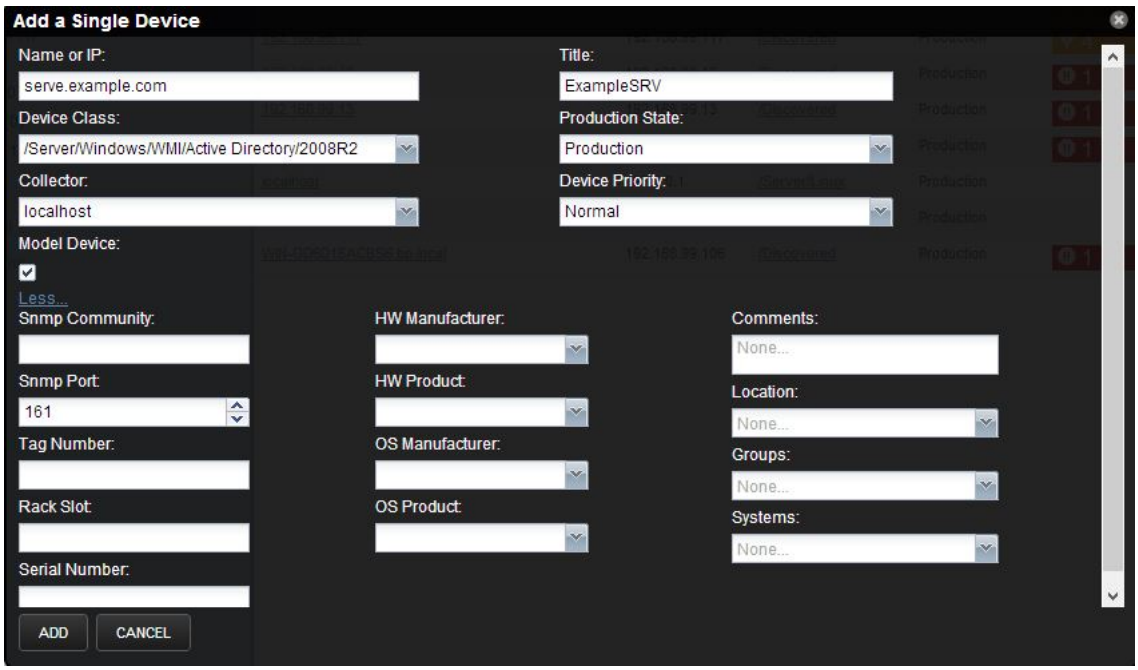
Samotné přidání je pak velmi jednoduché, pod záložko Infrastructure klikneme na tlačítko monitoru s pluskem a vybereme, zda chceme přidat jedno či více zařízení. Pro více zařízení se použije stejný průvodce jak při prvním spuštění, proto se jím nebudeme dále zabývat. Po zvolení přidání jednoho zařízení se objeví jednoduchý formulář, pro který musíme mít připravené:

- Název zaříení
- IP adresu, případně DNS jméno.
- Nastavení SNMP jako je community string a port.
- Ostatní infromace o serveru, například výrobce, umístění atd..
- Třídu zařízení, což je vlastně šablona.

Po kliknutí na tlačítko ADD se host automaticky přidá a pokud máme funkční šablonu a správně nataveno SNMP, tak je zařízení ihned monitorováno. Pro konfiguraci ostatních údajů musíte rozkliknout zařízení a pod záložkou Configuration Properties vyplnit požadované údaje, případně změnit již vyplněné, jako příklad lze uvédst SSH jméno a heslo.

Výše uvedený postup je základním scénářem. Pro využití protkolu WMI pod Windows je musíte prvně nakomfigurovat dle návodu[17] .

[17]"http://binarynature.blogspot.cz/2011/06/configure-windows-device-for-zenoss.html



Obrázek 13: Zenoss - přidání hosta

Provoz

Samotný provoz je téměř bezúdržbový, stačí pouze udržovat aktuální verze nainstalovaného software. Jediná věc, která mne na tomto nástroji zarazila byla dlouhá doba nabíhání, jelikož samotný Zenoss obsahoje několik až desítek démonů, takže v případě výpadku serveru s nástrojem příjdeme o celkem velké množství dat.

Výsledky testování

7.3 Nagios

Nagios je dnes jeden z nejpoužívanějších monitorovacích nástrojů. Stejně jako předchozí splňuje veškeré náležitosti vzešlé z požadavků analýzy. Jedná se primárně o linuxový nástroj, avšak bezproblému běží i na Unixu. Jako hlavní či doplňující nástroj používá po celém světě Nagios tisíce uživatelů. Z mé praktické zkušenosti s tímto nástrojem je velmi oblíben u poskytovatelů internetu.

Dostupnost služeb

V případě instalace agenta, který je dostupný pro OS Linux, Unix a Windows je možné sledovat i nesíťové atributy jako je využití disku, CPU, běžící služby, …Tuto vlastnost má společnou se Zabbixem, avšak primárně využívá protokul SNMP.

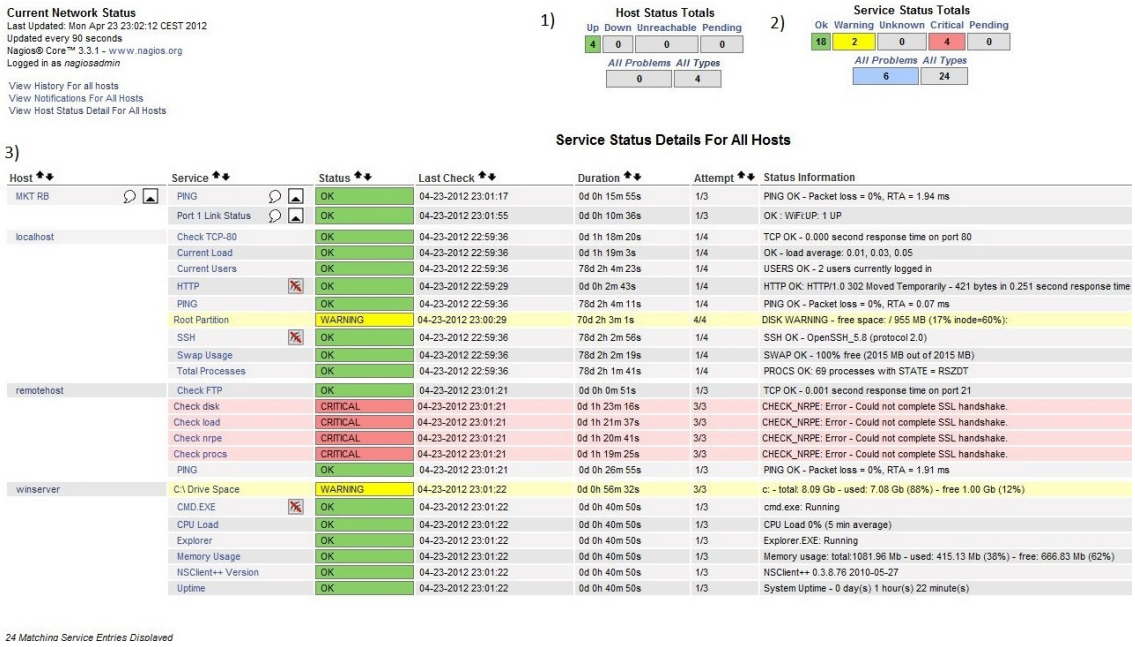
	Čas		Typ výpadku	Poznámka
1)	0:00:00		Výpadek služby na serveru	
	0:05:15			Pozdní reakce
2)	0:07:00		Restart služby	
				Nezaznamenal
3)	0:14:00		Přeplnění disku	
	0:18:35			
4)	0:21:00		Kritické využití CPU	
	0:23:35			
5)	0:28:00		Kritická teplota	
	0:23:15			
6)	0:35:00		Kritické využití šířky pásma do internetu	
	0:39:10			
7)	0:42:00		Webserver - WEB hi response	
				Nemonitorováno
8)	0:49:00		Dlouhá fronta emailů	
	0:56:30			Nemonitorováno
9)	0:56:00		Výpadek směrovače	
	1:00:05			
10)	1:03:00		Výpadky spojení	
				Nezaznamenal
11)	1:10:00		Výpadek spoje na server	
	1:13:45			
12)	1:17:00		Náhlé vypnutí serveru	
	1:19:30			
13)	1:24:00		Restart serveru	
	1:32:30			Po naběhnutí

Obrázek 14: Timeline výpadků

Přehled

Základní dashboard je velmi jednoduchý, avšak se z něj dozvíme vše podstatné. Zobrazuje 2 menší základní přehledy a to Host status na obrázku označen jedničkou, je zde sumarizace hostů a jejich stavů. Vedle tohoto přehledu najdeme Service Status označená dvojkou, zobrazuje podobný přehled pro všechny sledované služby. Pod těmito statusy je velký a detailní přehled označený trojkou. Tento přehled je rozdělen dle hostů a jejich služeb, zobrazuje tedy podstatné informace o stavu, včetně času a hodnoty poslední aktualizace hodnot a v případě chyby změní barvu podle důležitosti.

Dále je přehled o zařízení, který je velmi stručný. Obsahuje jeho status, časy poslední, plánované kontroly, poslední změnu stavu, ...Po pravé straně je však možnost nastavení vlastností monitorování u zařízení, jako je například vypnutí monitorování, nastavení času, kdy bude zařízení vypnuto, provést ihned kontrolu hodnot a další. Důležité je též menu, které je vlevo nahoře, z něhož se zanořujeme dále do detaiů o zařízení, ve smyslu historie alertů, trendy, report dostupnosti atd..



Obrázek 15: Nagios dashboard

Grafy

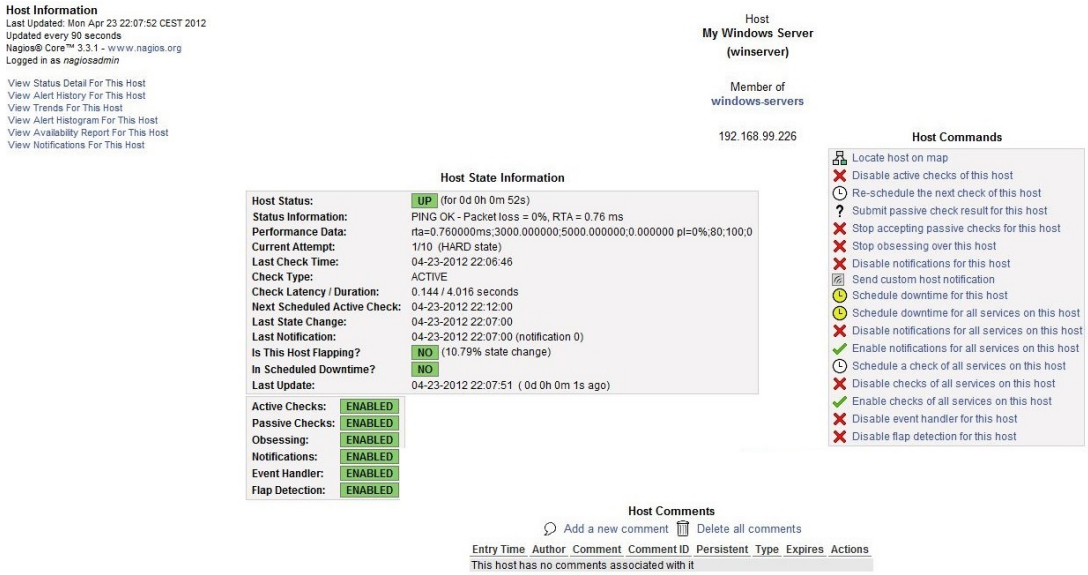
Základní instalace Nagiosu neobsahuje žádný nástroj pro vytváření grafů. Pro využívání této funkcionality je nutné doinstalovat přídavek Nagiosgraph, který je dostupný ze stránek Nagiosu. Návod krok po kroku napsal Tapas Mishra(2013), je nutné prvně projít relativně jednoduchou instlaci a následně pro každý prvek, ke kerému chceme vytvářet grafy musíme zaregistrovat službu. Nakonec stačí poze v konfiguračním souboru nadefinovat graf.

Instalace

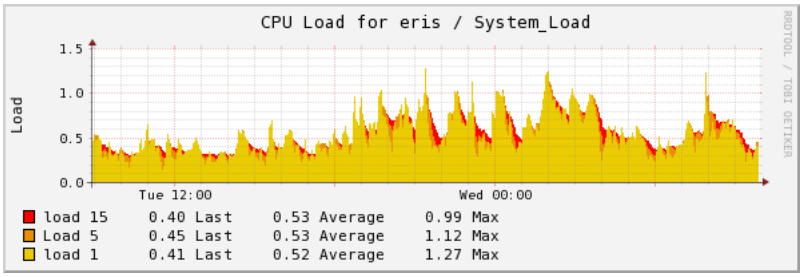
Samotná instalace můsí být provedena z konzole, jelikož Nagios nemá vlastní VM zdarma ani pro testovací účely, tato možnost je až při placené verzi. U Nagiosu to není velký problém, jelikož základní instlaci zvládne téměř každý a kompletní návod pro Ubuntu/RHEL/CenOS naleznete[18] na stránkách Nagiosu. Samozřejmě je potřeba nainstalovat i potřebné doplňky jako je Apache s PHP, databázi atd., avšak vše je v návodu pěkně vysvětleno, včetně instlace pluginů a základního spuštění samostatného jádra.

Konfigurace - server

[18]"http://assets.nagios.com/downloads/nagioscore/docs/Installing$_{N}agios_{C}ore_{F}rom_{S}ource.pdf$



Obrázek 16: Nagios přehled o hostovi



Obrázek 17: Nagios - CPU load Zdroj: https://www.nickebo.net/wp-content/uploads/2012/02/image.png

Pokud jsme pečlivě vše při instalaci nastavili, je server ihned připraven fungovat a není potřeba dále nic nastavovat, proto se můžeme zabývat pouze nastavneím hostů.

Konfigurace - host

Samostatná konfigurace je ze všech nástrojů nejnáročnější a doporučuje se využít předpřipravených šablon, kterých je plný internet. Konfigurace hostů, skupin, služeb a dokonce i grafů se provádí v konfiguračních souborech, webové rozhraní slouží pouze k interpretaci výsledků a nic v něm nenastavíte. Pro monitorování Windows stanic a vzdálených hostů je potřeba doinstalovat plugin NRPE[19]!

Konfigurační soubory objektů lze najít v následujících umístěních:

[19]"http://exchange.nagios.org/directory/Addons/Monitoring-Agents/NRPE–2D-Nagios-Remote-Plugin-Executor/details

cfg file=/usr/local/nagios/etc/hosts.cfg
cfg file=/usr/local/nagios/etc/services.cfg
cfg file=/usr/local/nagios/etc/commands.cfg

Subory obsahují definice objektů , které Nagios používá pro monitorování. Objektové konfigurační soubory obsahují definice hostitelských počítačů , skupin , kontaktů, skupin kontaktů , služeb, příkazy apod.

Nagios dále zpracuje všechny konfugurační soubory v následujících umístěních:

cfg dir=/usr/local/nagios/etc/commands
cfg dir=/usr/local/nagios/etc/services
cfg dir=/usr/local/nagios/etc/hosts

Krok po kroku: V konfuguračním souboru HOST nadefinujeme zařízení např.:

define host(
name host
use host
check command checkhostalive
max check attempts 15
notification interval 30
notification period 24x7
notification options d,r
register 0
)

Následně vytvoříme ve složce hosts soubor host.cfg s následujícími parametry:

define host(
host name MKT
alias MikroTik
address 192.168.99.1
check command checkhostalive
check interval 5
retry interval 1
max check attempts 5
check period 24x7
retain nonstatus information 0
notification interval 30
notification period 24x7
)

define hostgroup(
hostgroup name smerovace
alias Smerovace
members MikroTik
)

define service(
host name remotehost
service description CheckNRPE
check command checknrpe
max check attempts 5
check interval 5
retry interval 3
check period 24x7
notification interval 30
notification period 24x7

)

Existuje mnoho možností, které lze nastavit, detailní informace jsou v dokumentaci[20] . Tímto jsme provedli základní konfiguraci hosta. Následně na monitorovaném hostovi v našem případě Windwos musíme spustit NSClienta[21] a vše bude od této chvíle fungovat.

Provoz

Samostatný provoz Nagiosu je opět velmi pohodlný a nepotřebuje téměř žádnou údržbu. Výjimka je snad jen ta, že je relativně náročný na operační paměť, avšak při monitorování desítek hostů Vám základních 512MB bude dostačovat. Další nevýhoda je, pokud v konfuguračním skriptu uděláte chybu, tak služba nenaběhne a může se stát, že budete na několik minut bez dohledu, což se negativně projeví na grafech, proto doporučuji konfigurační soubory otestovat prvně na testovacím stroji.

Výsledky testování

[20]"http://nagios.sourceforge.net/docs/3$_0$/*objectdefinitions.htmlhost*

[21]"http://nsclient.org/nscp/

	Čas		Typ výpadku	Poznámka
1)	0:00:00		Výpadek služby na serveru	
	0:00:55		Agent	Rychlá reakce jako u Zabbixu
2)	0:07:00		Restart služby	
	0:09:59		Agent	
3)	0:14:00		Přeplnění disku	
	0:19:55		Agent	
4)	0:21:00		Kritické využití CPU	
	0:26:20		Agent	
5)	0:28:00		Kritická teplota	
	0:26:10		Agent	
6)	0:35:00		Kritické využití šířky pásma do internetu	
	0:39:50		Agent	
7)	0:42:00		Webserver - WEB hi response	
				Nemonitorováno
8)	0:49:00		Dlouhá fronta emailů	
	0:50:30		Agent	
9)	0:56:00		Výpadek směrovače	
	1:00:05		Ping	
10)	1:03:00		Výpadky spojení	
			Ping	Krátkodobé nezaznamenal
11)	1:10:00		Výpadek spoje na server	
	1:13:55		Ping	
12)	1:17:00		Náhlé vypnutí serveru	
	1:20:20		Ping	
13)	1:24:00		Restart serveru	
	1:35:00		Agent	

Obrázek 18: Timeline výpadků

7.4 Cacti

Cacti je kompletní grafické řešením navržené tak, aby využilo funkčnost RRDtool a možnost ukládání dat a následné vykreslování grafů na maximum. Poskytuje pokročilé možnosti ve využívání šablon a několik metod získávání dat.

Dostupnost služeb

Cacti samo o sobě nemá zabudovány mechanismy na ověřování dostupnosti, avšak z grafů je to vždy patrné, jestli je dostupnost 100% nebo 0%. Pro monitorování dostupnosti existuje oficiální template, který funguje na počítání dostupnosti z UPTIME, kdy vždy porovnává novou hodnostu s tou starou. Dostupnost služeb běžících nad operačním systémem umí Cacti monitorovat opět pouze pomocí SNMP.

Přehled

Cacti nemá vlasní zabudovaný přehled. Tyto vlastnosti se dají nainstalovat jako rozšíření, jedním z nejlépe hodnocených je hmib[22], případně je možnost grafy

[22]"http://docs.cacti.net/plugin:hmib

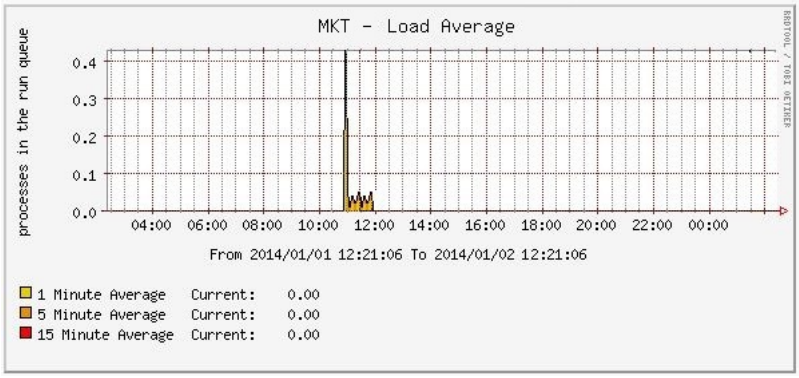
vytvořené v Cacti pomocí pluginu zakomponovat do Nagiosu, který vlastní grafy neumí.



Obrázek 19: Cacti dashboard s pluginem hmib. Zdroj: http://docs.cacti.net/$_{m}edia/plugin$: $hostresourcesmibdashboard.png$

Grafy

Jak již bylo zmíněno, Cacti se na grafy přímo zaměřuje, takže je to jeho silná stránka. Přímo v oficiální dokumentaci[23] . Jedná se o jedniný nástroj z celého testu, který umí přímo pracovat s NetFlow.



Obrázek 20: Cacti - CPU load

[23]"http://docs.cacti.net/templates

Instalace

Pro otestování byla opět využita předpřipravená appliance, která již obsahovala vše potřebné. Samotná instalace je relativně složitá oproti ostatním nástrojům, avšak i v tomto případě je dobře popsána v manuálu.

Konfigurace - server

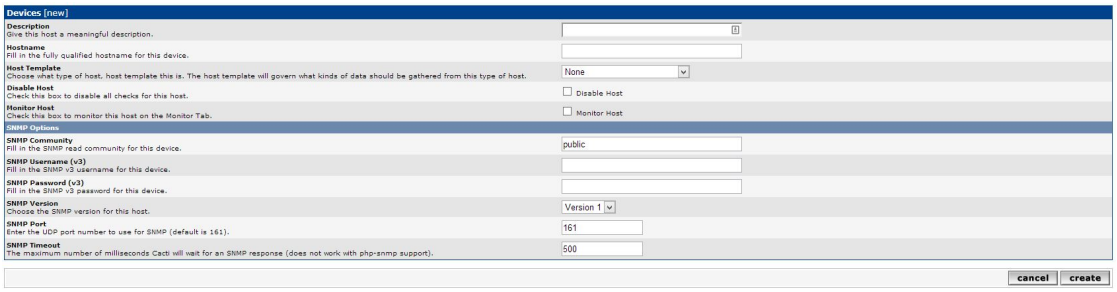
Cacti jako jeden z mála nástrojů dovoluje pomocí webového rozhraní nastavit hodně věcí. Z těch důležitých bych zmínil nastavení používané verze nainstalovaného SNMP, RRDTools, nastavení veškerých cest k souborům, maximální počet požadavků na SNMP za sekundu a mnohé další.

Konfigurace - host

Pod systémeme Windows byl využit SNMP server společnosti Informant Systemstems[24], jelikož na něj, narozdíl od vestavěného Windowsového SNMP serveru existuje několik povedených šablon, takže práce s nasazením výrazně ubyde. Stačí tedy software nainstalovat a pouze nastavit vlastnosti SNMP. Stejně tak na systémech Linux či směrovačích a přepínačích nastavíme pouze SNMP.

Krok pokroku: Jako první krok musíme najít šablonu pro zařízení, které chceme monitorovat, jelikož vtváření šablon je složitější a vyžaduje pokročilejší znaslosti. Jsou dvě možnosti, jak do Cacti nahrát šablonu, přímo ve webovém rozhraní kliknout na položku import templates a nyní stačí vybrat soubor se šablonou, případně ho vložit v textové podobě do boxu pro to určeného.

Následně vytvoříme monitorované zařízení v položce Devices klikneme na ADD a zobrazí se následující okno:

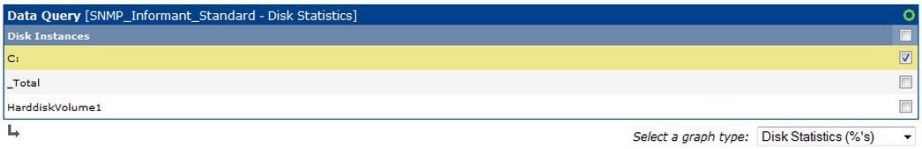


Obrázek 21: Cacti - vytvoření hosta

Do pole Description zadáme název zařízení a pozor, v poli Hostname musí být IP adresa, nebo DNS jméno, dále je postup jako u většiny nástrojů, přiřazení šablony, zatržení, že chceme zařízení monitorovat, vyplnění údajů pro SNMP.

[24]"http://www.snmp-informant.com/downloads.htmSNMP$_{I}nformant_{F}reeware_{P}roducts$

V následujícím kroku klikneme na Create a poté hned na Create graph, zde již máme na výběr všechny grafy, které šablona obsahovala.



Obrázek 22: Cacti - výběr grafu

V posledním kroku již pouze toto zařízení přiřadíme do stromu grafů v položce Graph tree.

Provoz

Samotné Cacti bylo provozováno pouze po dobu testů a v tomto případě nebylo vůbec potřeba cokoliv kontrolovat či měnit a svou úlohu odvedlo dokonale. Nároky na hardware byly minimální, téměř jsem ani nebyl schopen je zjistit.

Cacti je velmi komplexní a profesionální nástroj vytvořený na monitorování středních a rozlehlých sítí, který má obrovskou komunitu, díky které lze na internetu najít téměř veškerou funkcionalitu, co si jen administrátor může přát. Cacti je v hojné míře využíván jako doplňující nástroj Nagiosu, do kterého existuje několik integračních doplňků. Jednou z nevýhod je relativně vysoká náročnost na nasazení a zprovoznění všech požadovaných doplňků a šablon.

Výsledky testování

7.5 The Dude

Tento program je vyvinutý společností MikroTik, takže je hojně používaný v sítích poskytovatelů internetu, kteří využívají jejich produkty, avšak díky jeho kvalitě a přehlednosti jej využívá spousta administrátorů jako doplněk k ostatním monitorovacím nástrojům.

Dostupnost služeb

Jelikož je tento nástroj určený primárně na dohled nad dostupností z hlediska sítě, má však možnost stejně jako Cacti pomocí protokolu SNMP provádět dohled i nad službami operačního systému.

Přehled

The Dude poskytuje základní přehled ve formě mapy, kde je zobrazena barva dle stavu a případně i některé hodnoty, celá obrazovka je díky tomu velmi přehledná.

Zajímavější z hlediska dohledu je následně možnost si zařízení rozkliknout více do podrobna, nalezneme zde veškeré infromace i zařízení a v případě MikroTiku vše

	Čas		Typ výpadku	Poznámka
1)	0:00:00		Výpadek služby na serveru	
	9:15:00			Nezaznamenal v čase
2)	0:07:00		Restart služby	
				Nezaznamenal
3)	0:14:00		Přeplnění disku	
	0:19:35			
4)	0:21:00		Kritické využití CPU	
	0:23:20			
5)	0:28:00		Kritická teplota	
	0:31:10			
6)	0:35:00		Kritické využití šířky pásma do internetu	
	0:39:40			
7)	0:42:00		Webserver - WEB hi response	
				Neměřeno
8)	0:49:00		Dlouhá fronta emailů	
				Neměřeno
9)	0:56:00		Výpadek směrovače	
	0:59:40			
10)	1:03:00		Výpadky spojení	
	1:05:05			Jako jediný
11)	1:10:00		Výpadek spoje na server	
	1:13:45			
12)	1:17:00		Náhlé vypnutí serveru	
	1:23:20			
13)	1:24:00		Restart serveru	
	1:28:20			Rychlá reakce (ping)

Obrázek 23: Timeline výpadků

dokáže automaticky zdetekovat. Na obrázku je například vidět, jak vypadá přehled služeb a rozhraní poskytnutých přímo routerem.

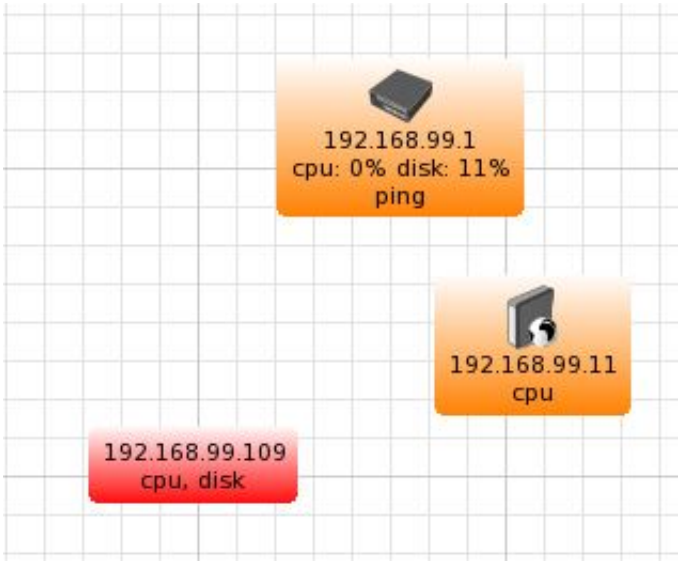
Grafy

Grafy se dají jednoduše vytvářet z jakýchkoliv získaných údajů. Stačí v záložce Charts vytvořit nový graf, pojmenovat ho a přidat objekt, což je vlastně grafovaná vlastnost, například CPU. Problém nastane, pokud máte hodně monitorovaných zařízení, jelikož zde je kompletní seznam a nelze ho nijak filtrovat.

Instalace

Jednou z mnoha výhod The Dude je jeho instalace, jelikož běží pod operačním systémem Windows, je pro něj připraven krátký instalační průvodce, kterým když projdete, ihned se program spustí a máte hotovo.

Konfigurace - server



Obrázek 24: The Dude dashboard

Samotný program má možnost v nastavení změnit různé intervaly dotazování, barvy grafů, nastavení SNMP, případně koordinaci s dalším serverem a dále je zde možné nastavit vzdálený přístup pomocí šifrovaného HTTPS.

Konfigurace - host

Stejně jako u Cacti byl pro dohled využit pouze protokol SNMP, který byl ve stejné konfiguraci.

Krok pokroku: Tento nástroj obsahuje velmi kvalitní automatické vyhledávání zařízení, které lze monitorovat. Na následujícím obrázku je vidět, že pro vyhledávání je potřeba zadat síť, která se má prohledat, agent, který se má použít, což je typicky výchozí agent programu a možnost zjištění názvu zařízení a režim detekce.

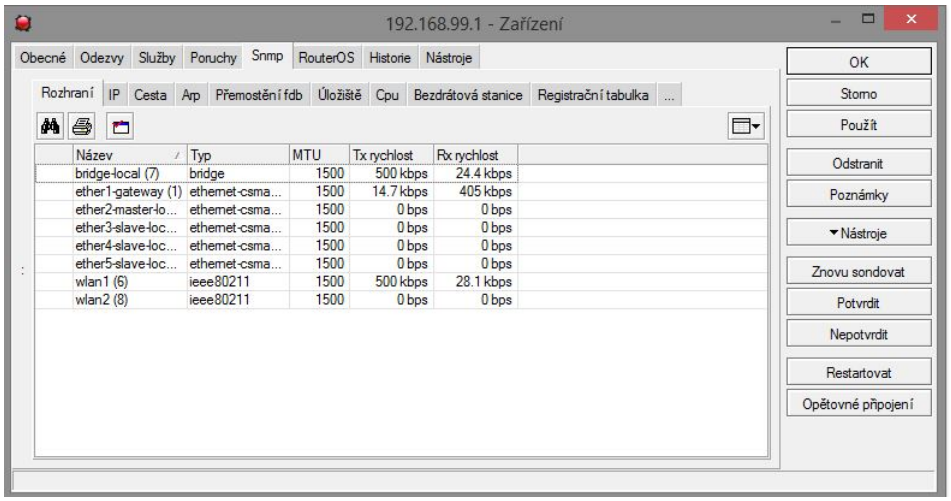
Druhá možnost je ruční přidání, což se provede v záložce Devices kliknutím na ikonku s červeným plus.

Pokud již máme zařízení nalezena a nejedná se o prvky s RouterOS, po rozkliknutí zařízení pouze přidáme potřebné OID, která chceme monitorovat.

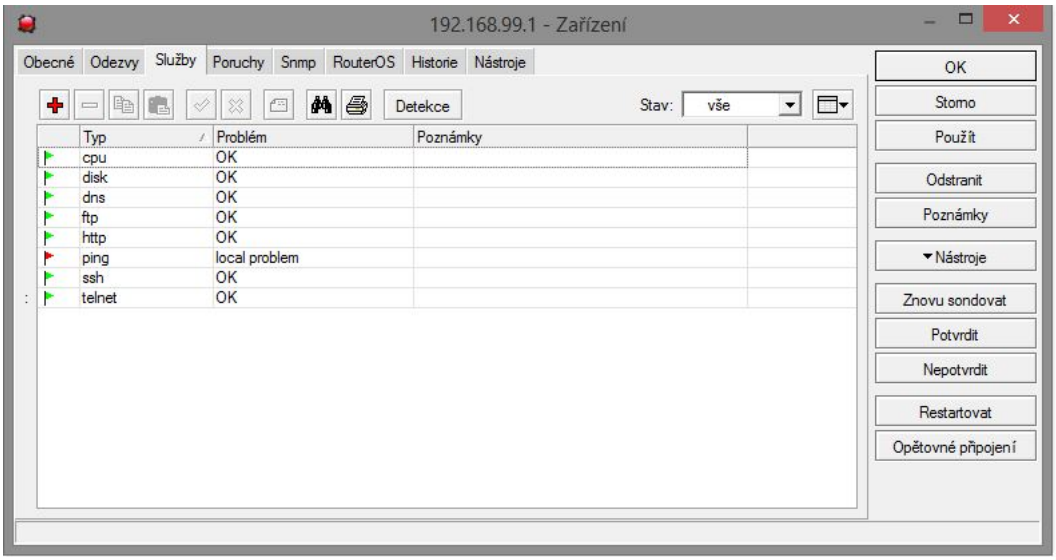
Provoz

Samotný provoz je velmi nenáročný a lze mít tento nástroj i na starších serverech. Stejně jako Cacti se The Dude většinou používá v kombinaci s ostatními nástroji. Hlavní výhodou je jeho přehlednost a v neposlední řadě i fakt, že je téměř celý v češtině a konfiguraci zvládne téměř každý.

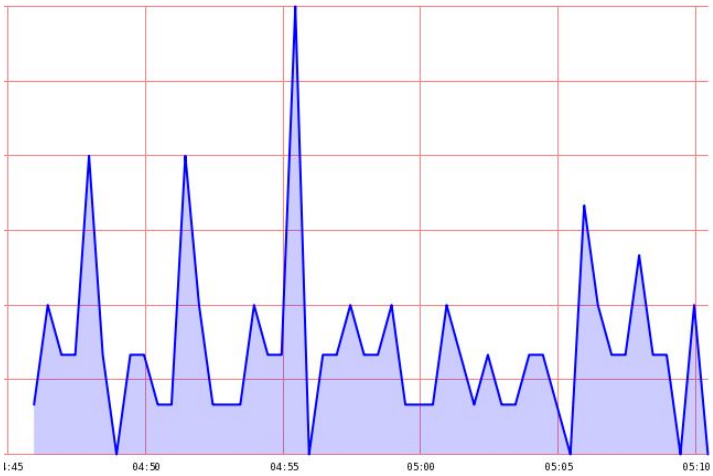
Výsledky testování



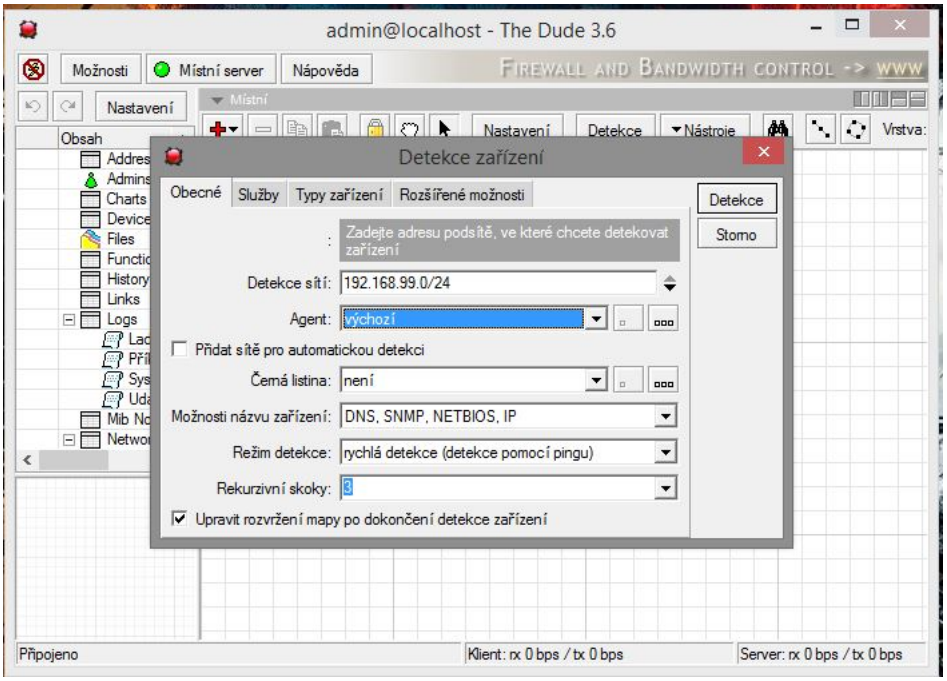
Obrázek 25: The Dude – zobrazení živého přehledu o rozhraních



Obrázek 26: The Dude – služby, které automaticky detekoval na routeru



Obrázek 27: Jednoduchý graf v The Dude



Obrázek 28: The Dude a autodetekce zařízení

	Čas		Typ výpadku	Poznámka
1)	0:00:00		Výpadek služby na serveru	
				Nezaznamenal v čase
2)	0:07:00		Restart služby	
				Nezaznamenal
3)	0:14:00		Přeplnění disku	
	0:17:55			
4)	0:21:00		Kritické využití CPU	
				Nezaznamenal v čase
5)	0:28:00		Kritická teplota	
	0:23:15			
6)	0:35:00		Kritické využití šířky pásma do internetu	
	0:35:50			Okamžitá reakce
7)	0:42:00		Webserver - WEB hi response	
				Neměřeno
8)	0:49:00		Dlouhá fronta emailů	
	0:56:30			Neměřeno
9)	0:56:00		Výpadek směrovače	
	1:01:15			
10)	1:03:00		Výpadky spojení	
	1:04:00			Okamžitá reakce
11)	1:10:00		Výpadek spoje na server	
	1:11:45			Okamžitá reakce
12)	1:17:00		Náhlé vypnutí serveru	
	1:18:30			Okamžitá reakce
13)	1:24:00		Restart serveru	
	1:28:00			Ryhlá reakce (ping)

Obrázek 29: Timeline výpadků

8 Ekonomické zhodnocení

U projektů spočívajících v zásahu do IT infrastruktury je stanovení nákladů velmi obtížné, nejtěžší je totiž stanovit náklady na lidské zdroje, jelikož tato položka se velmi těžko fixně stanovuje. Tuto položku jsem stanovil vlasním odhadem na základě zkušenosti s nasazením tohoto nástroje. Ekonomické zhodnocení bude provedeno pro nástroj Zabbix, který byl vybrán jako nejvhodnější. Z hlediska financí se platí hardware, což v našem případě znamená pouze server, software je open source a největší položku bude tedy tvořit samostatná lidská práce.

8.1 Náklady

Budeme uvažovat zakomponování do již stávající infrastruktury, přičemž server bude hostovaný externě jako virtuální stroj.

Jako jednotka měření nákladů na lidskou práci je použit tzv. manday, což je 8 pracovních hodin. Hodinová sazba je počítána z reálné mzdy, kterou společnost vyplácí zaměstnancům, kteří budou systém nasazovat, což je 300 Kč/hod.

Tabulka 2: Náklady na zavedení

Produkt	**Množství**	**Cena**	**Poznámka**
Virtuální server	1	cca 1 200 Kč/měs.	Cena za dostatečně výkony HW
Instalace + testování	14 mandays	33 600 Kč	Jednorázově
Reálné nasazení	6 mandays	14 400 Kč	Jednorázově
Provoz	5 H/měs.	1 500 Kč	Čas věnovaný správě serveru a průběžnému ladění

Celkové uvažované náklady za nasazení a půlroční provoz jsou tedy: 48 000 Kč jednorázově na kompletní nasazení monitorovacího systému. 16 200 Kč za fixní náklady na provoz a údržbu. Variabilní náklady na větší zásahy nejsou uvažovány, jelikož v tomto období by měly veškeré opravy a úpravy spadat pod fixní náklady.

8.2 Reálný přínos

I když se na první pohled zdá částka za nasazení a provoz monitorovacího systému vysoká, v konečném důsledku nám může ušetřit statisíce. Jak již bylo krátce zmíněno v úvodu, díky monitorování dosáhneme vyší dostupnosti našich služeb a tím nižších sankcí za různá SLA, která společnost poskytuje, případně předejde výpadku vlastních služeb.

V případě výpadku je správce ihned uvědoměn o výpadku a může ho řešit, čímž sníží nedostupnost služeb z až několika desítek minut na minimum v závislosti na druhu problému. V případě, že je celý wehosting redundantní, pomůže systém zavčas

Tabulka 3: Náklady na zavedení

Popis	**nedodržení SLA**	**paušál za SLA**
SLA na výpadky webhostigu malý e–shop	500 Kč/min	1 000Kč
SLA na výpadek webhostigu velký e–shop	1 500 Kč/min	8 000Kč
SLA na výpadek webhostigu - navštěvovaný web	500 Kč/hod	1 000Kč
SLA na reakci na výpadek u partnerů	nad 30min 500 – 10 000Kč	7 000Kč

zachytit výpadek primárního serveru, namísto situace, kdy přestane odpovídat i sekundární server.

Ve druhém případě – problém u klienta je společnost schopna zareagovat v případě volného technika téměř ihned, čímž se úplně vyhne základním poplatkům za nedodržení SLA.

Budeme–li uvažovat nejhorší možný scénář a to výpadek na 15 minut u hostingu a jeden nezjištěný výpadek u klienta, náklady by se pohobovaly v řádech statisíců korun. Nutno dodat, že jsou uvažovány jedny z nejtvrdších SLA, které má společnost a vždy záleží na samotné smlouvě mezi klientem a poskytovatelem, avšak i v případě menších pokut za výpadek se nám předcházením výpadků investice brzy vrátí.

Vyjádřeno přesně číselně pomocí ukazatele ROI: a

ROI = ((čistý zisk – počáteční investice) / počáteční investice) * 100 = %
čistý zisk je uvažována částka placená klienty za SLA a je počítána pro:

- 10x malý e–shop
- 2x velký e–shop
- 10x webové stránky s SLA
- 10 partnerů s SLA na řešení lokálního IT

ROI = ((10 000 + 16 000 + 10 000 + 70 000)*6) - 64 200 / 64 200) = 106 000 / 64 200 = 165%

Z této hodnoty plyne, že v případě zabránění všem výpadkům pomocí dohledového systému by byla návratnost investice za půl roku 165% , což vypadá jako ohromná výhoda, avšak do výpočtu nejsou zahrnuty náklady na provoz, obnovu a údržbu infrastruktury, které si ze zisku z SLA odečtou cca 75%, tudíž reálná hodnota ROI je zhruba 40%. Pro srovnání průměrná hodnota podobného ukazatele ROE se v Informační a komunikační činnosti pohybude okolo 15% [25] .

[25]"http://www.businessinfo.cz/app/content/files/zpravodajstvi-pro-export/Financni-analyza-2012.pdf

Závěrem tedy nutno zdůraznit, že náklady na zavedení a provoz monitorovacích serverů jsou minimální v porovnání s možnými následky, avšak nutno doplnit, že samotný monitoring bez redundance a ostatních podpůrných systémů není samospásný a je potřeba při stanovení SLA přemýšlet více do hloubky celého systému. Dalším přínosem je, že máme vždy reálné časové údaje o výpadku, včetně několika přecházejících údálostí a jsme schopni zabránit v budoucnu podobných chybám. V neposlední řadě je díky analýze využití prostředků možné lépe plánovat rozvoj infrastruktury v nejkritičtějších místech a tím lépe využít finanční zdroje na obnovu a údržbu.

9 Závěr

Cílem práce bylo porovnat dostupné monitorovací nástroje a otestovat je. V důsledku analýzy uživatelských požadavků však byly vybrány pouze open-source produkty. V práci je detailně rozebrán každý vybraný nástroj a otestovaný na připravené topologii, přičemž nakonec byl vybrán jako nejvhodnější Zabbix, kterému byla věnována největší pozornost i v této práci. Následně vytvořeno i ekonomické zhodnocení, ze kterého vzešlo, že nasazení dohledového systému ušetří společnosti velké množství finanční prostředků a rozhodně se vyplatí.

10 Literatura

Douglas R. Mauro a Kevin J. Schmidt. *Essential SNMP. 2nd ed.*.Beijing: O´Reilly, 2005, 442 s. ISBN 05-960-0840-6..

Stephen J. Bigelow *Mistrovství v počítačových sítích: správa, konfigurace, diagnostika a řešení problémů. Vyd. 1. Překlad Petr Matějů.*. Brno: Computer Press, 2004, 990 s. ISBN 80-251-0178-9..

James M. Kretchmar *Administrace a diagnostika sítí: pomocí OpenSource utilit a nástrojů. 1. vyd.*. Brno: Computer Press, 2004, 216 s. ISBN 80-251-0345-5..

Leskiw Aaron *What is SNMP How Do I Use It?* .[online].2012.[cit. 2013-01-16]. Dostupné z: <http://www.networkmanagementsoftware.com/snmp-tutorial> .

Bouška Petr *Začínáme s monitoringem sítě.* [online]. 1. 9. 2009 [cit. 2012-11-16]. Dostupné z: <http://www.samuraj-cz.com/clanek/zaciname-s-monitoringem-site> .

Bouška Petr *SNMP - Simple Network Management Protocol.* [online]. 20. 12. 2006 [cit. 2013-01-16]. Dostupné z: <http://www.samuraj-cz.com/clanek/snmp-simple-network-management-protocol> .

CISCO SYSTEMS, Inc. *MIB User Quick Reference.* [online]. [cit. 2013-01-16]. Dostupné z: <http://www.cisco.com/en/US/docs/ios/11_0/*mib*/*quick*/*reference*/*mtext.html* > .

ManageEngine, ZOHO Corp. *A beginner's guide to SNMP.* [online]. 27. 11. 2012 [cit. 2013-01-16]. Dostupné z: <http://www.snmplink.org/snmparticles/abeginnersguide/1> .

Microsoft *SNMP – zprávy protokolu.* [online]. [cit. 2013-01-16]. Dostupné z: <http://technet.microsoft.com/cs-cz/library/cc738074(v=ws.10).aspx> .

Microsoft *SNMP – zprávy protokolu.* [online]. [cit. 2013-01-16]. Dostupné z: <http://technet.microsoft.com/cs-cz/library/cc738074(v=ws.10).aspx> .

Habrman Zdeněk *Monitorování aktivních prvků počítačové sítě.*[online]. 2013 [cit. 2013-10-09]. Diplomová práce. Univerzita Tomáše Bati ve Zlíně, Fakulta aplikované informatiky. Vedoucí práce Jiří Korbel. Dostupné z: <http://theses.cz/id/pq519m/>..

Pleskot Vít *Dohledové systémy pro počítačové sítě.*[online]. 2012 [cit. 2013-10-09]. Bakalářská práce. Univerzita Pardubice. Fakulta elektrotechniky a informatiky. Vedoucí práce Josef Horálek. Dostupné z: <http://dspace.upce.cz/handle/10195/48433>..

KOCOURKOVÁ LUCIE *Udržitelnost a rozvoj monitorovacích systémů v komerční společnosti.*[online]. 2013 [cit. 2013-10-09]. Diplomová práce. Masarykova univerzita, Fakulta informatiky. Vedoucí práce Jaroslav Ráček. Dostupné z: <http://is.muni.cz/th/262486/fi$_{ma}$2/ > ..

SZABO GÁBOR *Monitoring lokální sítě prostřednictvím Nagiosu.*[online]. 2013 [cit. 2013-10-09]. Bakalářská práce. Masarykova univerzita, Fakulta informatiky. Vedoucí práce Eva Hladká. Dostupné z: <http://is.muni.cz/th/359846/fi$_b$/ > ..

DVOŘAN JAN *Návrh a implementace monitoringu síťových linek na platformě MikroTik.*[online]. 2013 [cit. 2013-10-09]. Bakalářská práce. Mendelova univerzita v Brně, Provozně ekonomická fakulta. Vedoucí práce Ing. Martin Pokorný, Ph.D.. Dostupné z: <https://is.mendelu.cz/auth/lide/clovek.pl?id=34399;zalozka=7;studium=43619>..

SZANISZLO TOMÁŠ *Správa heterogenních síťových prvků.*[online]. 2012 [cit. 2013-10-09]. Bakalářská práce. Masarykova univerzita, Fakulta informatiky. Vedoucí práce Jan Kasprzak. Dostupné z: <http://is.muni.cz/th/359894/fi$_b$/ > ..

Top 5 best network monitoring tools.[online]. 2012 [cit. 2013-10-09]. Dostupné z: <http://www.thegeekstuff.com/2009/09/top-5-best-network-monitoring-tools/>..

Top 10 best network monitoring tools.[online]. 2012 [cit. 2013-10-09]. Dostupné z: <http://www.siliconindia.com/news/enterpriseit/Top-10-Best-Network-Monitoring-Tools-nid-131075-cid-7.html/>..

WIKIPEDIA *Comparsion of network monitoring systems.*[online]. 2012 [cit. 2013-10-09]. Dostupné z: <http://en.wikipedia.org/wiki/Comparison$_o f_n etwork_m onitoring_s ystems$ > ..

BURDA PAVEL *Průzkum a ověření možností SNMPv3 na Cisco IOS.* [online]. [cit. 2013-01-16]. Dostupné z: <http://wh.cs.vsb.cz/sps/images/3/31/Burda-SNMPv3.pdf> .

CISCO SYSTEMS, INC. *Introduction to Cisco IOS NetFlow - A Technical Overview.* [online]. [cit. 2013-01-16].
Dostupné z: <http://www.cisco.com/en/US/prod/collateral/iosswrel/ps6537/ps6555/ps6601/prod$_w hite_p aper$0900$aecd$80406232.$html$ > .

PŘISPĚVATELÉ WIKIPEDIE *Netflow.*[online].Wikipedie: Otevřená encyklopedie, c2012, Datum poslední revize 1. 06. 2012, 07:09 UTC [cit. 2012-06-15]. Dostupné z: <http://cs.wikipedia.org/w/index.php?title=Netflowoldid=8605944>.

PETERKA JIŘÍ *IP - Internet Protocol.*[online].IP - Internet Protocol. archiv článků a přednášek Jiřího Peterky. [cit. 2013-07-09]. Dostupné z: <http://www.earchiv.cz/anovinky/ai1843.php3>.

Microsoft Corporation *Internet Control Message Protocol (ICMP) Basics.*[online].support.microsoft.com, Datum poslední revize 23. 02. 2007, [cit. 2013-07-09]. Dostupné z: <http://support.microsoft.com/kb/170292/en-us>.

InetDaemon Enterprises *CPU - Central Processing Unit.*[online]. 2013 [cit. 2013-10-09].
Dostupné z: <http://www.inetdaemon.com/tutorials/computers/hardware/cpu/>..

InetDaemon Enterprises *Random Access Memory (RAM).*[online]. 2013 [cit. 2013-10-09].
Dostupné z: <http://www.inetdaemon.com/tutorials/computers/hardware/memory/>..

InetDaemon Enterprises *Hard Disk.*[online]. 2013 [cit. 2013-10-09].
Dostupné z: <http://www.inetdaemon.com/tutorials/computers/hardware/drives/hard-disk.shtml>..

SolarWinds *Avalibility.*[online]. 2013 [cit. 2013-10-09].
Dostupné z: <http://www.solarwinds.com/it-management-glossary/what-is-network-availability.aspx>..

IETF. RFC4253. 1990 Dostupné z: <http://www.ietf.org/rfc/rfc1157.txt>.

IETF. RFC791. 1981 Dostupné z: <http://www.ietf.org/rfc/rfc791.txt>.

IETF. RFC791. 1981 Dostupné z: <http://www.ietf.org/rfc/rfc791.txt>.

IETF. RFC792. 1981 Dostupné z: <http://www.ietf.org/rfc/rfc792.txt>.

IETF. RFC793. 1981 Dostupné z: <http://www.ietf.org/rfc/rfc793.txt>.

IETF. RFC768. 1980 Dostupné z: <http://www.ietf.org/rfc/rfc768.txt>.

IETF. RFC768. 1990 Dostupné z: <http://www.ietf.org/rfc/rfc1157.txt>.

IETF. RFC4253. 2006 Dostupné z: <http://www.ietf.org/rfc/rfc4253.txt>.

Bouška Petr *OSI model.* [online]. 2007 [cit. 2012-11-16]. Dostupné z: <http://www.samuraj-cz.com/clanek/osi-model/> .

Microsoft *About WMI.* [online]. 2013 [cit. 2013-12-16].
Dostupné z: <http://msdn.microsoft.com/en-us/library/aa384642(v=vs.85).aspx>
.

Microsoft *WMI Arcitecture.* [online]. 2013 [cit. 2013-12-16].
Dostupné z: <http://msdn.microsoft.com/en-us/library/aa394553(v=vs.85).aspx>
.

Boháč Leoš *IP protokol.* [online]. 2013 [cit. 2013-12-12].
Dostupné z: <http://data.cedupoint.cz/oppa$_e$ − $learning/2_KME/054.pdf$ > .

David Klementa *Load Average*. [online]. 2013 [cit. 2013-12-16]. Dostupné z: <http://www.abclinuxu.cz/poradna/linux/show/260689> .

Přispěvatelé Wikipedie *IOPS*.[online].Wikipedie: Otevřená encyklopedie, c2012, Datum poslední revize 14. 11. 2013, 16:06 UTC [cit. 2013-12-20]. Dostupné z: <http://en.wikipedia.org/wiki/IOPS>.

Microsoft *Hyper-V Overview*. [online]. 2013 [cit. 2013-12-10]. Dostupné z: <http://technet.microsoft.com/cs-cz/library/hh831531.aspx> .

Zabbix documentation. [online]. 2013 [cit. 2013-10-09].
Dostupné z: <https://www.zabbix.com/documentation/> .

DMTF *Open Virtualization Format (OVF)*. [online]. 2013 [cit. 2013-08-07]. Dostupné z: <https://www.zabbix.com/documentation/> .

Tapas Mishra *Steps to configure graph with Nagios Core*. [online]. 2013 [cit. 2013-08-07]. Dostupné z: <http://www.linuxfunda.com/2013/04/02/steps-to-configure-nagiosgraph-with-nagios-core/> .

Sflow. [online]. 2013 [cit. 2013-08-07]. Dostupné z: <http://www.sflow.org/about/index.php> .