Monitorování počítačových sítí a aplikací pomocí programu Nagios.

Monitoring of computer networks and applications using Nagios.

Bc. Miroslav Mihok

Diplomová práce
2010

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky

Univerzita Tomáše Bati ve Zlíně
Fakulta aplikované informatiky
akademický rok: 2009/2010

ZADÁNÍ DIPLOMOVÉ PRÁCE
(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Bc. Miroslav MIHOK**
Studijní program:	**N 3902 Inženýrská informatika**
Studijní obor:	**Informační technologie**
Téma práce:	**Monitorování počítačových sítí a aplikací pomocí programu Nagios**

Zásady pro vypracování:

1. **Zpracujte literární rešerše na dané téma.**
2. **Analyzujte požadavky na systém umožňující sběr informací o topologii sítě.**
3. **Analyzujte využívané služby a protokoly.**
4. **Nakonfigurujte a zrealizujte monitorování počítačové sítě na Linuxové platformě ve firmě NESS pomocí programu Nagios.**
5. **V počítačové síti zrealizujte monitorování dostupnosti serverů, aplikačních procesů na serverech a zatížení jednotlivých částí sítě.**
6. **Nakonfigurujte oznámení administrátorovi pomocí emailů a SMS.**
7. **Proveďte srovnávací analýzu s jiným existujícím řešením.**
8. **Zhodnodte dosažené výsledky.**

Rozsah práce:

Rozsah příloh:

Forma zpracování diplomové práce: **tištěná/elektronická**

Seznam odborné literatury:

1. **DAVID, Josephsen. Building A Monitoring Infrastructure With NAGIOS. 1st edition. United States, Boston: Pearson Education, Inc., 2007. 230 s. ISBN 0-132-23693-1.**
2. **BARTH, Wolfgang. Nagios, System and Network Monitoring. 1st edition. Germany, Munich: Open Source Press GmbH, 2006. 462 s. ISBN 3-937514-09-0.**
3. **SCHUBERT, Max, et al. Nagios 3 Enterprise Network Monitoring Including Plug-Ins and Hardware Devices. 1st edition. United States, Burlington: Syngress Publishing, Inc., 2008. 384 s. ISBN 13: 978-1-59749-2.**
4. **DOSTÁLEK, Libor, KABELOVÁ, Alena. Velký průvodce protokoly TCP/IP a systémem DNS. 2. vydání. Česká republika, Praha: Computer Press, 2000. 426 s. ISBN 80-7226-323-4.**
5. **Kolektiv autorů. Linux-Dokumentační projekt. 4. vydání. Česká republika, Praha: Computer Press, 2008. 1336 s. ISBN 978-80-251-1525-1.**
6. **KRETCHMAR, James. Open Source Network Administration. 1st edition. United States, Upper Saddle River: Prentice Hall Professional, 2003. 238 s. ISBN 13: 9780130462107.**
7. **BAUER, Kirk, CAMPI, Nate. Automating UNIX and Linux System Administration. 2st edition. United States, New York: APress, Inc. 2003. 549 s. ISBN 1-59059-212-3.**

Vedoucí diplomové práce: **Ing. Miroslav Matýsek, Ph.D.**
Ústav počítačových a komunikačních systémů

Datum zadání diplomové práce: **19. února 2010**

Termín odevzdání diplomové práce: **8. června 2010**

Ve Zlíně dne 19. února 2010

L.S.

prof. Ing. Vladimír Vašek, CSc.
děkan

prof. Ing. Vladimír Vašek, CSc.
ředitel ústavu

ABSTRAKT

Cieľom tejto diplomovej práce je zoznámiť čitateľa s voľne dostupným a bezplatným dohľadovým systémom na monitorovanie počítačových sietí.

V teoretickej časti je popísaný dôvod prečo vlastne treba monitorovať počítačovú sieť, sú rozobrané jednotlivé metódy a nástroje monitorovania a bližšie sa zoznámenie s konkrétnym monitorovacím nástrojom Nagios. Ďalej bude popísaný postup jeho konfigurácie a v praktickej časti inštalácia a aplikácia jeho funkcionalít na súčasne najpoužívanejšie operačné systémy. Systém bude schopný zasielať oznámenia emailom, SMS správami a generovať štatistické grafy z nameraných hodnôt. Na záver bude vykonané porovnanie s iným existujúcim riešením.

Kľúčové slová: Nagios, monitorovanie, počítačová sieť, SNPM protokol, monitorovací systém, SAP, RRDtool, Cacti.

ABSTRACT

Focus of this thesis is to introduce the reader to freely reachable and free of charge system for monitoring of computer networks.

In theoretical part there is mentioned reason why i tis necessary to monitor computer network. There are mentioned different methods and tools of monitoring and closer introduction with specific monitoring system Nagios. There is also procedure of its configuration mentioned and at practical part of thesis there is installation and application of its functions in accordance with current most used operating systems. System will be able to send notifications by email, SMS or generate statistical diagrams with measured values. At the end there will be comparison with already existing solution.

Keywords: Nagios, monitoring, computer network, SNMP protocol, monitoring system, SAP, RRDtool, Cacti.

Ďakujem svojmu zamestnávateľovi NESS Slovensko a.s. za poskytnutie technického vybavenia a mojim kolegom za odborné vedomosti. Ďalej by som chcel poďakovať svojej rodine za výborné podmienky, ktoré mi boli pre moje štúdium vytvorené a svojmu vedúcemu diplomovej práce pánovi Ing. Miroslavu Matýskovi, Ph.D. za odborné vedenie a cenné rady, ktoré mi boli nápomocné pri písaní diplomovej práce.

Prohlašuji, že

- beru na vědomí, že odevzdáním diplomové/bakalářské práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb. o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších právních předpisů, bez ohledu na výsledek obhajoby;
- beru na vědomí, že diplomová/bakalářská práce bude uložena v elektronické podobě v univerzitním informačním systému dostupná k prezenčnímu nahlédnutí, že jeden výtisk diplomové/bakalářské práce bude uložen v příruční knihovně Fakulty aplikované informatiky Univerzity Tomáše Bati ve Zlíně a jeden výtisk bude uložen u vedoucího práce;
- byl/a jsem seznámen/a s tím, že na moji diplomovou/bakalářskou práci se plně vztahuje zákon č. 121/2000 Sb. o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon) ve znění pozdějších právních předpisů, zejm. § 35 odst. 3;
- beru na vědomí, že podle § 60 odst. 1 autorského zákona má UTB ve Zlíně právo na uzavření licenční smlouvy o užití školního díla v rozsahu § 12 odst. 4 autorského zákona;
- beru na vědomí, že podle § 60 odst. 2 a 3 autorského zákona mohu užít své dílo – diplomovou/bakalářskou práci nebo poskytnout licenci k jejímu využití jen s předchozím písemným souhlasem Univerzity Tomáše Bati ve Zlíně, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly Univerzitou Tomáše Bati ve Zlíně na vytvoření díla vynaloženy (až do jejich skutečné výše);
- beru na vědomí, že pokud bylo k vypracování diplomové/bakalářské práce využito softwaru poskytnutého Univerzitou Tomáše Bati ve Zlíně nebo jinými subjekty pouze ke studijním a výzkumným účelům (tedy pouze k nekomerčnímu využití), nelze výsledky diplomové/bakalářské práce využít ke komerčním účelům;
- beru na vědomí, že pokud je výstupem diplomové/bakalářské práce jakýkoliv softwarový produkt, považují se za součást práce rovněž i zdrojové kódy, popř. soubory, ze kterých se projekt skládá. Neodevzdání této součásti může být důvodem k neobhájení práce.

Prohlašuji,

- že jsem na diplomové práci pracoval samostatně a použitou literatúru jsem citoval. V případě publikace výsledků budu uveden jako spoluautor.
- že odevzdaná verze diplomové práce a verze elektronická nahraná do IS/STAG jsou totožné.

Ve Zlíně …….

podpis diplomanta

OBSAH

ÚVOD

V súčasnej elektronickej dobe sa už asi žiadna firma, alebo organizácia nezaobíde bez informačných technológií. Väčšina ľudí vníma počítač iba ako prostriedok na posielanie emailov, surfovanie po internete a použitie kancelárskeho softvér. Samozrejme, že tí technickejšie zdatnejší vieme aj o inom využití počítačov, napr. na programovanie aplikácií, ďalej môžu slúžiť ako výrobné nástroje, alebo ich využiť na zber dát.

Samotné počítače, či už ich hardwarové alebo softwarové vybavenie býva čoraz zložitejšie a pri ich implementovaní dochádza mnohokrát k nepredvídateľným problémom, či je to už technická chyba (vyhorenie dôležitej súčasti počítača, nenaštartovanie servera po výpadku elektrickej siete), alebo chyba zlyhaním ľudského faktoru (administrátor nastaví zle oprávnenia). Takéto problémy majú často na svedomí aj úplnú nedostupnosť systému, čo má za následok aj finančné straty. Pri nedostupnosti sa užívatelia začnú sťažovať a často krát sa k správcovi servera hlásenie o probléme dostane po dlhom čase od vyniknutia poruchy.

Typický príklad pracovného dňa administrátora siete:

Je desať hodín v pondelok ráno. Riaditeľ pobočky zúri, pretože čaká na dôležitý email, ktorý mu nebol ešte doručený. Administrátor rýchlou kontrolou zisťuje, že správy nie sú uviaznuté vo fronte, neexistuje žiadna zmienka v logu a mail od odosielateľa dorazil. Tak kde je problém? Centrálny poštový server spoločnosti nereaguje na odozvy programu Ping. To je zrejme koreň problému. Ale IT oddelenia v kancelárii spoločnosti trvá na tom, že nie je na vine, ale tvrdí, že sieť v sídle beží hladko, takže problém musí niesť sieť na pobočke. Vyhľadávanie chyby pokračuje a nakoniec sa zistí, že VPN (Virtual Private Network) linka do ústredia bola nefunkčná pretože na záložnej linke neboli nastavené smerovacie pravidlá. Konečný výsledok je mnoho strávených minút s hľadaním chýb, podráždený riaditeľ (rokovania, pre ktoré bol e-mail nevyhnutne potrebný už dávno skončili) a spotený administrátor.

Pokiaľ je ale nasadený nejaký monitorovací dohľadový systém, vtedy väčšinou správcovia systému vedia o vzniknutom probléme takmer okamžite. Ihneď ho majú presne lokalizovaný a môžu ho rýchlo odstrániť. Vo veľkej časti takto odpadá dohadovanie s užívateľom, ktorý nemusí byť odborníkom v informačných technológiách. Je samozrejmé, že nasadením monitorovacieho systému sa prudko znižuje doba nedostupnosti a takisto aj ňou spôsobená škoda. Pomocou týchto systémov, je možné predísť viacerým

problémom, napr. zastaveniu servera z dôvodu preplnenia diskového poľa je možné vysledovať ešte skôr, než vôbec nastane.

Pri vhodnom nasadení monitorovania, je možné mať pod dohľadom aj nekritické zariadenia ako sú napr. sieťové tlačiarne, ktoré vedia poslať informáciu o zostávajúcom množstve náplne v toneri a či nie je v stave, kedy potrebuje opravu apod.

Ak firma disponuje veľkým počtom systémov a zariadení, tak je priam potrebné nasadiť nejaký dohľadový systém. V prípade, že chce ešte aj ušetriť finančné prostriedky, tak ideálnou voľbou je popozerať sa po voľne šíriteľnom softvér. Jednou z mnoho možností je NAGIOS.

I. TEORETICKÁ ČASŤ

1 SIEŤOVÝ MANAŽMENT

1.1 Prečo monitorovať počítačovú sieť

Účelom monitorovania počítačovej siete je sledovať a kvalifikovať, čo sa v sieti deje a samozrejme veľmi často, rýchlo a jednoducho posúdiť jej kvalitatívne parametre, akým sú napríklad priepustnosť siete, oneskorenie a poskytnutie výkonnostných charakteristík pre prevádzkovateľov, alebo náročnejšieho užívateľa. Pomocou monitoringu zisťujeme, či je linka v prevádzke, alebo je v danej chvíle plne vyťažená. Optimálny a spoľahlivý výkon servera sa nekončí iba jeho nainštalovaním a nakonfigurovaním, ale aj po týchto činnostiach treba aj naďalej sledovať a optimalizovať jeho výkonnosť. Práve z týchto dôvodov prichádzajú na rad monitorovacie a vizualizačné systémy, ktoré tieto zariadenia pravidelne monitorujú a vďaka ktorým môže správca rýchlo zareagovať na prípadné problémy.

Analýzou týchto zozbieraných dát sa naskytuje aj ďalšia možnosť využitia monitorovania:

- Ladenie výkonu: zisťovanie a zníženie prekážok, vyvážené využívanie zdrojov atď.
- Riešenie problémov: identifikácia, diagnostika a opravy chýb.
- Plánovanie: odhad rozsahu požadovaných prostriedkov.
- Vývoj a dizajn nových technológií: porozumenie súčasným novým sieťovým trendom a novým vyvíjajúcim sa technológiám.
- Pochopenie zložitosti riadenia.
- Poskytnutie údajov pre modelovanie a ďalšiu simuláciu.
- Potvrdenie správnosti fungovania siete a jej parametrov, ktoré boli pred jej realizáciou predpovedané.

1.2 Motivácia na používanie sieťových meraní.

Pre bližší pohľad môžeme rozdeliť predchádzajúci zoznam do troch častí.

1. ISP - Internet Service Providers.
2. Používatelia
3. Predajcovia

V tabuľke (Tab. 1.) sú uvedené aspekty, prečo tieto zúčastnené strany majú záujem na sledovanie siete.

Tab. 1. Motivácia na používanie sieťových meraní.

	Cieľ	Meranie
ISP	plánovanie kapacít	využitie šírky pásma
	účtovanie na základe využívania kapacít	Paketov za sekundu
		Straty paketov
		Dosiahnuteľnosť
		Diagnostika smerovania
Používatelia	Monitorovanie výkonu	Dostupnosť pásma
	Plánovanie aktualizácií	Doba odozvy
	Dojednanie služieb na zákazku	Strata paketov
	Optimalizácia doručovania obsahu	Dosiahnuteľnosť
	Používanie pravidiel	Kvalita služieb
		Hostiteľský výkon
Predajcovia	Zlepšenie výkonu	Analýza log súborov
	Optimalizovanie konfigurácie zariadení	Analýza trace súborov
	Vykonávanie realtime ladení	

ISP majú veľký záujem o prenos maximálneho množstva dát s minimálnymi nákladmi a s maximálnym ziskom, používatelia majú zase úplne odlišné požiadavky. Oni radšej preferujú malé platby za poskytovanie spojenia a z čo najviac možnou šírkou prenosového pásma.

2 MONITOROVACIE METÓDY A NÁSTROJE

2.1 Meracie metódy

Meracie metódy môžeme rozdeliť do dvoch skupín:

1. Pasívne
2. Aktívne

2.1.1 Pasívne monitorovanie

Účelom pasívneho monitorovania je len počúvanie sieťového prenosu na sieti, pričom do siete nie sú posielané žiadne testovacie pakety. Vyhodnocujú sa iba časové a objemové charakteristiky užívateľskej prevádzky. Z toho plynie, že pasívne monitorovanie nespôsobuje dodatočné zaťaženie počítačovej siete a zariadení. Sledujú sa iba charakteristiky, ktoré sa nedajú zistiť aktívnym monitorovaním.

Existujú tri spôsoby ako zhromažďovať dáta zo siete:

1. Kopírovaním dát v uzle siete.
2. Pasívnym počúvaním.
3. Hardwarový merací prístroj.

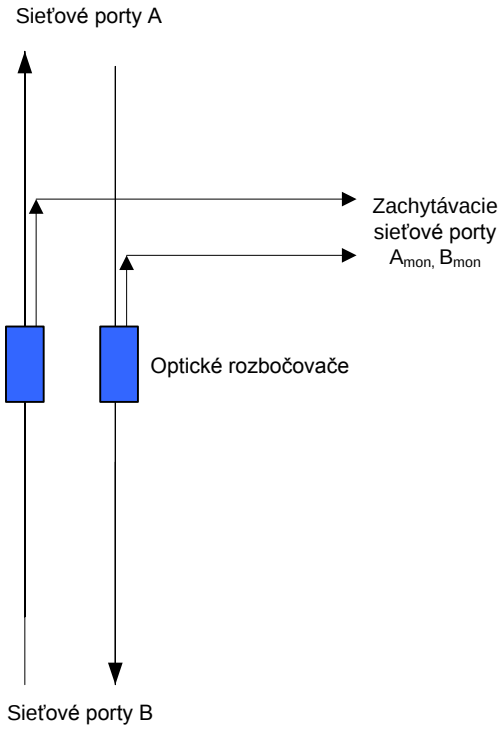
2.1.1.1 Kopírovanie dát v uzle siete

Niektoré sieťové zariadenia, napr. prepínače (switche), ktoré pracujú na druhej sieťovej vrstve Layer2 OSI (Open Systems Interconnection) modelu, môžu byť nakonfigurované takým spôsobom, aby preposielali, alebo zrkadlili všetky pakety, ktoré pretekajú jeho portom na druhý port, kde budú zhromažďované. Pri tomto riešení môže nastať výkonnostný problém (preťaženie, presnosť), kedy sa prepínač snaží preposlať viacej rýchlostných liniek na jeden port.

2.1.1.2 Pasívne počúvanie

Pri pasívnom počúvaní dáta na medenom, alebo optickom spojení, možno zachytávať pomocou rozbočovača (splitter). Na obrázku (*Obr. 1*) je znázornené, ako v prípade optickej linky je pomocou rozbočovača presmerovaná časť svetelného signálu na inú optickú linku. Rozbočovač je pasívnym prvkom a preto merania neovplyvňujú bežnú

prevádzku. Výhodou je, že aj v prípade výpadku napájania, rozbočovače si plnia svoju funkciu, pretože nevyžadujú žiadne napájanie.



Obr. 1. Zachytávanie sieťovej prevádzky na optickom vedení pomocou rozbočovača.

2.1.1.3 Hardvérový merací prístroj

Použitie hardvérového meracieho prístroja je asi najhoršia možnosť. Konektor je pripojený do meracieho zariadenia, ktoré doslova kopíruje všetky prichádzajúce dáta. Nevýhoda je, že pokiaľ bude výpadok elektrickej energie, tak bude ohrozená aj sieťová prevádzka.

V podstate je zriadenie pasívneho monitorovacieho systému relatívne ľahké. Stačí iba pripojenie monitorovacieho zariadenia nejakým spôsobom k sieti a potom sa začnú zhromažďovať dáta a neskôr analyzovať. Netreba však ale zabúdať aj na ochranu osobných údajov a mať pred zberom dát zodpovedanú otázku, ako ďalej so získanými údajmi, ako ich spracovať a ako k nim pristupovať. Preto by monitorovanie a monitorovacie zariadenia mali byť zabezpečené tak, aby sa k zachyteným dátam nedostala nepovolaná osoba.

Obrovské množstvo zachytených dát môže vyvolať aj veľké problémy. Veľká nevýhoda je, že ak sa budú monitorovať linky s veľkou prenosovou rýchlosťou so snahou

zachytávať všetky pakety, tak aj 1 TB úložného priestoru medzi hodnotou a jednotkou bude môcť uložiť iba niekoľko hodín takýchto zachytených dát.

2.1.2 Aktívne monitorovanie

Aktívne monitorovanie sa spolieha na schopnosť aplikácie testovacích paketov do siete, alebo poslania paketov na serveri a aplikácie, čím vlastne meria a sleduje ich služby. Takto môžeme merať priepustnosť počítačovej siete, jej priepustnosť, stratovosť. Bohužiaľ, takéto testovanie má za následok aj možné zvýšenie zaťaženia siete a niekedy aj jej preťaženie.

Pri aktívnom monitorovaní sa treba zamyslieť, či tento typ monitorovania naozaj prináša realistické výsledky. Ak sa napríklad testuje dostupnosť servera paketom ICMP (Internet Control Message Protocol), neovplyvňuje a neuprednostňuje ISP prenos tohto paketu pred bežnými paketmi na sieti? V tom prípade by hodnoty boli skreslené a neodpovedali by skutočným hodnotám.

Ďalším možným problémom môže byť prípad, keď sa opakovanie dopytu na testovanie nastaví na dlhší čas a medzi dvoma meraniami môže prísť k výpadku siete. V tomto prípade sa administrátor ani nedozvie, že prišlo k anomálii na sieti.

2.2 MONTOROVACIE NÁSTROJE

Na monitorovanie počítačových sietí je v súčasnosti na trhu veľmi veľa dostupných nástrojov. Rozdiel je hlavne vo vyhotovení riešenia a v konečnom dôsledku v cene. Existujú rôzne komerčné a profesionálne riešenia, ale tie sú pre väčšinu firiem veľkou počiatočnou investíciou a namiesto toho sa radšej poobzerajú po riešení z oblasti voľne dostupných produktov, ale na úkor presnosti a obmedzenejších možností využitia.

2.2.1 Základné softvérové diagnostické nástroje

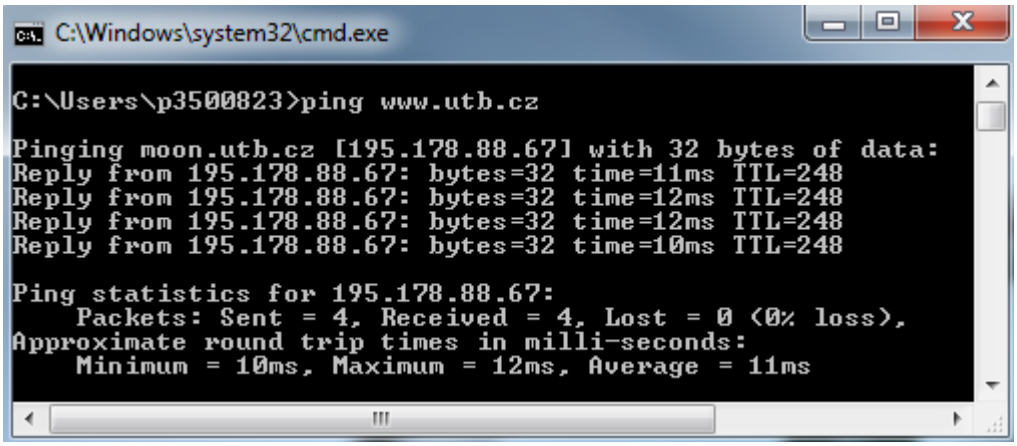
Softvérové nástroje sú vyznačované ako nástroje s veľkou možnosťou ich konfigurovania, ktoré poväčšine bývajú pevne viazané k typu operačného systému. Mnohé z nich sú špeciálne určené na účely napr. diagnostiku bezpečnosti, využitia systémových prostriedkov atď. Pravdepodobne v každom operačnom systéme existujú základné diagnostické nástroje, pomocou ktorých vie sieťový administrátor rýchlo, jednoducho a presne posúdiť kvalitatívne parametre počítačovej siete.

2.2.1.1 Ping

Pomocou tohto základného príkazu protokolu TCP/IP (Transmission Control Protocol/Internet Protocol) je možné jednoducho detekovať a riešiť problémy s dosiahnuteľnosťou a konektivitou. Pomocou neho môžeme otestovať dostupnosť počítača, servera, alebo iného sieťového zariadenie nielen v danej lokálnej sieti, ale ak je k dispozícii internet, tak kdekoľvek na svete.

Príkaz *ping* na dokazovaný počítač posiela správu *ICMP echo request* , pomocou ktorej sa pýta, či je dosiahnuteľný. Ak áno, tak dotazovaný cieľový hostiteľ preposiela odpoveď správou *ICMP echo reply*. Na obrázku (*Obr. 2*) je možné vidieť, že veľkosť posielaného paketu je 32 bajtov, minimálna odozva bola 10ms a max. 12ms a hodnota TTL (Time To Live) je 248, čo znamená, že paket môže prejsť ešte 248 smerovačmi a a až potom bude zahodený. Vždy, keď prejde smerovačom, tak sa hodnota TTL zníži o 1, čím je zabezpečené, že paket nebude zbytočne blúdiť po sieti a nebude vytvorená slučka.

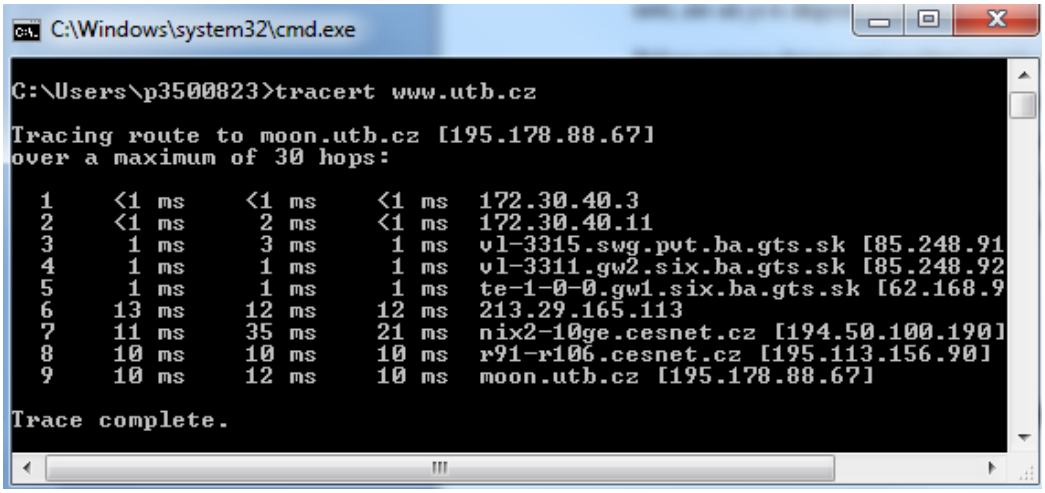
Protokol ICMP je služobný protokol, ktorý je súčasťou IP protokolu. Slúži k signalizácii mimoriadnych udalostí v sieťach, ktoré sú založené na IP protokole. Protokolom ICMP je možné signalizovať najrôznejšie situácie, skutočnosť je však taká, že konkrétne implementácie TCP/IP podporujú vždy iba určitú časť týchto signalizácií a navyše z bezpečnostných dôvodov môžu byť na smerovačoch ICMP signalizácie zahadzované [4].



Obr. 2. Výstup z vykonania príkazu „ping www.utb.cz“.

2.2.1.2 Tracert

Program tracert je jednou z ďalších utilít protokolu TCP/IP, ktorá umožňuje sledovanie cesty, ktorou sa paket dostane k cieľovému hostiteľovi. Ako testovací paket sa používa vo Windows ICMP paket a v Unix UDP (User Datagram Protocol) paket. Zdroj postupne vyšle tri pakety, ktoré pokiaľ prejdú hostiteľom tak prepošle ICMP time exceeded, TTL sa zníži o jedna a pakety sa prepošlú ďalej. V prípade, že sa jedná už o cieľového hostiteľa, tak sa prepošle ICMP echo reply. Cestou sa rozumie zoznam rozhraní routerov v ceste medzi zdrojom a cieľom.



Obr. 3. Výstup z vykonania príkazu „tracert www.utb.cz“.

Akonáhle sa vo výpise v mieste časového údaja objaví hviezdička, tak to znamená, že daný uzol nekomunikuje.

2.2.2 Hardvérové nástroje

Hardvérové nástroje majú výhodu, že pracujú samostatne, čiže nie sú závislé na ďalších prostriedkoch, pričom poskytujú rovnaké možnosti ako softvérové nástroje. Uskutočňujú operácie na nižších úrovniach komunikácie v siete, ako je sledovanie časových a objemových charakteristík paketu a jeho klasifikácie podľa protokolu, alebo iných údajov a následne spracovanie na vyšších úrovniach, ako sú výpočty dlhodobých štatistík, alebo rozpoznávanie možných bezpečnostných útokov podľa obsahu vybraných filtrovaných paketov.

Potrebné hardvérové monitorovacie nástroje je dnes možné získať od niekoľko výrobcov, ale ich nevýhodou je vysoká nadobúdacia cena.

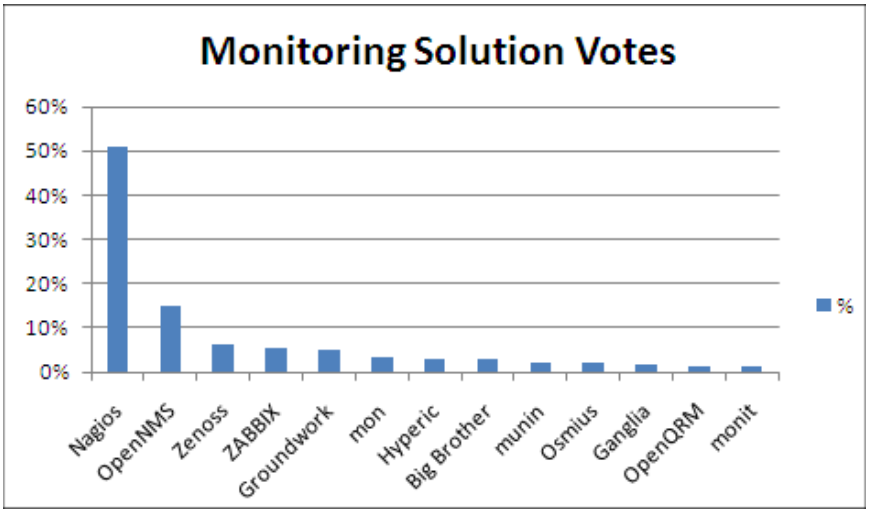
3 DOHĽADOVÝ SYSTÉM - NAGIOS

3.1 OPIS

Vývoj programu začal v roku 1999. Pôvodný názov projektu bol Netsaint[1], ale v roku 2002 bol ukončený a ďalej pokračoval pod novým názvom – *NAGIOS*. Autorom je pán Ethan Galstad, ktorý je v súčasnosti aj prezidentom spoločnosti Nagios Enterprises.

Názov Nagios je rekurzívnym akronymom slovného spojenia N.A.G.I.O.S. – „Nagios Ain't Gonna Insist On Sainthood“, čo by sa do slovenčiny dalo voľne preložiť ako: „Nagios, nie je pripravený stať sa svätým“. Názov a logo Nagios sú ochrannou známkou Nagios Enterprises.[2] Program je vydávaný pod licenciou GNU GPL (GNU's Not Unix General Public Licence), čiže je voľne k dispozícii na osobné a aj komerčné použitie a je ho možné kopírovať, distribuovať, modifikovať.

Je to veľmi populárny monitorovací systém, o ktorom svedčí aj anketa, ktorá bola zverejnená na diskusnom fóre pre fanúšikov distribúcie Linux.[3] viď. (*Obr. 4*).



Obr. 4. Anketa o najlepší monitorovací nástroj. [18]

[1] http://netsaint.sourceforge.net/

[2] http://www.nagios.com/legal/trademarks/

[3] http://www.linuxquestions.org/questions/2009-linuxquestions-org-members-choice-awards-91/network-monitoring-application-of-the-year-780663/

Tento systém nedrží iba dohľad nad sieťovými zariadeniami (servermi, switchmi, tlačiarňami), ale kontroluje aj služby, ktoré poskytujú (mailová pošta, web server, databáza). Je naprogramovaný tak, aby informoval o prípadných výpadkoch skôr, než to urobí samotný zákazník, užívateľ, alebo v horšom prípade nadriadený.

V dobe písania tejto diplomovej práce je k dispozícii posledná stabilná verzia s označením verzie 3.2.1, ktorá je určená pre systému na báze UNIX/Linux.

3.2 Požiadavky na monitorovací systém

3.2.1 Požiadavky na funkcionalitu

3.2.1.1 Monitorovanie služieb a stavov operačných systémov

- Možnosť v pravidelných intervaloch testovať dostupnosť sieťových služieb.
- Merať hodnoty dôležitých parametrov sieťových služieb.
- Možnosť spustenia monitorovania sieťovej služby na žiadosť.

3.2.1.2 Oznamovanie problémov

- V prípade vzniknutého problému bude administrátor upozornený pomocou mailovej správy.
- V prípade vážnejšieho problému bude administrátor upozornený pomocou SMS správy (Short Message Service).
- Informovanie o probléme bude doplnená dostatočným množstvom relevantných informácií.
- Možnosť definovania kontaktných skupín.

3.2.1.3 Administrácia a správa

- Administrácia a vizualizácia pomocou webového rozhrania.
- Možnosť definovania užívateľov na správu webového rozhrania s rôznymi oprávneniami.

3.2.2 Hardvérové požiadavky

V dokumentácii nie je nikde napísaná konkrétna doporučená zostava na monitorovací server s použitím Nagios. Podľa vyjadrenia jedného spokojného užívateľa na diskusnom portáli[4] Nagios, tak počítačový desktop s dvojjadrovým procesorom a s 5GB operačnej pamäte by mal stíhať spracovať minimálne 2000 služieb za minútu.

3.2.3 Softvérové požiadavky

- Web server napr. Apache a vyššie.
- PHP 4.3 a vyššie.
- MySQL 4.1 a vyššie.
- PEAR Module: HTML_Template_IT 1.1 a vyššie.
- PHP Extension: gettext.
- PHP Extension: mysql.
- PHP Extension: ftp.
- Povolený Javascript v internetovom prehliadači.

3.3 Architektúra

3.3.1 Démon

Démon Nagiosu je hlavnou súčasťou jadra. Pri jeho naštartovaní sa z príde k tomu, že sa načítavajú z konfiguračných súborov nastavenia a začína sa monitorovanie zariadení a služieb. Komunikácia démona s okolím je realizovaná pomocou súborov, do ktorých ukladá svoje výstupy a aj z nich načítava vstupné údaje.

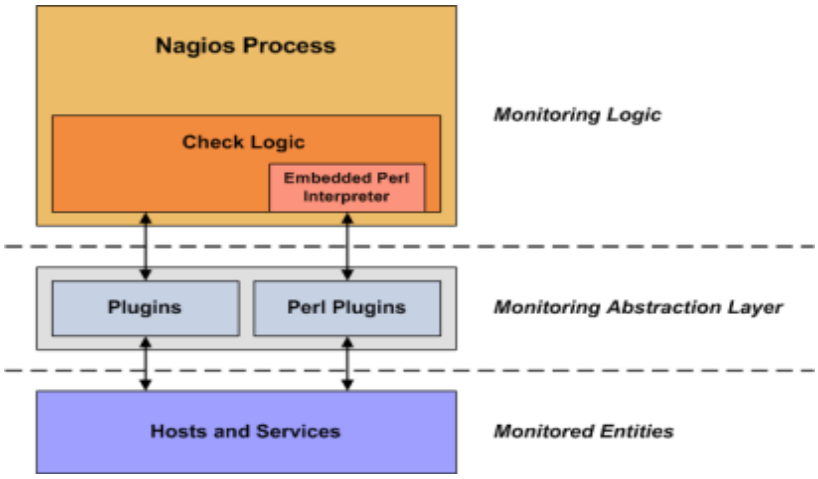
3.3.2 Zásuvné moduly - Pluginy

Samotné jadro Nagiosu nevie kontrolovať služby a ani nevie oznamovať ich zmeny, ale kontrolu necháva tzv. zásuvným modulom.

[4] http://www.mail-archive.com/nagios-users@lists.sourceforge.net/msg30193.html

Zásuvné moduly sú veľkosťou malé, nezávislé skripty, ktoré sa používajú na kontrolu služieb na vzdialenom hostiteľovi. Môžu byť vo forme Perl skriptu, Shell skriptu, a spúšťajú sa z príkazového riadka. Výstup z vykonania by mal byť vždy presmerovaný na STDOUT (Standard output), čiže to, čo vypíše zásuvný modul na terminál. Výstupný reťazec by nemal mať viacej ako 80 znakov.

Zásuvné moduly nie sú distribuované z jadrom programu, ale sa stiahnuť z oficiálnej stránky programu, alebo zo stránok dobrovoľných vývojárov zásuvných modulov.[5]



Obr. 5. Blokové schéma začlenenia zásuvných modulov v architektúre Nagios. [10]

3.3.2.1 Návratové kódy

Nagios vyhodnocuje stav hostiteľa alebo jeho služby návratovými kódmi zo zásuvných modulov. Nasledujúca tabuľka (*Tab. 2*) poukazuje na zoznam návratových kódov, spolu zo statusom služby, alebo hostiteľa [10].

Tab. 2. Návratové kódy z zásuvného modulu.

Návratový kód zásuvného modulu	Status služby	Status hostiteľa
0	OK	UP
1	WARNING	UP alebo DOWN
2	CRITICAL	DOWN/UNREACHABLE
3	UNKNOWN	DOWN/UNREACHABLE

[5] http://nagiosplug.sourceforge.net/

3.3.2.2 Príklad použitia zásuvného modulu

mirino@ubuntu:/usr/local/nagios/libexec$ ***./check_icmp*** *localhost -w 100.0,20% -c 500.0,60%*

OK - localhost: rta 0.129ms, lost 0%|rta=0.129ms;100.000;500.000;0; pl=0%;20;60;; rtmax=0.280ms;;;; rtmin=0.077ms;;;;

Ako príklad bol použitý zásuvný modul *check_icmp*, pomocou ktorého zisťujeme odozvu hostiteľa alebo služby pomocou ICMP paketu. Posiela sa 5 paketov a z ich výsledných hodnôt odoziev sa vypočíta priemerná hodnota a určí sa výsledný stav statusu.

V syntaxe príkazu boli použité dva parametre:

- -w *100.0,20%* (ak priemerná hodnota odoziev paketov je minimálne 100 ms, alebo sa tratí 20% paketov, čo je v tomto prípade 1 paket, tak zásuvný modul vráti návratovú hodnotu WARNING)
- *-c 500.0,60% (*ak priemerná hodnota odoziev paketov je minimálne 500 ms, alebo sa tratí 60% paketov, čo sú v tomto prípade 3 pakety, tak zásuvný modul vráti návratovú hodnotu CRITICAL*)*
- Hodnoty pod 100ms = návratový hodnota OK [10]

3.4 Konfiguračné súbory

Nástroj Nagios je veľmi rozsiahly monitorovací systém a preto je konfigurácia rozdelená do viacerých menších súborov, ktoré sú zadefinované v hlavnom konfiguračnom súbore *nagios.cfg*, uloženého v adresári /usr/local/nagios/etc. V tomto súbore sa nachádzajú aj odkazy na súbory, kde sú zadefinované informácie o objektoch, ktoré sú používané v Nagios.

Tab. 3. Objekty používane v Nagios a ich konfiguračný súbor.

Konfiguračný súbor	Definícia
cfg_dir=/etc/nagiosql/hosts	zariadenia, ktoré budú monitorované
cfg_dir=/etc/nagiosql/services	monitorované služby
cfg_file=/etc/nagiosql/contacts.cfg	kontakty, použité pri notifikácii
cfg_file=/etc/nagiosql/timeperiods.cfg	časový interval odosielania notifikácie
cfg_file=/etc/nagiosql/commands.cfg	príkazy, ktoré spúšťajú akcie
cfg_file=/etc/nagiosql/contactgroups.cfg	definovanie kontaktných skupín

cfg_file=/etc/nagiosql/hostgroups.cfg	definovanie skupín pre hostiteľov
cfg_file=/etc/nagiosql/servicegroups.cfg	definovanie skupín pre služby

3.4.1 Definovanie zariadení (hosts)

V tomto konfiguračnom súbore sa definujú monitorované zariadenia, napr. pracovné stanice, servery, tlačiarne alebo zariadenia, ktoré majú pridelenú adresu IP, alebo doménové meno.

V prvých troch parametroch sú zadefinované menné a kontaktné údaje monitorovaného zariadenia, nasledujúci parameter *check_command* popisuje, aký zadefinovaný príkaz, ktorého definícia je popísaná v konfiguračnom súbore *commands.cfg* sa má spustiť. Ďalším parametrom je *max_check_attempts*, ktorý udáva maximálny počet opakovaní testu, pokiaľ nebude zariadenie vyhodnotené s chybovým hlásením. Ak by už nastala taká udalosť, tak parametrom *retry_interval* sa určí hodnota v minútach za aký čas budú vykonávané ďalšie testy, ináč sú testy vykonávané každých 5 minút o čom hovorí parameter *check_interval*.

Posielanie oznámení bude každých 120 minút, v režime 24 hodín a 7 dní v týždni, kontaktnej skupine admin, v prípade že zariadenie je v stave DOWN – (d), nedostupné v stave UNREACHABLE – (u), alebo v stave RECOVERY – (r), ak už začalo komunikovať. Posledným nepovinným parametrom sa nastaví obrázok, ktorý bude prislúchať danému zariadeniu pri jeho grafickom vyobrazení.

Príklad:

define host {

host_name	localhost_UBUNTU
alias	localhost UBUNTU (Vmware)
address	127.0.0.1
check_command	*check-host-alive*
max_check_attempts	*10*
check_interval	*5*
retry_interval	*1*
check_period	*24x7*

contact_groups *admins*

notification_interval *120*

notification_period *24x7*

notification_options *d,u,r*

icon_image ubuntu.png

}

3.4.2 Definovanie testovanej služby (services)

Nastavovanie parametrov testovanej služby je veľmi podobné ako pri konfigurácii definície zariadení, s tým rozdielom, že už na uvedenom príklade nie sú definované všetky parametre (oznamovacie, kontrolné parametre atď.), ale je využitá šablóna (local-service), ktorá má v sebe tieto parametre obsiahnuté. Táto možnosť veľmi uľahčí a sprehľadní prácu pri konfigurácii, pretože odpadá prácnosť neustáleho vypisovania najčastejšie používaných parametrov.

Príklad:

define service {

host_name localhost_UBUNTU

service_description Root Partition

use local-service

check_command check_local_disk!20%!10%!/

register 1

}

V tabuľke (*Tab. 4*) sú znázornené možnosti pri nastavovaní oznamovacích pravidiel pri definícii služieb.

Tab. 4. Hodnoty parametra notification_options pri definícii služieb.

Parameter ***notifcation_options***	
Hodnota parametra	**Význam**
w	varovný stav služby - WARNING - odoslanie oznámenia
c	varovný stav služby - CRITICAL - odoslanie oznámenia
r	varovný stav služby - RECOVERY - odoslanie oznámenia

u	varovný stav služby - UNREACHABLE - odoslanie oznámenia
n	oznámenia nebudú posielané

3.4.3 Definovanie kontaktov (contacts.cfg)

V tomto konfiguračnom súbore sa definujú kontaktné údaje o osobách, ktoré majú byť informované o vzniknutých udalostiach. V príklade je zadefinovaný užívateľ *nagiosadmin,* ktorý bude nonstop informovaný o poruchách a informácia mu bude poslaná na emailovú adresu a mobilný telefón.

Príklad:

define contact {

contact_name nagiosadmin

alias *Nagios Admin*

email *mirino@localhost*

pager *+421911xxxyyy*

host_notification_period *24x7*
service_notification_period *24x7*
host_notification_options *d,u,r,s*
service_notification_options *w,u,c,r,s*
host_notification_commands *notify-host-by-email,notify-host-by-sms*
service_notification_commands *notify-service-by-email,notify-service-by-sms*

}

3.4.4 Definovanie časových intervalov odosielania notifikácie (timeperiods.cfg)

V tomto kroku sa definujú časové intervaly, počas ktorých bude monitorovací systém generovať a odosielať oznámenia osobám, ktoré boli zadefinované v contacts.cfg. V príklade možno vidieť už prednastavené nastavenie po inštalácii, z ktorého je jednoduché vyčítať, že oznámenia sa budú posielať nonstop.

Príklad:

define timeperiod {

timeperiod_name 24x7

alias	24 Hours A Day, 7 Days A Week
friday	00:00-24:00
thursday	00:00-24:00
wednesday	00:00-24:00
tuesday	00:00-24:00
monday	00:00-24:00
sunday	00:00-24:00
saturday	00:00-24:00

}

3.4.5 Spôsoby editácie súborov

3.4.5.1 Editácia súborov manuálne

Mnoho užívateľov kopíruje štandardne priložené predlohy konfiguračných súborov a potom ich modifikuje podľa potreby s textovým editorom. Na linuxovej platforme je možné využiť populárny editor Vi, Pico a na platforme Windows aplikáciu Notepad [11].

Pri vytváraní a editácii konfiguračných súborov treba dodržať následné pravidlá“

- Riadky, ktoré začínajú znakom „**#**“, sú považované za komentár a nebudú spracované.
- Znaky, ktoré nasledujú po znaku „**;**“, sú považované za komentár a nebudú spracované.

3.4.5.2 Editácia súborov grafickými nástrojmi cez Front End

Kvôli zväčšeniu komfortu pri konfigurovaní a administrácii Nagiosu, bol vyvinutý nástroj NagiosQL[6]. Je to nástroj založený na webovom serveri s podporou PHP[7]

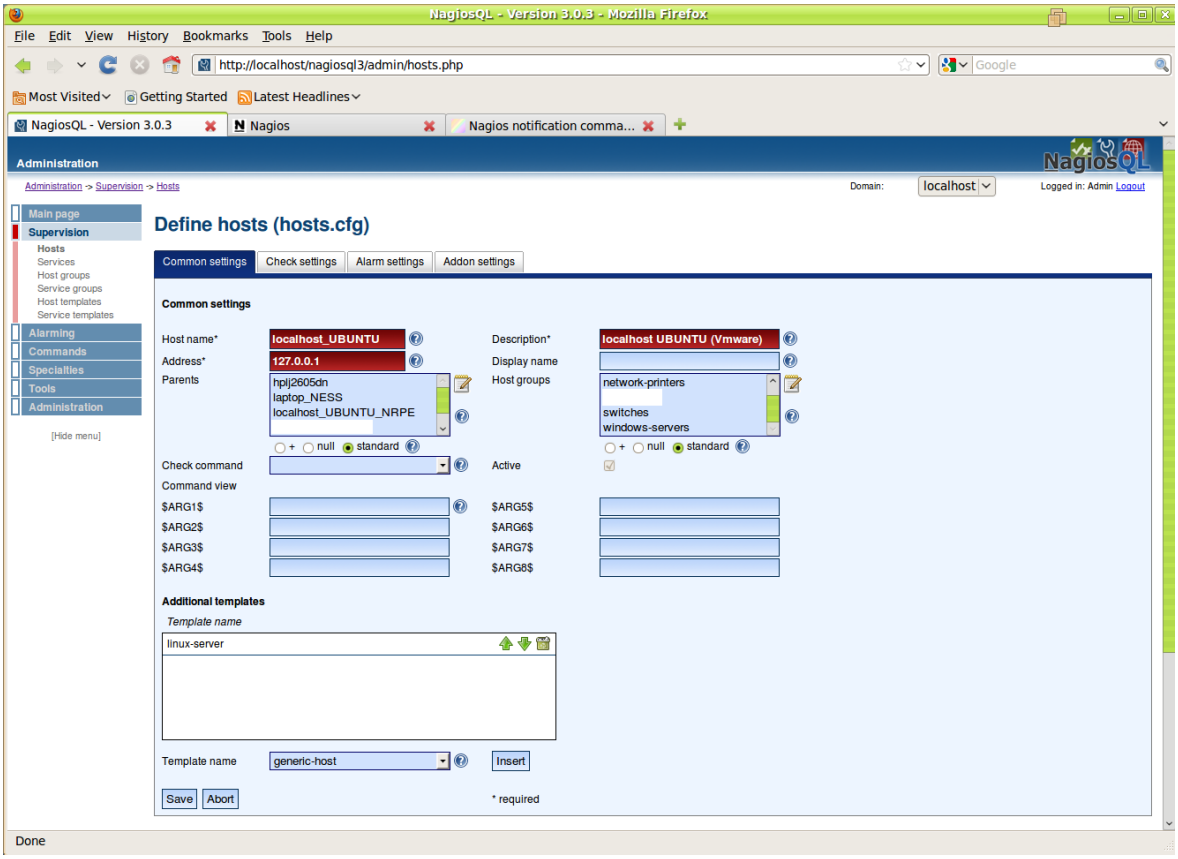
[6] http://www.nagiosql.org/

[7] http://php.net/

(Hypertext Preprocessor) a databáze MySQL[8] (My Structured Query Language), ktorý je nápomocný pri tvorbe komplikovaných nastavení.

Jeho hlavnými črtami sú:

- Vytváranie, mazanie, upravovanie a kopírovanie nastavení.
- Vytváranie a export konfiguračných súborov.
- Jednoduchá automatická záloha konfiguračných súborov.
- Kontrola konzistencie a syntaxu konfiguračných súborov.
- Podpora databázovej platformy.
- Správa užívateľov [17].



Obr. 6. Ukážka Front End konfiguračného nástroja NagiosQL.

[8] http://www.mysql.com/

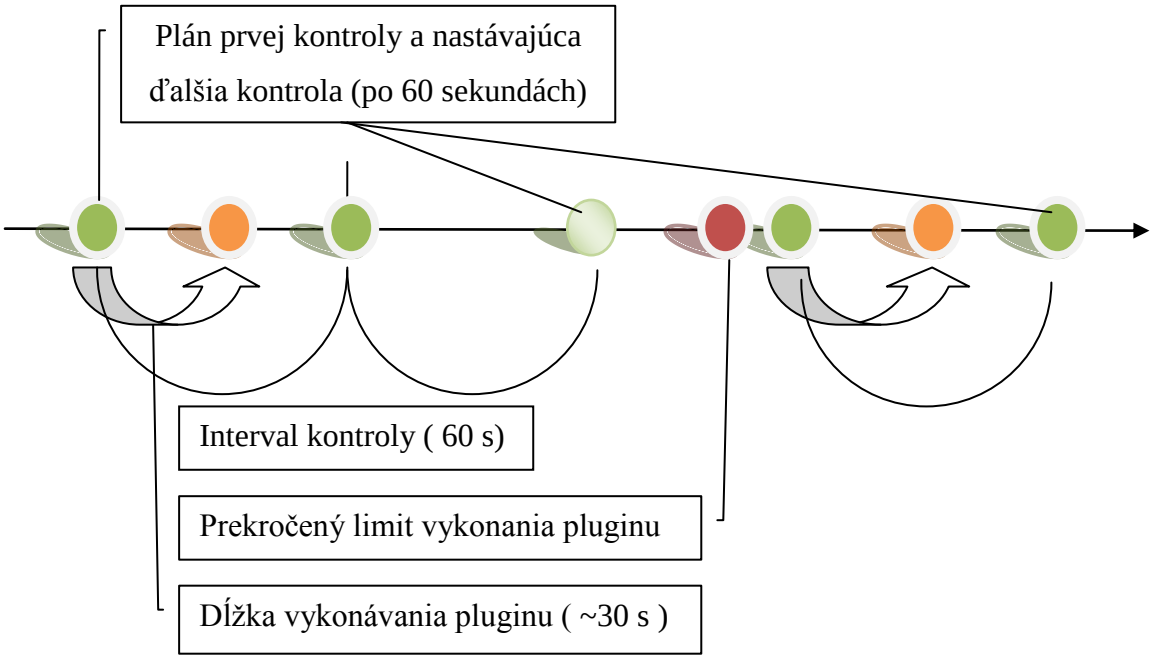
3.5 Plánovanie testov

Jadro Nagiosu obsahuje šikovný plánovač, s mnohými užívateľsky nadefinovateľnými možnosťami, pomocou ktorých je možné ovplyvniť jeho úlohy. Aby tieto úlohy boli efektívne vykonávané a spolupracovali so svojím okolím, tak je dôležité pochopiť, ako plánovač pracuje.

3.5.1 Interval kontroly

Všetky interné procesy Nagiosu, vrátane hostiteľských kontrol a kontrol servisov, sú umiestnené v tzv. globálnej fronte udalostí. Naplánovanie kontrolných akcií je možné pomocou užívateľských definícií, ale nie použitím zadania absolútneho dátumu, alebo času v crone (Unix, Linux), alebo v plánovači úloh (Windows). Je to dané z dôvodu, pretože Nagios nevie kontrolovať, ako dlho bude vykonávaný monitorovací program (zásuvný modul). Miesto toho, sa môže Nagiosu povedať, ako dlho má čakať, pokiaľ bude ukončený a má sa spustiť znova.

Je potrebné poznamenať, že interval kontrol stačí zadefinovať iba pri kontrole servisov. Je možnosť ho zadať aj pri špecifikovaní intervalu kontrol cieľového hosťa. To ale nie je potrebné z dôvodu, že hostiteľské kontroly sa zvyčajne vykonávajú až po zlyhaní kontroly služby. Pokiaľ nejdú služby, tak je predpoklad, že hostiteľ je nedosiahnuteľný. Prednastavená dĺžka intervalu kontroly je nastavená na 60 sekúnd.



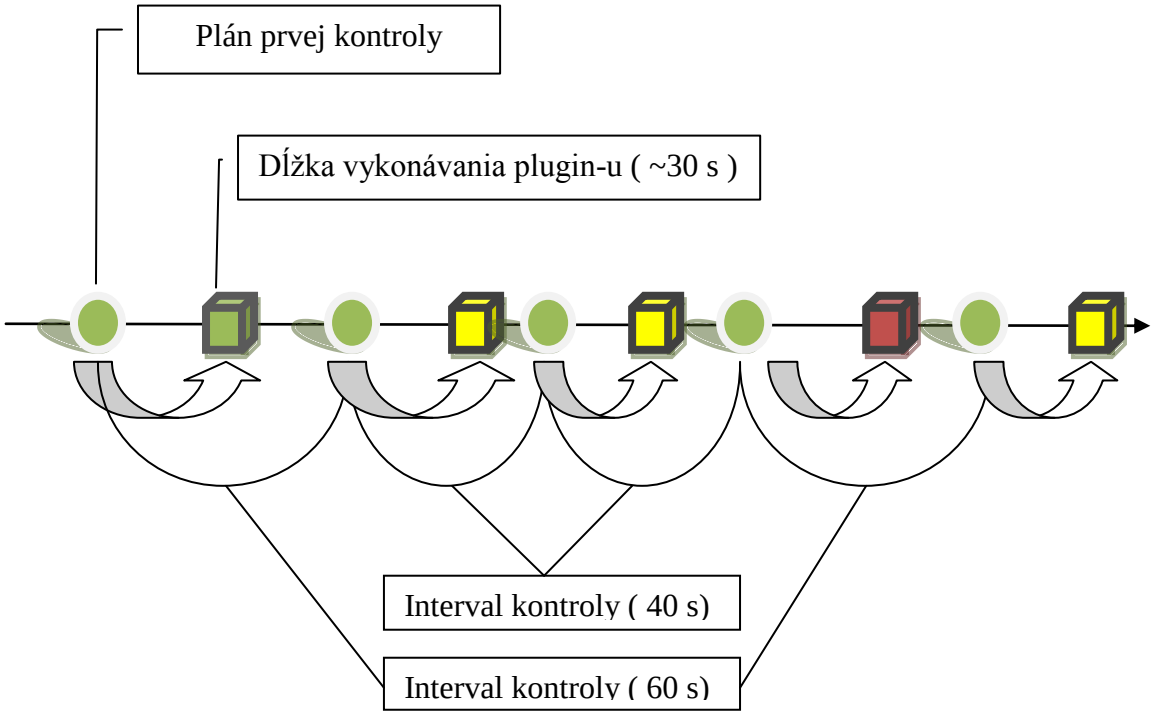
Obr. 7. Interval kontroly.

Na obrázku (

Obr. 7) je znázornené ako je naplánovaná globálna fronta udalostí. Každá udalosť je vložená s časovou pečiatkou definujúcou v akom čase sa má kontrola spustiť. Potom, čo Nagios spustí zásuvný modul, tak čaká na návratovú hodnotu úspešného vykonania. Ak prebehne všetko v poriadku, tak sa štandardne naplánuje ďalšia kontrola podľa definovaného časového intervalu opakovania. Môže nastať situácia, kedy zásuvný modul je vykonávaný dlhšiu dobu a prekročí tento interval opakovania. Keďže spustenie bude už v minulosti, tak Nagios jednoducho preplánuje ďalšie spustenie zásuvného modulu o krátky čas neskôr. Príklad: veľká vyťaženosť servera, alebo dlhé odozvy na sieti [1].

3.5.2 Interval kontroly po vyhodnotení statusu

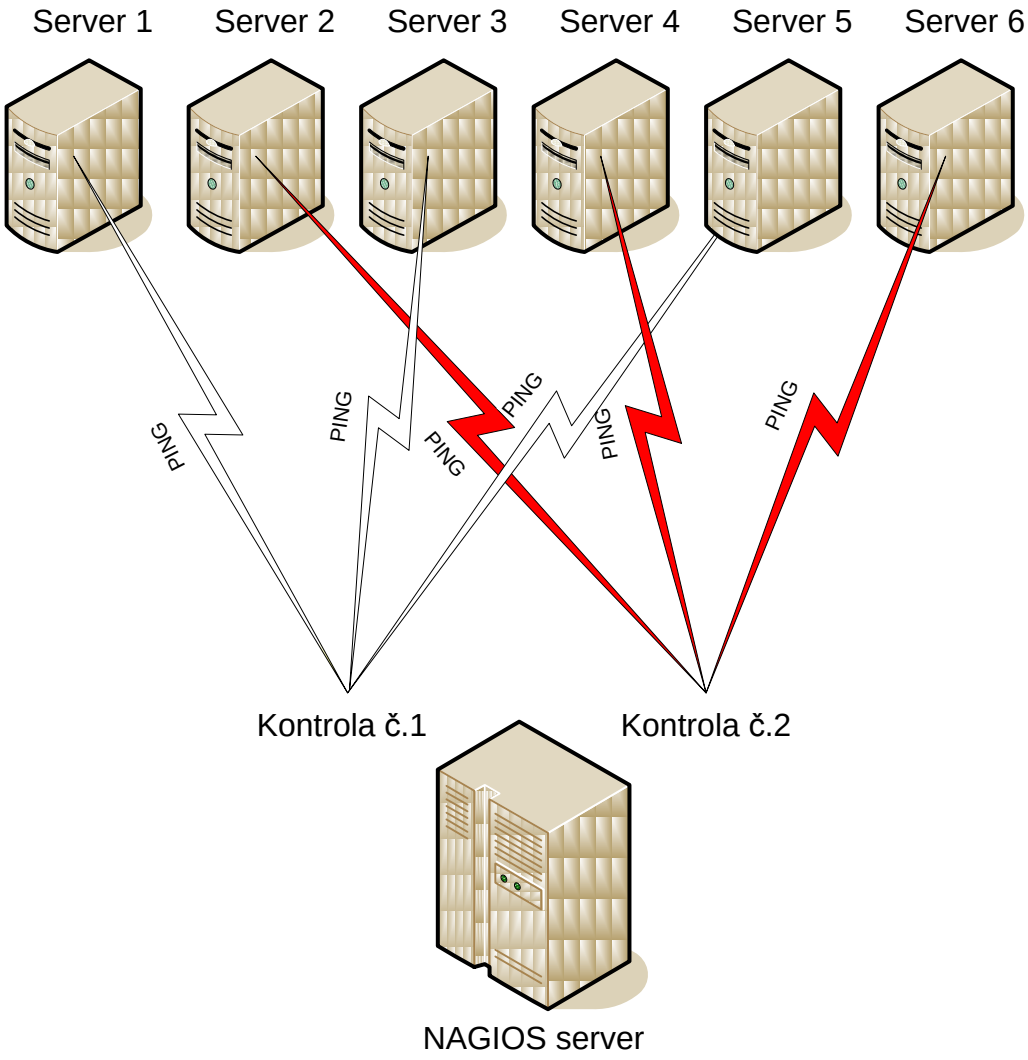
V prípade, že kontrola servisu skončí inak, ako s návratovým kódom 0 (OK), tak Nagios preplánuje interval kontroly na inú hodnotu ako je pôvodných 60 sekúnd. Takéto nastavenie je označované ako *retry_check_interval*. Ako je načrtnuté na obrázku (*Obr. 8*), tak v prípade, že sa zistí nedostupnosť servisu (žltá kocka) (WARNING), Nagios , aby si bol naozaj absolútne istý, že servis je DOWN vykoná ešte ďalšie tri opakované kontroly a až potom označí servis za nedostupný a posiela notifikáciu emailom, alebo SMS. Samozrejme aj maximálny počet opakovaní, ktoré sa majú vykonať je možné nakonfigurovať [2].



Obr. 8. Interval kontroly po vyhodnotení statusu.

3.5.3 Rozloženie záťaže

Pri štartovaní Nagiosu sa obvykle načítava dlhý zoznam hostiteľov a služieb. Prvoradou úlohou nastáva, čo najrýchlejšie zistiť stav každého prvku tak , ako je to možné. Teoreticky by to malo prebiehať od začiatku zoznamu na koniec a potom opäť na začiatok. Táto voľba však nie je optimálna cesta, pretože generuje priveľké zaťaženie na vzdialenom hostiteľovi. Ak server, ktorý je na vrchole zoznamu a má nakonfigurovaných napr. 15 služieb, tak by sa Nagios snažil skontrolovať postupne všetky naraz. Namiesto toho, sa používa tzv. *interleave factor*, ktorý je znázornený na (*Obr. 9*).



Obr. 9. Rozloženie záťaže pomocou parametra - interleave factor.

Na modelovom príklade je šesť serverov a chceme monitorovať iba ich dostupnosť pomocou programu PING, čiže iba jednu službu. Ak budeme mať nastavený *interleave factor=2*, tak pri prvej kontrole sa bude monitorovať server 1,3,5 a pri druhej kontrole 2,4,6. Týmto spôsobom je záťaž rozdistribuovaná a spôsob je výhodnejší ako metóda zhora-nadol. Interleave factor je užívateľsky definovateľný a je ho možné vypočítať podľa nasledujúceho vzorca.

Interleave factor = (celkový počet služieb / celkový počet hostiteľov)

Takýto spôsob odľahčenia záťaže na vzdialených hostiteľoch však neodľahčí server, kde je nainštalovaný Nagios, ktorý musí po spustení spracovávať, odosielať a prijímať veľké množstvo kontrol. Intenzívna záťaž sa dá po prvých pár minút po štarte obmedziť vložením malej časovej odmlky medzi kontrolami, ktoré by ináč boli vykonávané súbežne. Táto časová perióda sa *inter-check delay*.

inter-check delay = (priemerný kontrolný interval pre všetky služby) / (celkový počet služieb)

3.6 Monitorovacie protokoly

3.6.1 Protokol SNMP

Protokol SNMP (Simple Network Management Protocol), ktorý bol vytvorený na vzdialenú správu zariadení počítačovej siete. SNMP je štandard používaný pre správu sietí. Je to aplikačný protokol, ktorý poskytuje služby správy nad UDP protokolom. UDP protokol je ale nespoľahlivý, takže nie je záruka, že pakety nebudú pri posielaní stratené [7]. SNMP je založený na modely klient/server. Klientsky program nazývaný sieťový manažér vytvára virtuálne spojenia so serverom. SNMP agent beží na sledovanom sieťovom zariadení. Agent monitoruje stav zariadenia a poskytuje o ňom informácie manažérom. Informácie poskytované agentom sú usporiadané podľa databázy MIB (Management Information Base), ktorá svojou štruktúrou zodpovedá danému zariadeniu.

Výhodou tohto riešenia je, že nie je potrebné heslo do privilegovaného režimu. Na prístup k zariadeniu sa používa komunitný reťazec (community string), ktorý je možné definovať iba na čítanie. Tento reťazec sa posiela v nezašifrovanej podobe, čo však je tiež problém, ale oveľa menší ako pri posielaní hesla cez telnet, pretože týmto reťazcom dokážeme informácie iba čítať. Prípadné odchytenie tohto reťazca nebude mať také následky ako by to bolo v prípade odchytenia hesla do privilegovaného režimu. Existujú však metódy na zabránenie zneužitia tohto reťazca. Sú nimi napríklad definovanie zoznamu IP adries, ktoré majú právo čítať tieto údaje zo sieťových zariadení, alebo vytvorenie samostatnej siete pre správu zariadení. Nevýhodou tejto metódy je nižšia rýchlosť vzhľadom na to, že SNMP klient na zariadeniach beží s takmer najnižšou prioritou. Ďalšou nevýhodou je iba obmedzené množstvo štandardizovaných informácií [12].

4.1.1 Vznik SNMP

Protokol SNMP vznikol na konci 80-tych rokov za účelom správy smerovačov na pracovnom výbore IAB (Internet Architecture Board). Bol vyvinutý ako jeden variant protokolu SGMP (Simple Gateway Monitoring Protocol), ktorý bol navrhnutý koncom roku 1987 práve pre výmenu informácií medzi smerovačmi a bránami v akademickej sieti. Druhým variantom, ktorý vznikol z protokolu SGMP na pôde organizácie ISO (International Organization for Standardization), bol protokol CMIP (Common Management Information Protocol). Aj keď CMIP bol pokusom ISO o vytvorenie štandardu s maximálnou možnou podporou protokolov a služieb s definovanou databázovou štruktúrou pre prenos pomocou protokolu TCP/IP, tak nenašiel väčšiu podporu u výrobcov ani používateľov a nedočkal sa významnejšieho rozšírenia.

Na druhej strane SNMP protokol (rovnako ako súbor protokolov TCP/IP) preukázal veľkú životaschopnosť. Relatívna jednoduchosť a implementácie ho urobila populárnym, pretože presne splňoval požiadavky na narastajúce potreby sieťovej správy. Od mája roku 1989, kedy bol formalizovaný, sa stal štandardom pre správu sietí, ktoré sú založené na protokole TCP/IP, slúžiaci ako jednotný prostriedok pre správu konfigurácie, kapacity, bezpečnostného zaistenia sietí a všetkých zariadení siete.

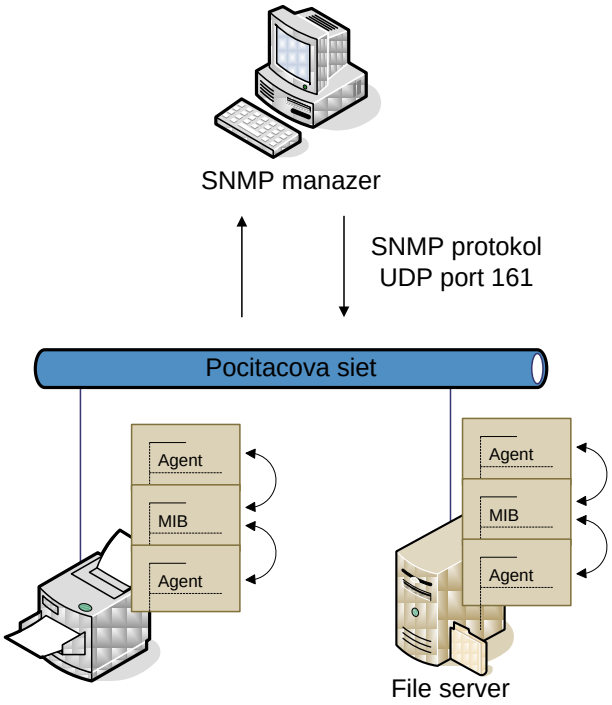
V súčasnej dobe existuje špecifikácia štandardu SNMPv1 a jeho rozšírenia - SNMPv2c a SNMPv3. Dôvodom pre návrh nových verzií protokolu SNMP bola hlavne nedokonalosť prvej verzie z hľadiska bezpečnosti [12].

Je dôležité poznamenať, že verzii SNMPv1,2 sa pri autentifikácii posielanie prihlasovacieho mena a hesla posiela vo forme čistého textu, tzn. nie je šifrované. Vo verzii SNMPv3 je už tento mechanizmus upravený a komunikácia je šifrovaná, takže potenciálny útočník sa už nedostane k údajom a k privilegovanému prístupu. Nevýhodou je, že veľa zariadení túto novú verziu nepodporuje. Možnosť, ako ostať chránený je buď zrušiť niektoré, alebo všetky SNMP prístupy, alebo použitím firewallu filtrovať SNMP sieťovú prevádzku z nedôveryhodných zdrojov [7].

4.1.2 Architektúra SNMP

SNMP komunikácia je založená na modely Manažér - Agent a umožňuje prenos komunikácie medzi správcom siete - manažérom a agentmi na jednotlivých sieťových zariadeniach. K tomu sa vyžaduje, aby každé zariadenie siete poskytovalo základné informácie o sebe samom a rovnako tak uľahčilo pridávanie ďalších informácií, špecifických pre dané zariadenie. Tieto základné a prídavné informácie sa spolu nazývajú MIB (Management Information Base). MIB je dátová hierarchická stromová štruktúra, ktorá zodpovedá danému konkrétnemu zariadeniu [13].

Protokol definuje dva typy sieťových zariadení - Manažérov a Agentov:



Obr. 10. Architektúra SNMP.

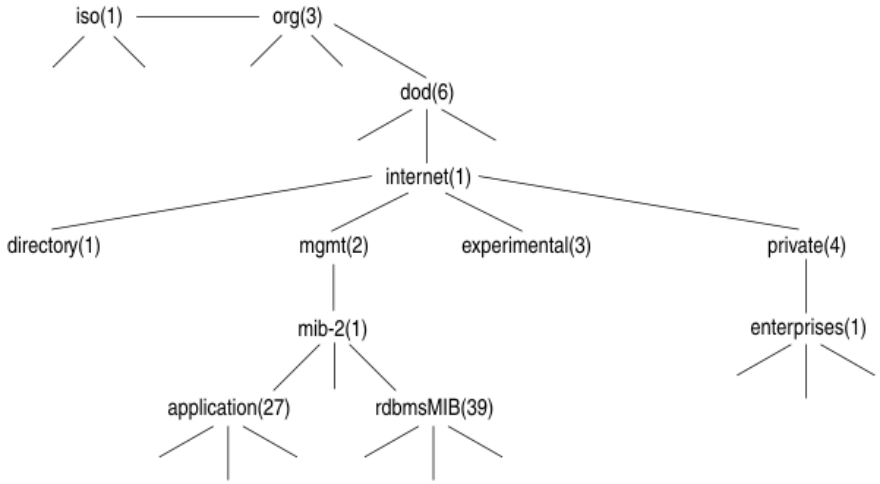
SNMP manažér - je program, ktorý beží na sieťovej stanici. U väčších systémov ide väčšinou o vyhradenú výkonnú pracovnú stanicu, na ktorom beží software NMS (Network Management Software). Funkcia tohto SNMP Manažéra potom spočíva v dotazovaní sa jednotlivých SNMP Agentov pomocou SNMP operácií. Zmyslom je získať všetky potrebné informácie o danom zariadení, ktoré agent reprezentuje. SNMP manažér poskytuje väčšinou grafické rozhranie , ktoré umožňuje prezentáciu získaných dát, sledovanie sieťových alarmov [13].

SNMP agent - je malý program bežiaci na sieťovom zariadení, ktoré ho reprezentuje a odpovedá na požiadavky SNMP Manažéra. Agent preto neustále monitoruje a zbiera

informácie o všetkých dostupných funkciách a stavoch daného zariadenia. K získaniu informácií o danom zariadení, musí manažér vyslať požiadavku na dané zariadenie a prejsť informácie poskytované agentom. Musí prejsť celú stromovú štruktúru MIB až k objektu, ktorý obsahuje potrebné dáta, aby mohol získané informácie interpretovať [13].

4.1.3 Management Information Base

MIB opisuje sadu objektov, ktoré sú predmetom administrácie. Spravované zariadenie môže implementovať jednu alebo viac MIB, v závislosti na jeho funkcií. Tieto MIB databázy sú veľmi podobné štandardným databázam v tom zmysle, že opisujú ako štruktúru, tak aj formát dát. MIB sú napísane podľa pravidiel SMI (Structure of Management Information) [7]. V súčasnosti už existuje i štandard SMIv2, spätne kompatibilný s predchádzajúcou verziou. MIB je teda dátová hierarchická stromová štruktúra, ktorá zodpovedá danému konkrétnemu zariadeniu [14].



Obr. 11. Hierarchia MIB. [20]

Nasledujúci reťazec predstavuje úplnú cestu do tabuľky, stĺpca, hodnoty k aplikácii MIB. Takýto slovný zápis je ale zriedkavý.

iso.org.dod.internet.mgmt.mib-2.application

Častejšie sa používa číselný ekvivalent zápisu cesty, tzv. OID (Object Identifier), ktorý jednoznačne popisuje časť dát, ktorú SNMP manažér môže zo zariadenia získať. Je písaný ako reťazec čísel oddelených bodkami, čiže predchádzajúci slovný zápis bude zapísaný ako:

1.3.6.1.2.1.27

4.1.4 Typy SNMP objektov

SNMP objekty môžu byť v zásade dvoch typov - skalárne hodnoty a tabuľky. Objekty typu skalár môžu nadobúdať iba jednoduché neštruktúrované hodnoty. Jedná sa o niekoľko typov [14]:

• **Integer** - jednoduché celé číslo. Hoci špecifikácia nedefinuje žiaden limit, väčšina implementácií obmedzuje tento typ veľkosti na 32 bitov.

• **Counter** - nezáporný integer, ktorý sa plynulo zväčšuje, až dosiahne max. hodnoty (2^{32} - 1), potom začína znovu od nuly. Ako už meno napovedá, používa sa najmä na počítanie zaujímavých udalostí v systéme.

• **Gauge** - nezáporný integer, jeho hodnota môže vzrastať i klesať, nikdy ale nemôže prekročiť max. hodnotu. Hodnota Gauge je maximálna, kedykoľvek modelovaná informácia je väčšia alebo rovnaká než toto maximum. V prípade následného poklesu sa zníži i hodnota Gauge. Absolútna možná hodnota je opäť (2^{32} - 1).

• **TimeTicks** - nezáporný integer reprezentujúci v stotinách sekundy čas od istej doby (možná hodnota (2^{32} - 1). Môže byť použitý k vyjadreniu doby chodu nejakého zariadenia od jeho zapnutia.

• **IpAddress** - 32 bitová IP adresa.

• **OCTET STRING** - sekvencia bytov. Používa sa k vyjadreniu buď reťazca znakov, napr. meno systému, alebo ľubovoľných binárnych dát, napr. MAC (Media Access Control) adresy zariadenia.

• **OBJECT IDENTIFIER** - reprezentuje meno uzlu. SNMP dovoľuje ešte tri iné typy skalárnych hodnôt (NULL, Opaque a Network Address), ktoré sa však nepoužívajú [14].

4.1.5 Bezpečnosť prístupu

Dôležitou súčasťou SNMP komunikácie je systém zabezpečení prístupov k objektom. Ide vlastne o definovanie prístupových práv k jednému SNMP Agentovi z rôznych SNMP Manažérov. Systém je v princípe veľmi primitívny, každý príkaz obsahuje

v sebe komunitný reťazec (Community String), ktorý funguje ako kombinácia používateľského mena a hesla. Tieto príkazy ani reťazce nie sú šifrované. Z dôvodu tejto bezpečnostnej trhliny vznikla veria 3 protokolu SNMP, ktorá už podporuje šifrovanie.

Najpoužívanejší štandardný Community String u SNMP zariadení je "public" pre read-only prístup a "private" pre read-write prístup. Je teda potrebné dávať pozor iba na to, že tieto hesla rozlišujú veľké a malé písmena [15].

4.1.6 Podpora SNMP

Najväčšou výhodou SNMP je jeho široká podpora zo strany výrobcov sieťových zariadení. SNMP agenti sú dostupní pre celú škálu sieťových zariadení od počítačov, rozbočovačov, prepínačov cez smerovače až po modemy a tlačiarne. Už to, že sa dostalo SNMP tejto širokej podpore medzi výrobcami i používateľmi, svedčí o jeho schopnosti uspokojiť takmer všetky požiadavky administrátorov na sieťovú správu a o jeho práve na existenciu [9].

Navyše je SNMP veľmi flexibilný a rozšíriteľný protokol. Agentov je možné takmer ľubovoľne rozširovať, aby pokryli nové funkcie zariadení.

SNMP je jednoduchý nepotvrdzovaný protokol, používaný rôznymi výrobcami hardvéru a softvéru, ktorý umožňuje správcom mať prehľad o jednotlivých zariadeniach na sieťach LAN (Local Area Network) i WAN (Wide Area Network).

V súčasnej dobe neexistuje k SNMP protokolu rozumná alternatíva pre efektívnu správu počítačových sieti [16].

II. PRAKTICKÁ ČASŤ

4 INŠTALÁCIA

V tejto kapitole bude popísaný zjednodušený postup inštalácie dohľadového systému Nagios. Nagios je možné nainštalovať dvoma spôsobmi. Buď sa použije balíček, ktorý je vytvorený na konkrétnu distribúciu, alebo sa použije inštalácia z binárnych súborov. V tejto práci bude popísaný druhý spôsob a preto budú na kompiláciu potrebné tieto balíčky a knižnice: gcc, make, autoconf, automake.

Celý proces inštalácie bude vykonávaný na operačnom systéme linux UBUNTU 9.10 (Karmic Koala) nainštalovaný vo virtualizačnom prostredí Vmware Workstation v. 7.0.1 build-227600.

4.1 Stiahnutie inštalačných balíčkov

Z domovskej internetovej stránky http://www.nagios.org/download/ si je potrebné stiahnuť balíček jadra systému, ktorý pozostáva z démona a webového rozhrania a balíček, v ktorom sú obsiahnuté zásuvné moduly.

http://prdownloads.sourceforge.net/sourceforge/nagios/nagios-3.2.1.tar.gz

http://prdownloads.sourceforge.net/sourceforge/nagiosplug/nagios-plugins-1.4.14.tar.gz

Nagios pre svoju inštaláciu potrebuje iba niekoľko závislých balíčkov a väčšina z nich nie je povinná. V prípade použitia Web rozhrania je potrebné mať nainštalovaný Web server s podporou CGI (Common Gateway Interface) a na zobrazenie pekných generovaných obrázkov a grafov tri grafické knižnice: libpng, libjpeg, gd library.

Pri inštalácii zásuvných modulov je už potrebné mať doinštalovaných súčastí viac. Je potrebný program PING, niektoré BIND nástroje (Berkeley Internet Name Domain) ako HOST, DIG, NSLOOKUP, ďalej knižnice OpenSSL (Secure Sockets Layer) a PERL (Practical Extraction and Report Language). Na použitie dotazovania sa sieťových objektov pomocou SNMP, je potreba net-snmp, perl-snmp.

4.2 Vytvorenie potrebných užívateľov u skupín

Predtým, ako sa spustí samotná inštalácia, tak je potrebné manuálne vytvoriť dve skupiny *nagios*, *nagcmd* a užívateľa *nagios,* ktorý bude zaradený do týchto skupín.

mirino@ubuntu:~$ sudo –s

root@ubuntu:~# /usr/sbin/useradd -m -s /bin/bash nagios

root@ubuntu:~# passwd nagios

root@ubuntu:~# /usr/sbin/groupadd nagios

root@ubuntu:~# /usr/sbin/usermod -G nagios nagios

root@ubuntu:~# /usr/sbin/groupadd nagcmd

root@ubuntu:~# /usr/sbin/usermod -a -G nagcmd nagios

root@ubuntu:~#/usr/sbin/usermod -a -G nagcmd www-data

4.3 Kompilácia a inštalácia Nagios (jadro programu)

Z adresára, kde je rozbalený inštalačný balíček, spustiť konfiguračný skript

root@ubuntu:~# ./configure --with-command-group=nagcmd

Kompilácia zdrojového kódu

root@ubuntu:~# make all

Inštalácia binárnych súborov, inicializačných skriptov, ukážkových konfiguračných súborov a nastavenie oprávnení na adresár s externými príkazmi

root@ubuntu:~# make install

root@ubuntu:~# make install-init

root@ubuntu:~# make install-config

root@ubuntu:~# make install-commandmode

Tab. 5. Adresárová štruktúra Nagios.

Typy súborov	Umiestnenie
Konfiguračné súbory	/usr/local/nagios/etc
HTML súbory pre Web rozhranie, dokumentácia	/usr/local/nagios/share
skripty CGI	/usr/local/nagios/share
démon Nagios a vykonávateľné programy	/usr/local/nagios/bin

Logovacie súbory	/usr/local/nagios/var
Zásuvné moduly	/usr/local/nagios/libexec

4.4 Kompilácia a inštalácia Nagios (Zásuvné moduly)

Z adresára, kde je rozbalený inštalačný balíček, spustiť konfiguračný skript

root@ubuntu:~# ./ ./configure --with-nagios-user=nagios --with-nagios-group=nagios

Kompilácia zdrojového kódu

root@ubuntu:~# make

root@ubuntu:~# make install

5 MONITOROVANIE

5.1 MONITOROVANIE SAP SYSTÉMOV

Existuje niekoľko spôsobov monitorovania SAP (System analyse und Programmentwicklung) systémov. Najjednoduchšie je kontrola portov, na ktorých sú príslušné SAP systémy spustené. Zvyčajne sú to porty 3200/3300 pre čísla systémov 00 a pre čísla systémov 01 porty 3201/3301 atď. Takáto jednoduchá kontrola sa môže vykonať zásuvným modulom *check_tcp*, ale v prípade, že vnútorné služby v SAP zlyhajú, tak koncový užívateľ sa nebude môcť do systému prihlásiť aj keď porty budú dosiahnuteľné. Ak je potrebné testovať zložité interakcie medzi komponentmi SAP, tak je potrebná komunikácia na aplikačnej vrstve [2].

5.1.1 Kontrola pomocou programu sapinfo

Program *sapinfo* je súčasťou voliteľného balíka RFC-SDK (Remote Function Call - Software Development Kit) používaného pri rozhraniach RFC. Balík je možné stiahnuť z SAP portálu na adrese http://service.sap.com, avšak na prihlásenie je potrebné zákaznícke číslo.

Po rozbalení balíka do zvoleného adresára je možné pomocou programu sapinfo jednoduchým spôsobom zistiť rôzne základné informácie o SAP inštalácii, napr.:

- verziu SAP systému.
- verziu databázy a hostiteľa, na ktorom je nainštalovaná.

mirino@ubuntu:~$ cd /usr/local/sap/rfcsdk/bin

mirino@ubuntu:/usr/local/sap/rfcsdk/bin$ ./sapinfo as host=172.XX.XX.XX sysnr=40

SAP System Information

Destination hostname.ness.com_AIO_40
Host hostname.ness.com
System ID AIO
Database AIO
DB host hostname.ness.com
DB system ORACLE
SAP release 700
SAP kernel release 700
RFC Protokoll 011
Characters 1100
Integers LIT

Floating P. IE3
SAP machine id 560
Timezone 3600

5.1.2 Kontrola pomocou zásuvného modulu check_sap.sh

Zásuvný modul check_sap.sh je skript, ktorý je založený na programe sapinfo. Je obsiahnutý v balíku so zásuvnými modulami pre Nagios v adresári contrib. Keďže pri inštalácii nie je automaticky nainštalovaný, tak je ho potrebné manuálne nakopírovať do adresára s zásuvnými modulami :

/usr/local/nagios/libexec

Ďalej je potrebné upraviť premennú sapinfocmd v skripte check_sap.sh zadaním cesty k programu sapinfo :

sapinfocmd=' /usr/local/sap/rfcsdk/bin/sapinfo '

Zásuvný modul môže byť spustený dvoma možnosťami a to buď s argumentom:

- **as** (dotazovanie sa na aplikačný server)

  check_sap.sh as connect string system_number

- **ms** (dotazovanie sa na message server)

  check_sap.sh ms connect string SID logon_group

5.1.2.1 Kontrola aplikačného servera SAP

*mirino@ubuntu:/usr/local/nagios/libexec$/./check_sap.sh **as** 172.XX.XX.XX 40*

OK - SAP server hostname.ness.com_AIO_40 available.

5.1.2.2 Kontrola message servera SAP

*mirino@ubuntu:/usr/local/nagios/libexec$/./check_sap.sh **ms** AIO*

OK - SAP server hostname.ness.com_AIO_40 available.

5.1.3 Kontrola SAP pomocou CCMS

SAP system obsahuje svoj vlastný monitorovací systém, ktorý sa volá CCMS (Computing Center Management System), kde lokálny agenti, určený pre zber dát

zhromažďujú dáta z jednotlivých hostiteľov. CCMS nie je určený iba pre SAP systémy, ale dokáže sledovať aj externé aplikácie tretích strán.

Vývojári samozrejme mysleli aj na možnosť monitorovania CCMS a naprogramovali príslušné zásuvné moduly na zber dát pre Nagios.

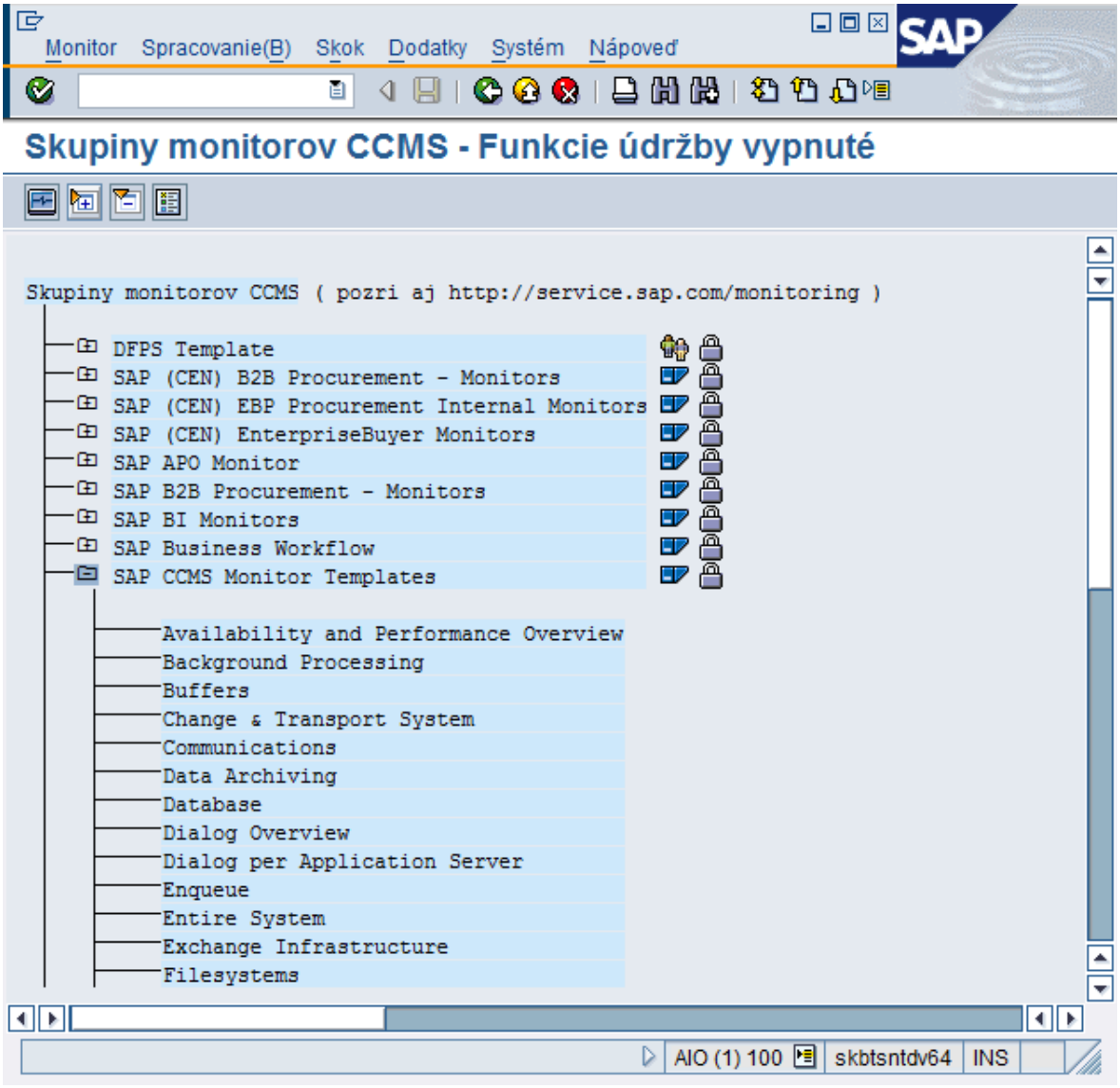
5.1.3.1 Všeobecne o CCMS

Je výkonný nástroj správcu systému, ktorý umožňuje vykonávať dohľad a komplexnú správu informačného systému spôsobom užívateľsky príjemným, opierajúc sa o všetky vymoženosti grafického užívateľského rozhrania. Podmodulom Alert monitor je správca systému pomocou semaforov upozorňovaný na neštandardné javy v systéme. Pomocou funkcií drill-down a drill-back umožňuje užívateľovi pohyb po rôznych informačných úrovniach systému. CCMS dovoľuje nielen analyzovať stavy minulé, ale aj predvídať možné kritické úzke miesta informačnej sústavy a upozorňovať na ne užívateľa.

CCMS integruje potrebné nástroje správy, monitorovania, riadenia a optimalizácie chodu komplexu technických aj programových prostriedkov, diagnostikovaním a nastavovaním výkonnostných parametrov systému na všetkých úrovniach:

CCMS podporuje, urýchľuje a automatizuje vykonávanie jednotlivých rutinných úloh, potrebných pre prevádzku systému na rôznych úrovniach informačného systému.

Na obrázku (*Obr. 12*) je znázornená základná obrazovka monitorovacieho nástroja v SAP – CCMS, ktorú je možné vyvolať pomocou spustenia transakcie RZ20. Monitorované spojenia sú prehľadne kategorizované v rôznych informačných skupinách.

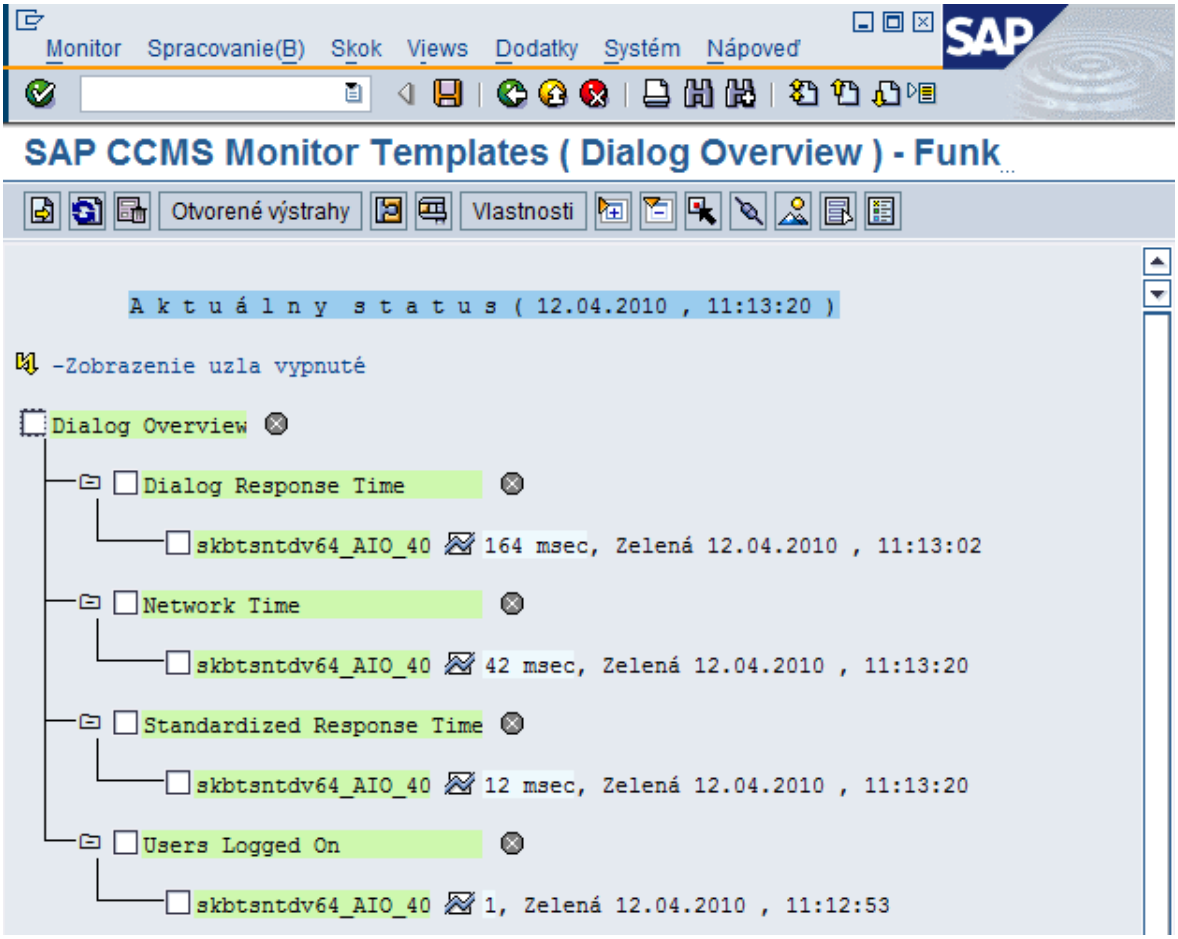


Obr. 12. Skupina monitorov v CCMS.

SAP vo svojich distribúciách ponúka niekoľko kolekcií monitorov už s prednastavenými hodnotami, ale ponúka aj možnosť, aby si administrátor mohol vytvoriť svoje vlastnú kolekciu monitorov.

Po rozkliknutí šablóny *Dialog Overview* sa zobrazia hodnoty atribútov *Dialog Response Time* dialógových odoziev tak ako je na obrázku (*Obr. 13*), ktoré poskytujú užitočné informácie o výkonnosti systému, presnejšie o priemernej dobe spracovania užívateľskej transakcie.

Ďalší atribút *Network Time* predstavuje, koľko času je potrebného pre systém, aby počas dialógového procesu posla dáta od klienta do SAP systému a späť.



Obr. 13. CCMS – aktuálny status dialógových procesov.

Väčšina meracích parametrov má nastavený limit varovania. Namerané hodnoty sú pre rýchlejší prehľad a orientáciu zobrazované:

- Zelenou farbou, ak hodnoty sú pod touto limitnou hranicou.
- Žltou farbou ak je limit mierne prekročený.
- Červenou farbou, ak je prekročená kritická hranica.

V nasledujúcich podkapitolách bude opísaný postup zásuvného modulu pre Nagios, ktorý pomocou dotazov na CCMS zbiera statusy zadefinované v CCMS:

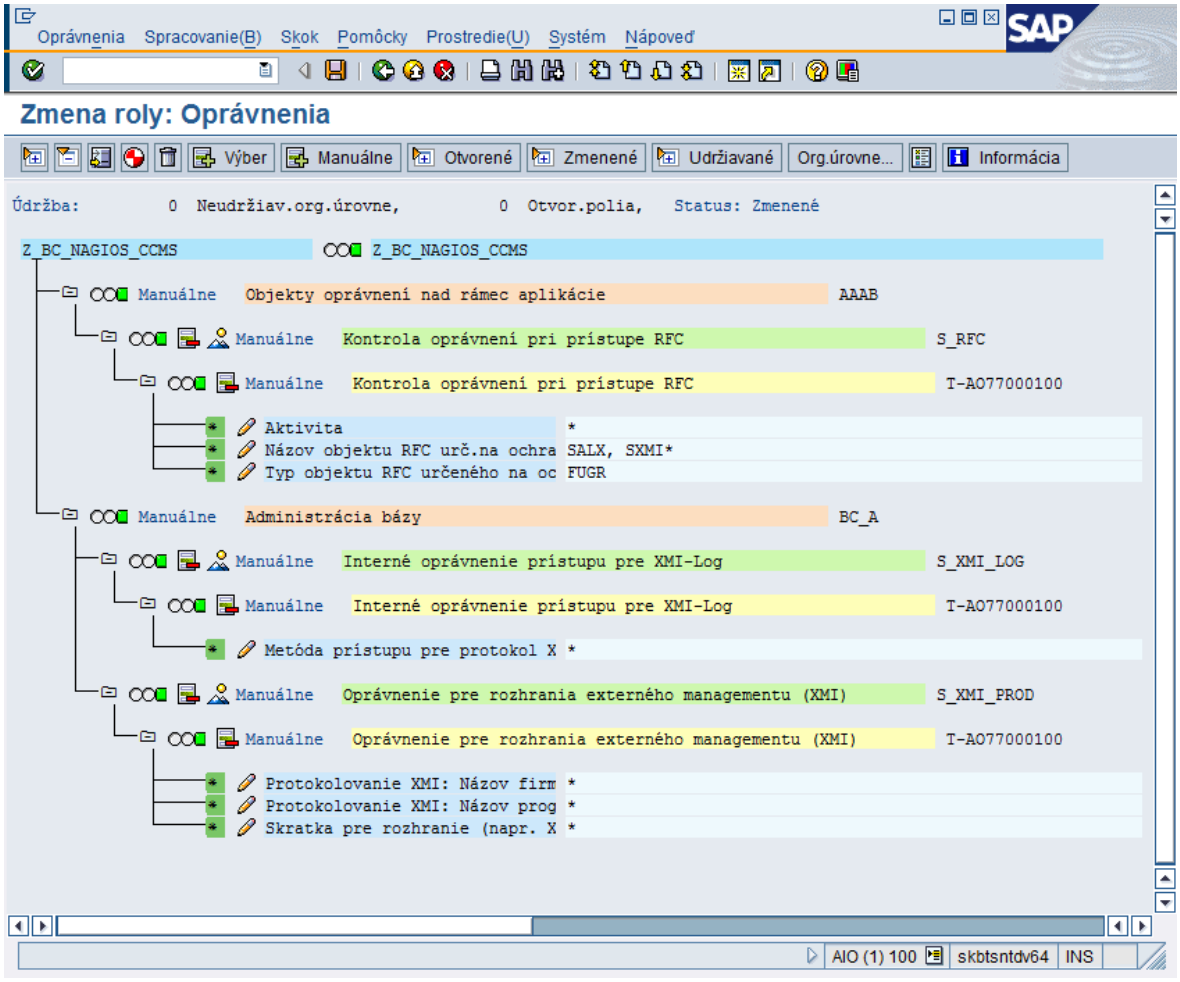
- OK – zelená.
- WARNING – žltá.
- CRITICAL = červená.

5.1.3.2 Zadefinovanie potrebných oprávnení.

Aby bolo možné načítanie informácií z CCMS do prostredia Nagios, tak je potrebné nastaviť oprávnenia na prístup do SAP. Načítanie dát sa vykonávané externými zásuvnými modulami pomocou RFC volaní, ktoré ale vyžadujú prihlásenie do SAP systému. Výhodou je, že nie je potreba nastavovať zložité pravidlá pre oprávnenia, ale užívateľ si vystačí s minimálnymi právami.

Role v SAP sa nastavujú pomocou generátora rolí. Po spustení transakcie PFCG (Profile Generator). Je potrebné vytvoriť rolu, ktorá bude obsahovať oprávnenia na tieto objekty, tak ako je znázornené na obrázku (*Obr. 14*) :

- S_RFC
- S_XMI_LOG
- S_XMI_PROD



Obr. 14. Autorizačné objekty potrebné na prístup z Nagios do SAP systému.

5.1.3.3 Inštalácia a konfigurácia Nagios CCMS zásuvného modulu.

Inštalačný súbor Nagios CCMS zásuvných modulov je možné stiahnuť z jeho domovskej stránky:

http://sourceforge.net/projects/nagios-sap-ccms/

Inštalácia je jednoduchá. Po rozbalení balíka je potrebné z jeho adresára *../src* spustiť príkaz *make*.

tar xzf sap-ccms-plugin-0.8.0
cd sap-ccms-plugin-0.8.0/src
make[9]

Po úspešnej kompilácii vzniknú v adresári sap-ccms-plugin-0.8.0/config tieto konfiguračné súbory, ktoré treba nakopírovať do adresára */etc/sapmon* a urobiť ich prístupnými pre užívateľa *nagios*.

- agent.cfg (obsahuje šablóny pre prístup k atribútom CCMS)
- login.cfg (obsahuje šablóny pre pripojenie k CCMS, prihlasovacie údaje)

A novovzniknuté zásuvné moduly nakopírovať do adresára */usr/local/nagios/libexec*

- check_sap
- check_sap_cons
- check_sap_cpu_load
- check_sap_instance
- check_sap_instance_cons
- check_sap_multiple
- check_sap_mult_no_thr
- check_sap_system

[9] Pri použitej distribúcii Ubuntu 9.10 Karmic po spustení príkazu *make* vznikla chyba, že kompilátor nemôže nájsť knižnicu *libstdc++.so.5*. Riešenie: http://bootstrapping.wordpress.com/2009/11/25/missing-libstdc-so-5-in-ubuntu-9-10-karmic/

- check_sap_system_cons

Príklad konfiguračného súboru */etc/sapmon/**agent.cfg*** :

[TEMPLATE_99]

DESCRIPTION=Dialog response time

MONI_SET_NAME=SAP CCMS Monitor Templates

MONI_NAME=Dialog Overview

*PATTERN_0=**

Príklad konfiguračného súboru */etc/sapmon/**login.cfg*** :

[LOGIN_AIO]

LOGIN=-d AIO -u user -p heslo -c 100 -h host.ness.com -s 40

Cieľom prepojenia SAP CCMS s Nagiosom je predovšetkým zobrazenie informácií o dostupnosti SAP systému v prostredí Nagios. Na tento účel je možné použiť CCMS zásuvný modul *check_sap_cons*[10], ktorý má nasledovnú syntax:

check_sap_cons <Template> <RFC Template>

- *Template :*číslo šablóny zapísanej v konfiguračnom súbore agent.cfg
- *RFC Template :* číslo šablóny zapísanej v konfiguračnom súbore login.cfg

Príklad:

mirino@ubuntu:/usr/local/nagios/libexec$ ./check_sap_cons 99 AIO

AIO host.ness.com _AIO_40 Dialog ResponseTime 1087 msec

AIO host.ness.com _AIO_40 Dialog FrontEndNetTime 33 msec

AIO host.ness.com _AIO_40 Dialog ResponseTime(StandardTran.) 13 msec

AIO host.ness.com _AIO_40 Dialog UsersLoggedIn 2

---

[10] Pri spustení príkazu vznikla chyba, že program nemôže nájsť knižnicu *sap_moni.so Riešenie: manuálne nakopírovať sap_moni.so* z adresára *../ nagios-sap-ccms/src/sap_moni* do adresára */usr/lib*

Obr. 15. Zobrazenie dostupnosti SAP systému.

5.2 MONITOROVANIE SYSTÉMOV LINUX/UNIX

Existuje niekoľko rôznych spôsobov, ako sledovať atribúty na vzdialenom linux/unix serveri. Jedným z nich je za pomoci SSH kľúčov (Secure Shell), vytvoreného SSL (Secure Sockets Layer) spojenia a zásuvného modulu Nagios *check_by_ssh,* spúšťať zásuvné moduly na vzdialenom serveri. Nevýhodou tejto metódy je extrémne zaťaženie na monitorovacom serveri, v prípade že chceme sledovať rádovo stovky služieb, alebo aj zničenie šifrovaného SSH spojenia [2].

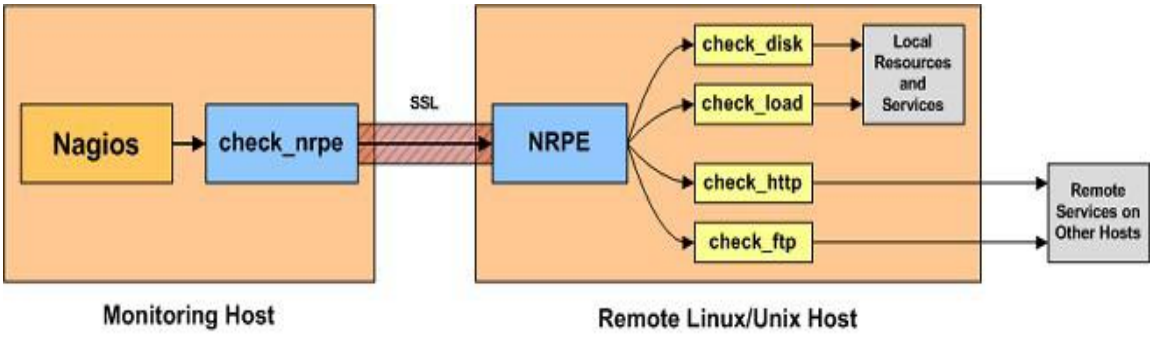
Druhá metóda je použitím doplnku NRPE (Nagios Remote Plugin Executor), ktorý umožňuje spúšťanie zásuvných modulov na monitorovanom vzdialenom serveri.

Ako je zrejmé z obrázka (*Obr. 16*), tak NRPE sa skladá z dvoch častí:

- zo zásuvného modulu *check_nrpe*, ktorý je umiestnený na serveri Nagios
- z démona NRPE, ktorý beží na vzdialenom sledovanom serveri

Komunikácia funguje nasledovným spôsobom:

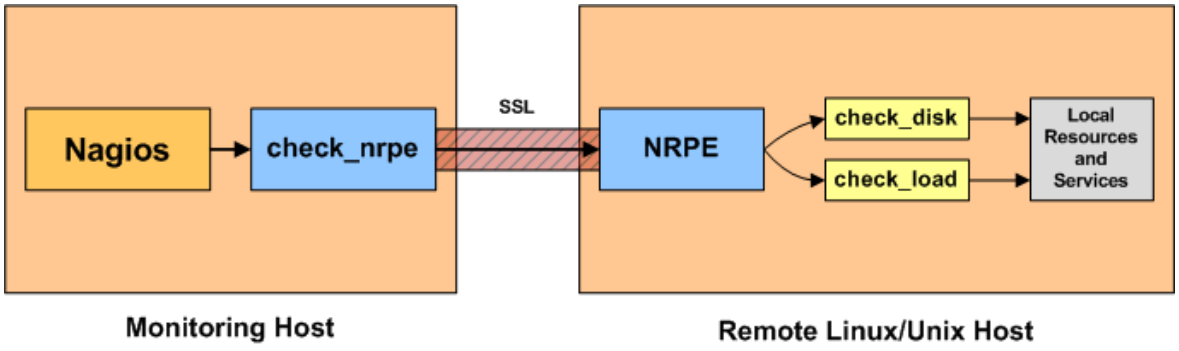
Nagios spustí zásuvný modul *check_nrpe* a zdelí mu, ktorú službu má skontrolovať. Zásuvný modul požiada NRPE démona o spustenie kontroly na vzdialenom serveri. Ak NRPE démon je nakonfigurovaný a oprávnený na spustenie lokálneho zásuvného modulu, tak v tom prípade ho vykoná a výsledok preposiela procesu Nagios.



Obr. 16. Blokové schéma kontroly Nagios a systémov Linux/Unix. [10]

5.2.1 Priama kontrola pomocou NRPE

Obrázok (*Obr. 17*) znázorňuje najčastejšie využitie NRPE démona. V tomto prípade sa sledujú iba lokálne zdroje na vzdialenom serveri, napr. vyťaženosť procesora, obsadenosť pamätí, alebo diskového poľa, swapu, počet prihlásených užívateľov atď.

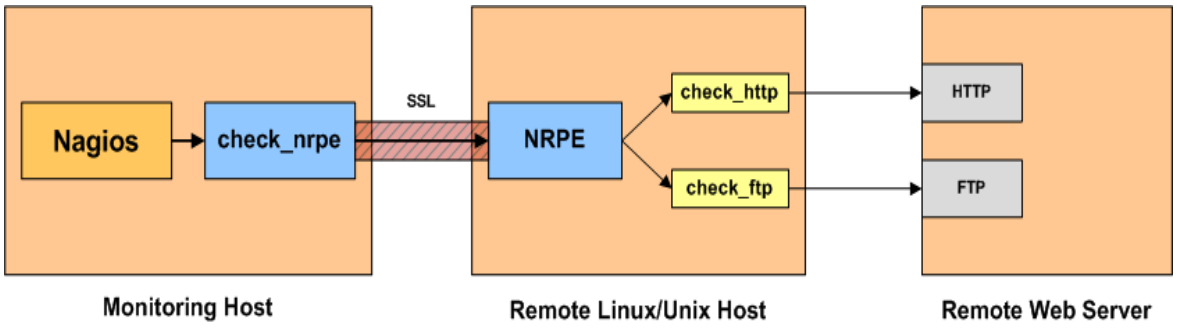


Obr. 17. Blokové schéma priamej kontroly NRPE. [10]

5.2.2 Nepriama kontrola pomocou NRPE

Obrázok (*Obr. 18*) znázorňuje, ako je možné využiť NRPE aj na kontrolu služieb a zdrojov na vzdialených serveroch, ktoré nie sú dosiahnuteľné z monitorovacieho stroja, z Nagiosu.

V tomto prípade, démon NRPE vystupuje v úlohe proxy servera.



Obr. 18. Blokové schéma nepriamej kontroly NRPE. [10]

5.2.3 Inštalácia NRPE

Inštalácia pozostáva z viacerých krokov, ale nie je vôbec zložitá. Je si ale potrebné uvedomiť, že na monitorovanom hostiteľovi sa inštaluje démon NRPE a na monitorovacom serveri sa inštaluje zásuvný modul NRPE.

5.2.3.1 Démon NRPE

Po stiahnutí a rozbalení inštalačného súboru sa treba postupne v nasledovnom poradí pospúšťať nasledovné príkazy.

http://prdownloads.sourceforge.net/sourceforge/nagios/nrpe-2.12.tar.gz

./configure[11]*, ./make all, ./make install-plugin, ./make install-daemon, ./make install-daemon-config, ./make install-xinetd*

Dôležitý je posledný príkaz, ktorý nainštaluje démona NRPE ako servis pod super démonom *xinetd*, ktorý má za úlohu spravovanie konektivity pomocou mechanizmu kontrolujúci prístupy. Sleduje, ktorý užívateľ má spustenú akú TCP službu a hlási to tomu, kto o to požiada. Častokrát býva služba xinted vypnutá, alebo pre istotu blokuje všetky prístupy [5].

Prístupy pre démona NRPE sa nastavujú parametrom *only_from = <IP adresa>* v konfiguračnom súbore */etc/xinetd.d/nrpe* a zapísaním portu, na ktorom bude NRPE démon počúvať, */etc/services* a vložiť riadok *nrpe 5666/tcp # NRPE.*

5.2.3.2 Zásuvný modul NRPE

Pri inštalácii zásuvných modulov sa taktiež použije predchádzajúci balík a spustia sa iba prvé tri kroky s príkazmi. Tým sa do adresára Nagios */usr/local/nagios/libexec* doinštaluje zásuvný modul *check_nrpe* [8].

Posledným krokom je už iba udeliť konfiguračným súborom Nagios syntax príkazu v *commands.cfg* a zadefinovať test služby v *services.cfg*.

```
define command {
    command_name        check_nrpe
    command_line        $USER1$/check_nrpe -H $HOSTADDRESS$ -c $ARG1$
            }
```

[11] Pri kompilácii treba mať doinštalovanú knižnicu , ináč sa vyskytne chyba: checking for SSL headers... configure: error: Cannot find ssl headers

```
define service {
        host_name                       localhost_UBUNTU_NRPE
        service_description             CPU Load
        use                             generic-service
        check_command                   check_nrpe!check_load
                }
```

5.3 MONITOROVANIE SYSTÉMOV WINDOWS

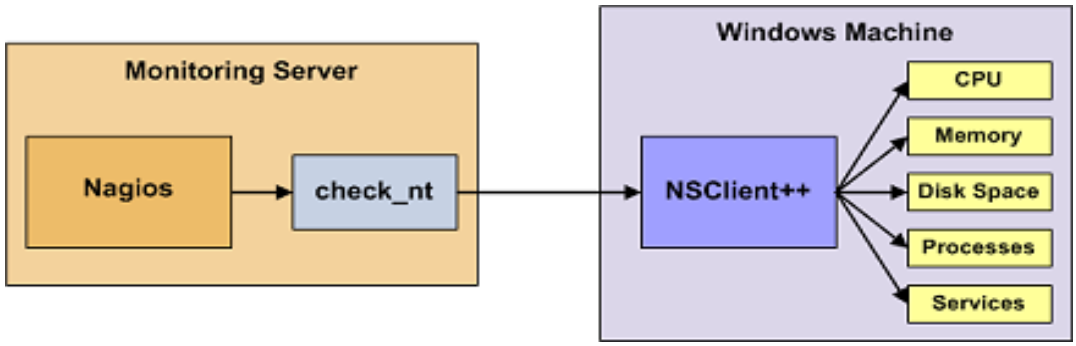
Monitorovanie bežiacich služieb na systémoch s operačným systémom Windows pomocou Nagios nie je takisto žiaden problém. Jednou z možností je využitie štandardnej funkcionality Windows a doinštalovať si podporu SNMP protokolu z inštalačného CD .

Druhá možnosť je použitie agenta NSCClient++.

5.3.1 Agent NSCClient++

Je to jednoduchý , bezpečný monitorovací agent, napísaný pre operačné systémy Windows. Tento agent slúži ako proxy medzi zásuvným modulom Nagios a monitorovanou službou , alebo atribútom na Windows serveri. Ak by nebol nainštalovaný, tak privátne služby, ako napríklad obsadenosť pamäte, disku, vyťaženosť procesora by nemohli byť monitorované.

V prípade monitorovania verejných služieb, ako napríklad HTTP (Hypertext Transfer Protocol), FTP (File Transfer Protocol), POP3 (Post Office Protocol) je možné využiť zásuvné moduly Nagios check_http, check_ftp, check_pop.



Obr. 19. Blokové schéma kontroly pomocou agenta NSClient++. [10]

Na obrázku (*Obr. 19*) je znázornený princíp zberu monitorovacích dát s využitím zásuvného modulu Nagios *check_nt* a agenta *NSClient++*. Pri dotazovaní o dáta, si zásuvný modul NSClient++ o ne požiada, uloží si ich do interného zásobníka a následne ich poskytne zásuvnému modulu check_nt na ďalšie spracovanie.

5.3.2 Inštalácia

Nevýhodou pri inštalácii je, že program nemá graficky prepracovanú inštaláciu a treba vykonávať manuálne inštalačné kroky.

- Z domovskej stránky stiahnuť agenta http://nsclient.org/nscp/downloads.
- Rozbaliť súbor do nového adresára C:\NSClient++.
- Spustiť príkazový riadok a z novovytvoreného adresára spustiť príkazy:

  NSClient++ /install

  NSClient++ SysTray install

- Štart/Stop sa vykoná nasledovné:

  NSClient++ /start

  NSClient++ /stop

5.3.3 Poinštalačné kroky

Po inštalácii je nutné otvoriť konfiguračný súbor NSC.INI a vykonať v ňom nasledujúce zmeny:

- Od komentovať všetky moduly, ktoré sa nachádzajú v sekcii [modules] s výnimkou CheckWMI.dll a RemoteConfiguration.dll.
- Ak je potrebné heslo pre klienta, tak v sekcii [Settings] je ho možné zadať.
- Odkomentovať možnosť „*allowed_hosts*“ v sekcii [Settings] a pridať IP adresu servera Nagios, alebo ponechať prázdne a budú sa môcť pripojiť všetcia hostitelia.
- Skontrolovať, či možnosť „*port*“ v sekcii [NSClient] je od komentovaná a nastavená na prednastavený port číslo *12489*.

5.4 POSIELANIE OZNÁMENÍ

Čo by to bolo za monitorovací nástroj, pokiaľ by pri detekovanej chybe neinformoval oprávnenú osobu? Len máloktorý správca počítačovej siete môže neustále pozerať na obrazovku a sledovať zmeny v monitorovacom nástroji. Preto samozrejme aj Nagios disponuje možnosťou posielania oznámení pomocou mailovej správy, SMS správy na mobilné zariadenie. Samozrejme, aby sa nestal z Nagiosu „spamový server“ , tak je nutné si dobre premyslieť, kedy budú oznámenia posielané, v akom množstve a koľkým adresátom. Nie vždy je totižto potrebné poslať oznámenie. Pre niekoho môže byť povaha poruchy, ktorú Nagios detekuje ako kritická a iný ju môže považovať za normálny stav.

5.4.1 Posielanie oznámení pomocou emailovej správy

Aby bolo možné doručiť oznámenie o poplachu z Nagios koncovému príjemcovi, tak je nutné využiť buď už funkčný poštový server, alebo si ho jednoducho na serveri nainštalovať. Jednoduchým riešením je použiť napríklad poštový server Postfix[12], ktorý je jednoduchý na inštaláciu a počiatočné nakonfigurovanie. Skladá sa z niekoľkých modulov, ktoré sa podieľajú na posielaní a prijímaní mailových správ. Je rýchly, jednoducho konfigurovateľný, bezpečný a kompatabilný [5].

mirino@ubuntu:~$ sudo apt-get install postfix

Príklad konfiguračného súboru *commands.cfg*:

define command {

command_name *notify-host-by-email*

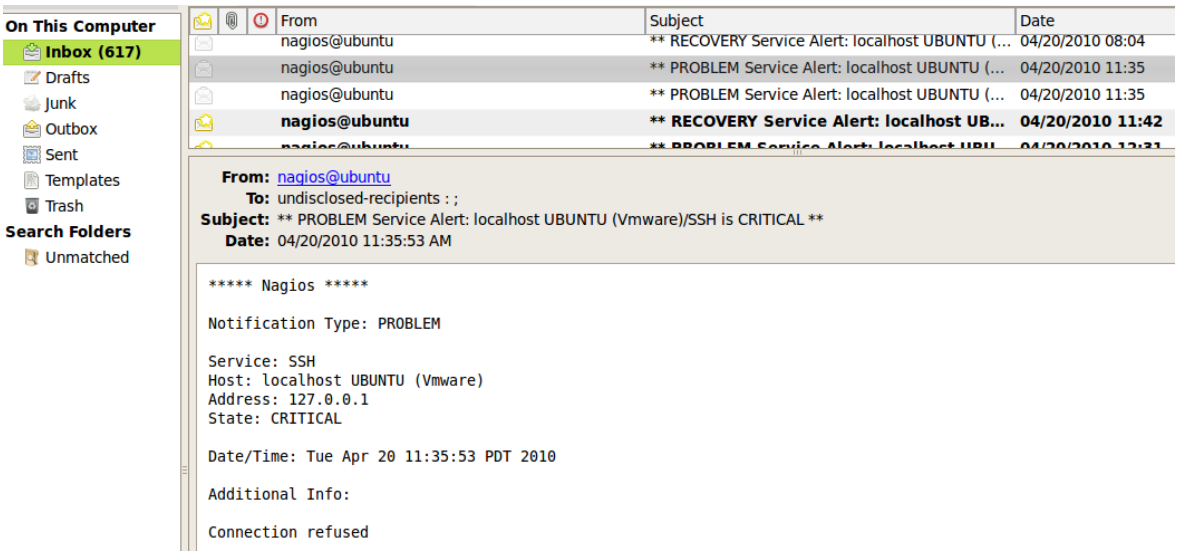
command_line */usr/bin/printf "%b" "Subject:** $NOTIFICATIONTYPE$ Host Alert: $HOSTNAME$ is $HOSTSTATE$ **\n***** Nagios *****\n\nNotification Type: $NOTIFICATIONTYPE$\nHost: $HOSTNAME$\nState: $HOSTSTATE$\nAddress: $HOSTADDRESS$\nInfo: $HOSTOUTPUT$\n\nDate/Time:*

[12] http://www.postfix.org/

$LONGDATETIME$\n" | /usr/sbin/sendmail $CONTACTEMAIL$

}

Na uvedenom príklade je možné vidieť, že sú volané dva príkazy. Príkazom *printf* sa vygeneruje predmet a telo emailovej správy a to s pomocou makier, ktoré začínajú so znakom $. Celý výstup je ďalej odovzdaný programu *sendmail*, ktorý mail odošle príjemcovi. Výsledok je možné vidieť na obrázku (*Obr. 20*), kde je mail doručený a zobrazený v poštovom klientovi.



Obr. 20. Výpis doručeného mailového oznámenia.

5.4.2 Posielanie oznámení pomocou SMS správy.

V súčasnej dobe už takmer všetcia vlastnia mobilný telefón a bola by škoda, keby sa nevyužil aj na posielanie oznámení z Nagiosu vo forme SMS správy. Aj keď jej dĺžka je ohraničená na maximálne 160 znakov, tak základné informácie o hostiteľovi, poruche, času a stavu zariadenia alebo služby sa do nej v pohode zmestia. Táto možnosť prináša obrovský prínos pre administrátorov, ktorý môžu byť o vzniknutom probléme informovaný kdekoľvek, aj keď nie sú priamo pri počítači.

Existuje viacero riešení posielania SMS správ z počítača na mobilný telefón. Na linuxových distribúciách je to napríklad pomocou dodatočným nainštalovaním obslužných

programov (Gnokii[13], Yaps[14]), ktoré komunikujú s mobilným telefónom, ktorý je pripojený k počítaču [11].

Ďalším riešením, je použitie SMS brán dostupných na internete. V tejto diplomovej práci bola využitá služba firmy Clickatell[15], ktorá poskytuje posielanie SMS správ cez HTTP/S API (Hypertext Transfer Protocol Secure / Application Programming Interface). Na tejto stránke je potrebné si vytvoriť prihlasovacie konto a povoliť komunikáciu HTTP/S API. Následne sa musia vykonať nastavenia v konfiguračnom súbore *command.cfg*, v ktorom sa zadefinuje príkaz, ktorý bude volaný pri poplachu v monitorovacom nástroji a posielaní oznámenia. Táto služba je však spoplatnená a k dispozícii je 10 testovacích SMS správ zdarma.

Niektorý mobilný operátori poskytujú službu, pomocou ktorej je možné poslať emailovú správu na ich emailové konto a tá je automaticky preposlaná na mobilný telefón.

Príklad konfiguračného súboru *commands.cfg*:

```
define command {
    command_name                    notify-host-by-sms
    command_line                    wget
"http://api.clickatell.com/http/sendmsg?user=PRIHLASOVACIE_MENO&password=HESLO&api_id=3232927&to=421911xxxyyy&text='$NOTIFICATIONTYPE$ Server is $HOSTSTATE$ ($HOSTOUTPUT$) @ $LONGDATETIME$'"
}
```

[13] http://www.gnokii.org/

[14] http://www.sta.to/ftp/yaps/

[15] http://www.clickatell.com/

6 VIZUALIZÁCIA

Zobrazovať informácie z dohľadového systému v textovej forme by asi nebola správna cesta, pretože človek si najviac zapamätá taký komplex informácii najlepšie vo forme obrázkov. Nagios je preto plne grafický a na zobrazovanie jeho výstupov sa používa webové rozhranie. Nemá síce špičkovo prepracovanú grafiku, ale za to sú informácie zobrazené veľmi prehľadne a ovládanie je intuitívne. Vďaka licenčnej politike Nagios túto slabosť vývojári po celom svete nenechali tak a prispôsobili rôzne vizualizačné produkty tretích strán na spoluprácu s Nagiosom.

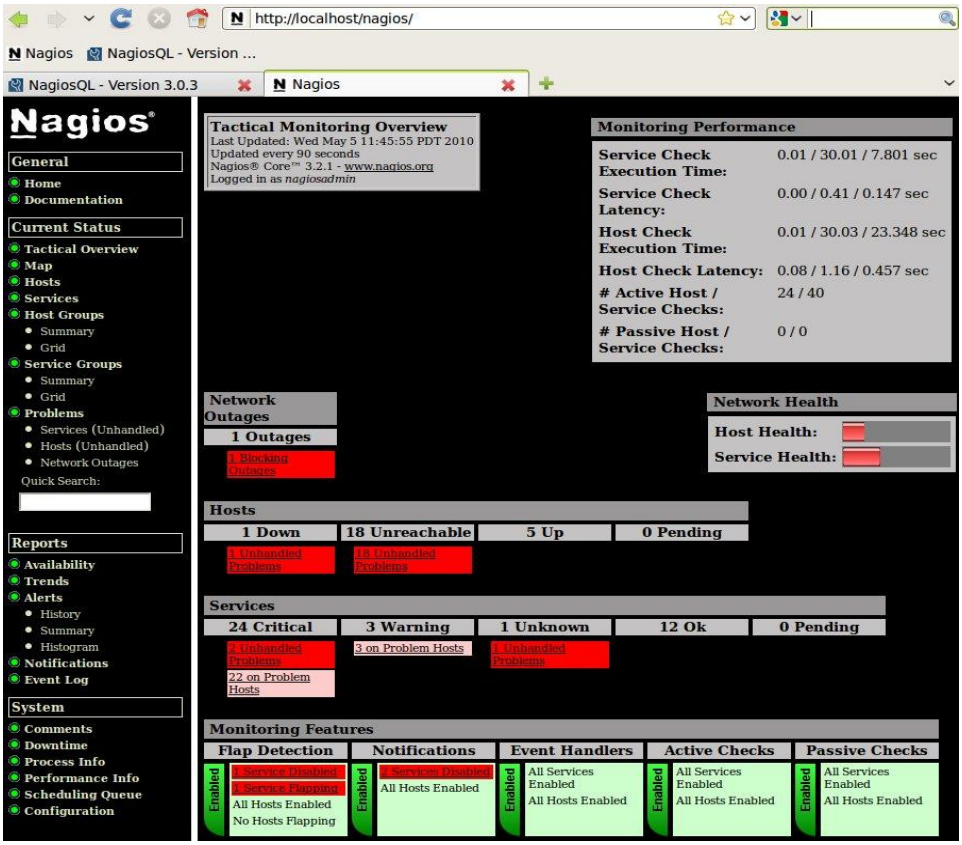
Predmetom tejto práce nie je detailné vysvetlenie užívateľskej práce s Nagios, takže pre názornú predstavu ako vlastne vyzerá webové rozhranie budú popísané iba základné obrazovky.

6.1 Vizualizácia pomocou štandardného webového rozhrania

6.1.1 Zobrazenie sumárneho prehľadu „Tactical monitoring overview“

Po otvorení stránky internetového prehliadača s odkazom na webové rozhranie Nagios, sa po úspešnom prihlásení otvorí úvodná obrazovka so sumárnym prehľadom o monitorovaných zariadeniach a službách. Ako je vidno na obrázku (*Obr. 21*), tak v strednej časti v sekcii „*Host*“ je zrejmé, že jedno zariadenie sa nachádza v stave *DOWN*, 18x*UNREACHABLE* (nedostupné), 5xUP (OK). Podobný stav je aj v sekcii „*Services*“.

Sekcie v spodnej časti zobrazujú informácie o tom, či je globálne povolené oznamovanie o poplachoch a iných povolených testovacích prvkoch.



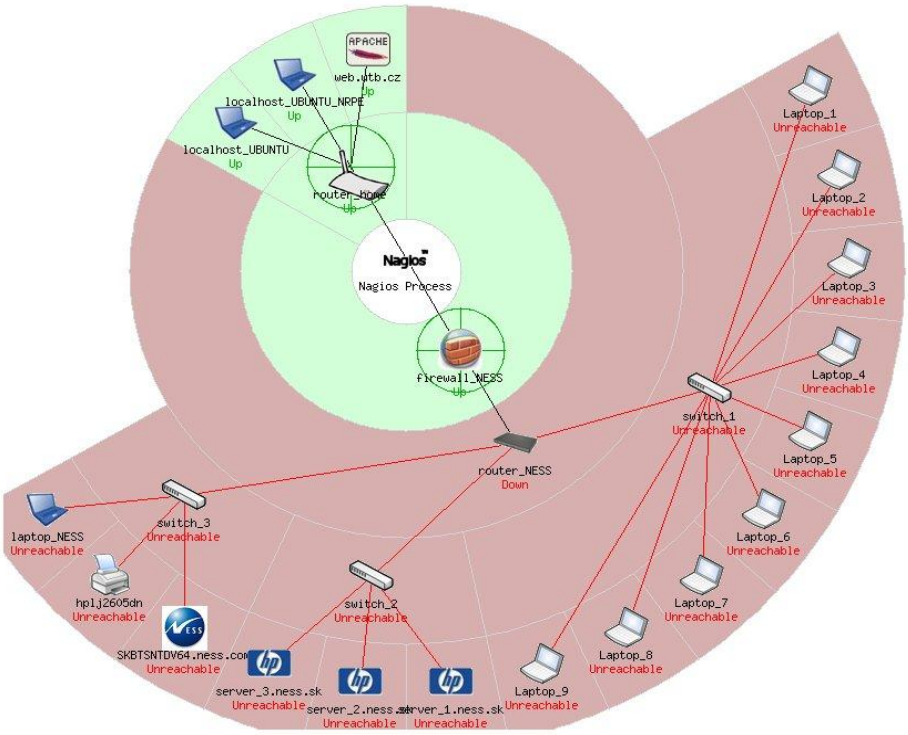
Obr. 21. Sumárny prehľad o stave monitorovaných zariadeniach a službách.

6.1.2 Zobrazenie topologickej mapy monitorovanej siete „Map“

Východiskovým bodom v topologickom vyobrazení siete je „Nagios Process“, tak ako je na obrázku (*Obr. 22*) a od neho sú pripojený hostitelia čiarami buď priamo, alebo ak hostiteľ má zadefinovaný v konfiguračnom súbore hosts.cfg parameter *parents*, tak sa môžu vytvoriť závislosti medzi hostiteľmi. V ukážke je možné pozorovať, ako v prípade keď router „router_NESS“ je v stave DOWN, tak aj ostatná celá podsieť hostiteľov je nedostupná a zobrazená červenou farbou.

Po nainštalovaní Nagios je táto mapa bez pekných ikoniek a namiesto nich sú iba symboly otáznika, ale na internete sa dajú stiahnuť súbory ikon s rôznymi vyobrazeniami.[16] Samozrejme, že sa dajú vytvoriť aj vlastné, ale maximálna veľkosť je ohraničená na 40x40 pixelov.

[16] http://www.intec.uni.cc/html/projects/fnagios.html



Obr. 22. Grafické zobrazenie závislostí monitorovaných zariadení.[17]

6.1.3 Zobrazenie detailného prehľadu o hostiteľoch „Hosts“

Po kliknutí v menu na položku „*Host*“ bude vypísaný detailný zoznam viď. (*Obr. 23*) s informáciami o zariadeniach, ktorý je rozdelený do niekoľkých stĺpcoch, ktoré obsahujú názov zariadenia, jeho aktuálny stav, posledný čas testovania, dobu celkového monitorovania a posledný stĺpec zobrazuje hodnoty meranej veličiny. Po kliknutí na odkaz zariadenia sa dá dostať o úroveň hlbšie, kde sa naskytajú ďalšie možnosti spojené s týmto zariadením a to napr. pozastavenie zasielania oznámení, preplánovanie opätovného testovania atď.

[17] Všetky názvy hostiteľov a zariadení v tejto diplomovej práci sú kvôli bezpečnosti vymyslené a zámerne zmenené a vo firme NESS SK a.s. sa nevyskytujú.

firewall_NESS	UP	05-05-2010 13:35:20	0d 5h 37m 28s	PING OK - Packet loss = 0%, RTA = 21.38 ms
hplj2605dn	UNREACHABLE	05-05-2010 13:35:20	0d 2h 19m 10s	(Host Check Timed Out)
laptop_NESS	UNREACHABLE	05-05-2010 13:34:00	0d 2h 20m 40s	(Host Check Timed Out)
localhost_UBUNTU	UP	05-05-2010 13:36:38	18d 13h 46m 25s	PING OK - Packet loss = 0%, RTA = 0.35 ms
localhost_UBUNTU_NRPE	UP	05-05-2010 13:36:58	18d 13h 45m 53s	PING OK - Packet loss = 0%, RTA = 0.18 ms
router_NESS	DOWN	05-05-2010 13:38:28	0d 2h 49m 38s	(Host Check Timed Out)
router_home	UP	05-05-2010 13:37:48	0d 7h 22m 56s	PING OK - Packet loss = 0%, RTA = 6.63 ms
server_1.ness.sk	UNREACHABLE	05-05-2010 13:38:58	0d 2h 14m 31s	(Return code of 127 is out of bounds - plugin may be missing)

Obr. 23. Zobrazenie detailného prehľadu o hostiteľoch.

6.1.4 Zobrazenie detailného prehľadu o službách „Services“

Ďalším odkazom v menu je „*Services*“, ktorý je podobný odkazu „Host“, s tým rozdielom, že je viacej zmeranejší na služby testované na zariadení.

laptop_NESS	Explorer	CRITICAL	05-05-2010 13:17:10	0d 2h 42m 35s	1/3	CRITICAL - Socket timeout after 10 seconds
localhost_UBUNTU	Current Load	OK	05-05-2010 13:15:21	6d 6h 57m 24s	1/4	OK - load average: 0.18, 0.08, 0.03
	Current Users	OK	05-05-2010 13:15:31	18d 13h 22m 19s	1/4	USERS OK - 1 users currently logged in
	HTTP	OK	05-05-2010 13:15:31	7d 6h 22m 2s	1/4	HTTP OK: HTTP/1.1 200 OK - 453 bytes in 0.004 second response time
	PING	OK	05-05-2010 13:17:29	18d 13h 24m 26s	1/4	PING OK - Packet loss = 0%, RTA = 0.06 ms
	Root Partition	OK	05-05-2010 13:15:41	18d 13h 22m 3s	1/4	DISK OK - free space: / 15022 MB (82% inode=86%):
	SSH	OK	05-05-2010 13:19:39	14d 4h 51m 5s	1/2	SSH OK - OpenSSH_5.1p1 Debian-6ubuntu2 (protocol 2.0)
	Swap Usage	OK	05-05-2010 13:19:03	18d 13h 24m 10s	1/4	SWAP OK - 100% free (894 MB out of 894 MB)
	Total Processes	OK	05-05-2010 13:17:48	18d 13h 21m 47s	1/4	PROCS OK: 88 processes with STATE = RSZDT
localhost_UBUNTU_NRPE	CPU Load	OK	05-05-2010 13:10:59	18d 13h 26m 17s	1/3	OK - load average: 0.00, 0.02, 0.00
router_NESS	Kontrola SMTP	CRITICAL	05-05-2010 13:18:17	0d 6h 37m 3s	1/3	CRITICAL - Socket timeout after 10 seconds
router_home	PING	OK	05-05-2010 13:16:28	0d 7h 3m 20s	1/3	PING OK - Packet loss = 0%, RTA = 2.89 ms
	Port 1 Bandwidth Usage	UNKNOWN	05-05-2010 13:18:26	0d 7h 1m 7s	3/3	check_mrtgtraf: Unable to open MRTG log file

Obr. 24. Zobrazenie detailného prehľadu o službách.

6.2 Vizualizácia pomocou nástroja tretích strán

6.2.1 Zber výkonnostných dát

Bola by asi veľká škoda, keby informácie a údaje, ktoré sú získavané pri monitorovaní siete a zariadení by nebolo možné ďalej spracovať. Tieto výkonnostné dáta sú vyparsované a ukladajú sa do súboru, odkiaľ môžu byť následne spracované iným externým programom, alebo sa výkonnostné dáta ukladajú priamo do tohto programu po každej kontrole služby alebo hostiteľa.

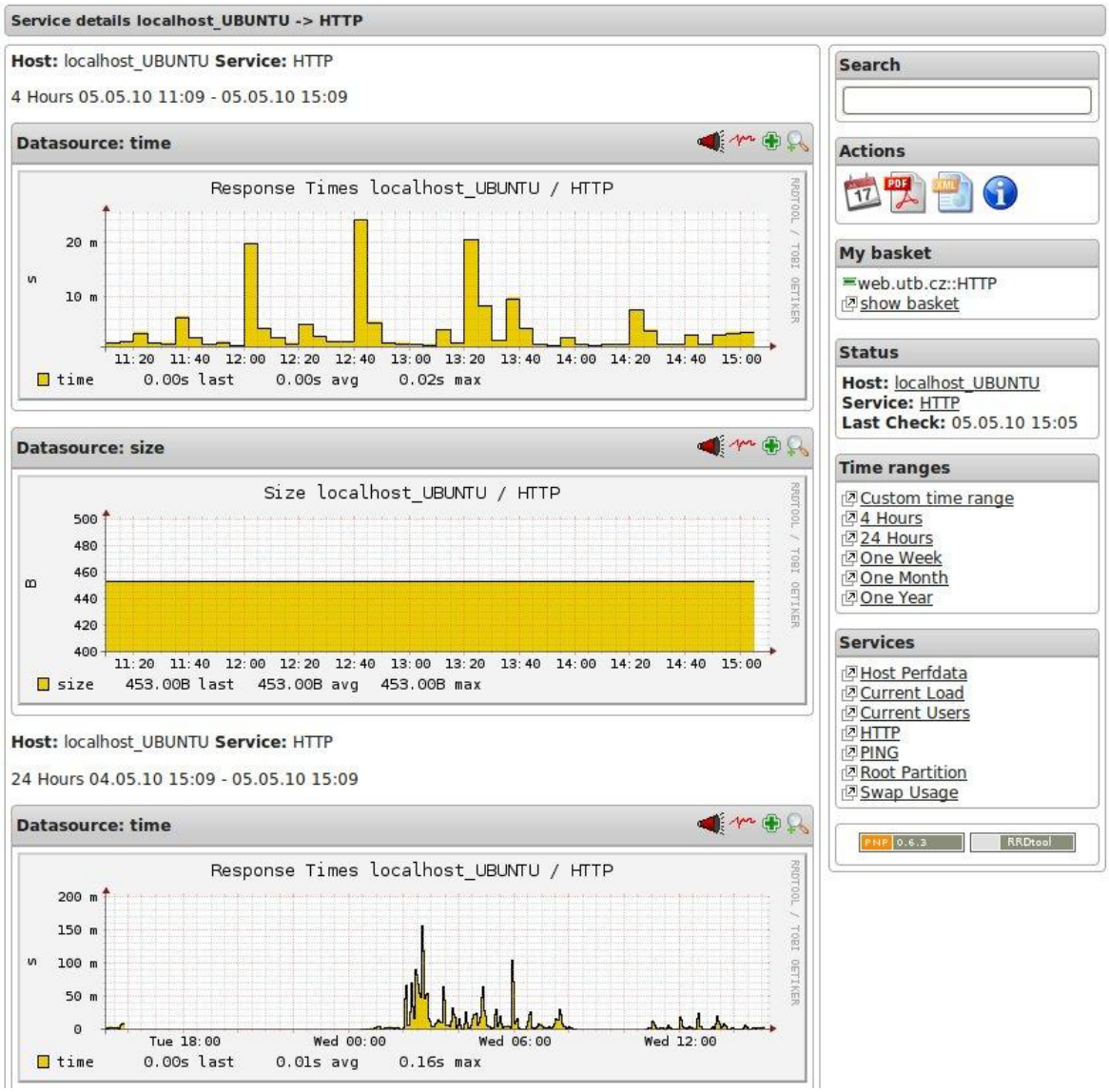
Nagios má dva typy zbieraných výkonnostných dát:

1. Interné výkonnostné dáta: obsahujú štatistiky o výkonnostných časoch jednotlivých testoch a rozdieloch medzi časom aktuálneho testu a plánovaného testu.
2. Výkonnostné dáta s výstupov zásuvných modulov: obsahujú napr. doby odozvy, obsadenosť partícii atď.

6.2.2 Zobrazenie grafov s výkonnostnými štatistikami

Aby bolo možné zo zozbieraných výkonnostných dát generovať grafy so štatistickými údajmi, tak je potrebné doinštalovať externý softvér, ktorý sa o všetko postará. Na tieto účely sa výborne hodí program *PNP4NAGIOS*[18]. Je napísaný v jazykoch Perl, PHP,C. Dáta zo zásuvných modulov sú ním automaticky analyzované a ukladané do RRD databázy (Round Robin Databases) [19]. Všetko sa deje pomocou perl skriptu *process_perfdata.pl,* ktorý zvládne spracovať dáta až z 2000 služieb a to každých 5 minút. Po nainštalovaní tohto doplnku pre Nagios je v jeho štandardnom web rozhraní možnosť zobrazenia výstupov ako je na obrázku (*Obr. 25*). Grafy pre konkrétnu monitorovanú veličinu sú zobrazené v piatich vyobrazeniach po odlišných časových úsekoch a to 4hodiny, 1 deň, 1 mesiac, 1 týždeň, 1 rok.

[18] http://docs.pnp4nagios.org/start



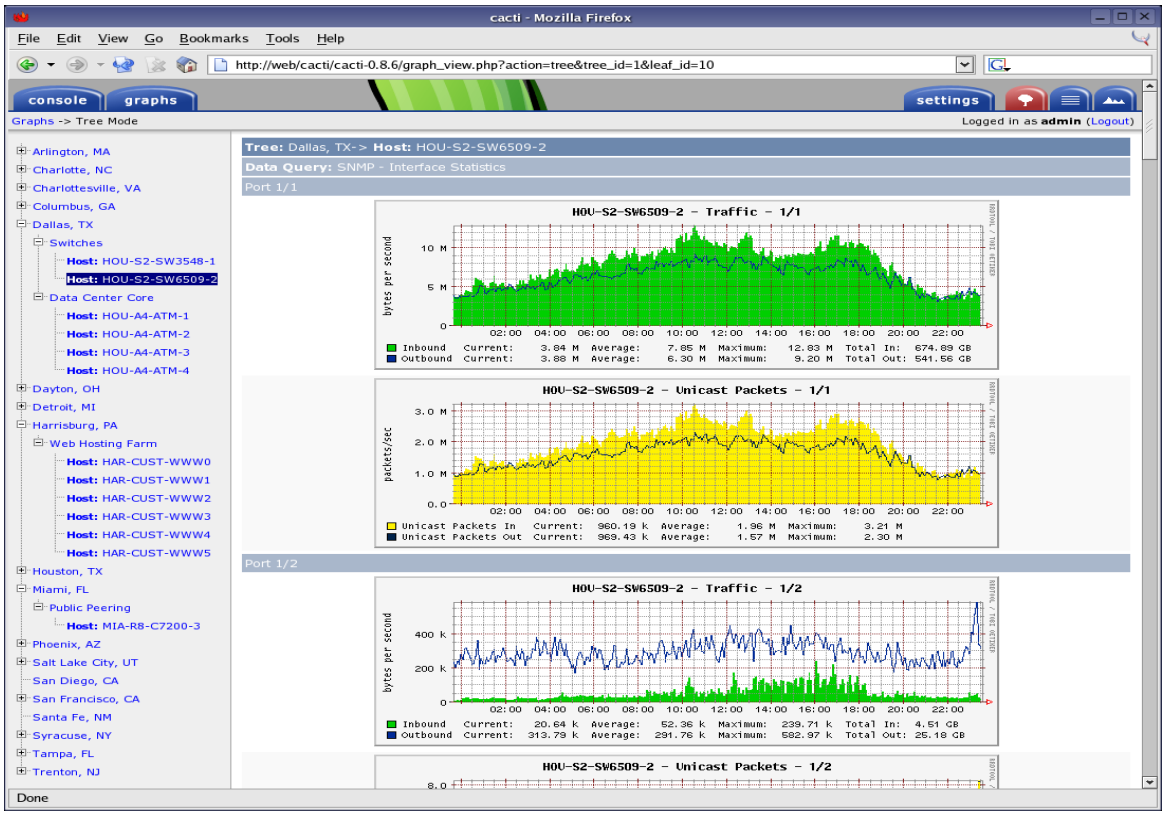
Obr. 25. Zobrazenie výkonnostných štatistík.

7 POROVNANIE S EXISTUJÚCIM RIEŠENÍM.

Toto porovnanie sa nebude zaoberať s komerčnými riešeniami, ktoré sú v takej miere prepracované, že porovnanie s Open Source produktmi by bolo neobjektívne a poukazovalo by v jasný neprospech v Open Source riešeniam. Jediným potešením by bolo, že Open Source riešenia sú zadarmo a komerčné riešenia napr. HP Open View[19], CISCO Works[20] sa pohybujú vo vyšších cenových hladinách.

7.1 Cacti

Cacti je takisto monitorovací nástroj, ktorý je na rozdiel od Nagiosu napísaný v jazyku PHP. Je ho možné ovládať pomocou webového rozhrania s napojením na databázu MySQL, v ktorej dokáže uchovávať namerané dáta, z ktorých pomocou RRDtool dokáže generovať krásne grafy, tak ako je vyobrazené na obrázku (*Obr. 26*) [3].



Obr. 26. Webové rozhranie programu Cacti. [19]

[19] http://www.managementsoftware.hp.com

[20] http://www.cisco.com/en/US/products/sw/cscowork/ps2425/

V nasledujúcich tabuľkách (*Tab. 6*), (*Tab. 7*) bude popísaná analýza porovnaní medzi Nagios a Cacti a stručný prehľad ich výhod a nevýhod.

Tab. 6. Porovnanie monitorovacích systémov.

	Nagios	**Cacti**
Funkcia		
monitorovanie systémov	áno	áno
monitorovanie sieťových služieb	áno	so skriptom
podpora snmp	so zásuvným modulom	áno
nutný vlastný monitorovací klient	áno	nie
posielanie správ na email, pager, mobil	áno	nie
posielanie štatistík	nie	so skriptom
Grafické rozhranie		
grafy	so zásuvným modulom	áno
zoom grafu	so zásuvným modulom	áno
grafický výstup	voliteľné	áno
autorizácia	áno	áno
odhlásenie užívateľa	nie	áno
Operačný systém		
Linux	áno	áno
Unix	áno	áno
Windows	nie	áno
Inštalácia a konfigurácia		
Debian (apt-get)	áno	áno
konfigurácia - s grafickým rozhraním	iba s nástrojom tretej strany	áno
konfigurácia - s konfiguračnými súbormi	áno	nie
Požiadavky		
Apache	áno	áno
PHP	nie	áno
MySQL	nie	áno
RRDTool	iba s nástrojom tretej strany	áno

Tab. 7. Výhody a nevýhody Nagios a Cacti.

	Nagios	**Cacti**
Výhody	Open Source	Open Source
	Veľká podpora vývojárskej komunity	Jednoduché používanie pomocou webového rozhrania
	Jednoduché používanie pomocou webového rozhrania	Integrovaná podpora vykresľovania grafov
	Schopnosť definície hierarchie monitorovaných zariadení a závislostí medzi nimi	Grafy môžu byť zväčšované pomocou Javascriptu

	Oznamovanie poruchy mailom alebo SMS	Správa užívateľov s rôznymi úrovňami oprávnení
Nevýhody	Slabšia podpora virtualizácie nameraných hodnôt, pretože chýba podpora vykresľovania grafov	Zložitejšia inštalácia
	Zložitejšia konfigurácia s nutnosťou predom zadefinovania monitorovaných zariadení	Obmedzená možnosť použitia interaktívnych dynamických grafov
	Mnoho zásuvných modulov nemá zadefinované príklady nastavení, takže je nutné zisťovať, ako vlastne pracujú	Frekvencia kontrol je prednastavená na 5min a jej manuálne zväčšenie má za následok stratu doposiaľ nameraných hodnôt

ZÁVER

Praktickým výsledkom tejto diplomovej práce je zavedenie dohľadového systému Nagios vo firme NESS Slovakia a.s., ktorý bude určite veľkým prínosom, keďže spoločnosť doposiaľ disponuje iba komerčným monitorovacím nástrojom, ktorého licenčná politika si vyžaduje nazrieť hlbšie do vrecka. Jediná počiatočná investícia doposiaľ popísaného voľne dostupného produktu Nagios je jeho inštalácia a konfigurácia. Čas strávený nad jeho spojazdnením sa ale rýchlo vráti vo forme kvalitného systému na monitorovanie počítačovej siete z jedného centrálneho bodu.

Ďalším prínosom je detailnejšie popísanie konfigurácie a inštalácie, pretože väčšina problémov pri monitorovacích systémoch sa netýka len chybného návrhu komunikácie medzi ním a monitorovanými zariadeniami, ale chyby často vznikajú už pri implementácii monitorovacieho systému, čo v konečnom dôsledku nemusí mať iba finančné následky.

Významným prínosom je nepochybne aj prepojenie Nagios s interným monitorovacím nástrojom CCMS v SAP systémoch, oznamovanie udalostí emailom a pomocou SMS správ a prepojenie Nagios s programom slúžiacim na generovanie štatistických grafov z nameraných hodnôt, čo bolo vo firme brané s pozitívnymi ohlasmi.

Výsledkom diplomovej práce je reálne fungujúci dohľadový systém. Práca poskytne dostatočný teoretický základ a praktický prínos pri implementácii a bude osožná začínajúcim administrátorom, ktorí sa pri nasadzovaní systému zbytočne nedostanú do stavu, kedy sa nebudú vedieť pohnúť vo svoje práci ďalej.

CONCLUSION

Practical outcome of this thesis is installation of system Nagios in Ness Slovakia a.s., that will certainly be a big asset, as Ness company is having only commercial monitoring tool at the moment and its licence politics is quite expensive. The only starting invetition of so far mentioned free reachable product Nagios is its installation and configuration. The time spent with its start-up will be quickly paid back to the company as high quality monitoring system of computer network from one central point.

The other asset of this system is more detailed description of configuration and installation, because main part of problems with monitoring systems are not connected only to incorrect proposal of communication between the system and monitoring devices, but mistakes often arise at implementation of monitoring system what should have not only financial consequences. One of significant assets is also switch of Nagios to internal monitoring tool CCMS at SAP systems, notification of events by email and SMS and switch of Nagios with program that is generating statistical diagrams with measured values, what met with really positive feedback within the company.

The result of this thesis is in reality working system. The thesis will provide with enough of theoretical information and practical asset at implementation. The thesis will be useful for administrators – beginners, who won't come to situation where they will not be able to move any further at their work while starting-up the system.

ZOZNAM POUŽITEJ LITERATÚRY

[3] DAVID, Josephsen. *Building A Monitoring Infrastructure With NAGIOS*. 1st edition. United States, Boston: Pearson Education, Inc., 2007. 230 s. ISBN 0-132-23693-1

[2] BARTH, Wolfgang. *Nagios, System and Network Monitoring.* 1st edition. Germany, Munich: Open Source Press GmbH, 2006. 462 s. ISBN 3-937514-09-0

[3] SCHUBERT, Max, et al. *Nagios 3 Enterprise Network Monitoring Including Plug-Ins and Hardware Devices.* 1st edition. United States, Burlington: Syngress Publishing, Inc., 2008. 384 s. ISBN 13: 978-1-59749-2

[4] DOSTÁLEK, Libor, KABELOVÁ, Alena. *Velký průvodce protokoly TCP/IP a systémem DNS.* 2. vydání. Česká republika, Praha: Computer Press, 2000. 426 s. ISBN 80-7226-323-4

[5] Kolektiv autorů. *Linux-Dokumentační projekt.* 4. vydání. Česká republika, Praha: Computer Press, 2008. 1336 s. ISBN 978-80-251-1525-1

[7] KRETCHMAR, James. *Open Source Network Administration.* 1st edition. United States, Upper Saddle River: Prentice Hall Professional, 2003. 238 s. ISBN 13: 9780130462107

[8] BAUER, Kirk, CAMPI, Nate. *Automating UNIX and Linux System Administration.* 2st edition. United States, New York: APress, Inc. 2003. 549 s. ISBN 1-59059-212-3

[9] Kolektiv autorů. *Net-SNMP* [online]. 2007, 2007-02-03 [cit. 2010-01-28]. Dostupný z WWW: http://www.net-snmp.org/docs/readmefiles.html

[10] GALSTAD, Ethan. *Official Nagios Documentation* [online]. 2010, 2009-06-16 [cit. 2010-01-28]. Dostupný z WWW: http://support.nagios.com/knowledgebase/officialdocs

[11] BURGESS, Chris. *The Nagios Book* [online]. 2005, 2005-01-05 [cit. 2010-02-02]. Dostupný z WWW: http://www.nagiosbook.org/html/index.html

[12] KLAŠKA, Ľuboš. *Vznik a principy SNMP* [online]. 2000, 2000-06-11 [cit. 2010-04-08]. Dostupný z WWW: http://www.svetsiti.cz/view.asp?rubrika=Tutorialy&temaID=23&clanekID=29

[13] KLAŠKA, Ľuboš. *Model Manager-Agent* [online]. 2000, 2000-06-12 [cit. 2010-04-08]. Dostupný z WWW: http://www.svetsiti.cz/view.asp?rubrika=Tutorialy &temaID=23&clanekID=30

[14] KLAŠKA, Ľuboš. *SNMP objekty a MIB* [online]. 2000, 2000-06-13 [cit. 2010-04-08]. Dostupný z WWW: http://www.svetsiti.cz/view.asp?rubrika=Tutorialy &temaID=23&clanekID=31

[15] KLAŠKA, Ľuboš. *Formát SNMP správ* [online]. 2000, 2000-06-14 [cit. 2010-04-08]. Dostupný z WWW: http://www.svetsiti.cz/view.asp?rubrika=Tutorialy &temaID=23&clanekID=32

[16] UNGVARSKÝ Imrich. *Nástroj pre analýzu topológie siete.* Slovenská republika, Košice, 2006. 68 s. Diplomová práca na Technickej univerzite na fakulte elektrotechniky a informatiky. Vedúci diplomovej práce Ing. František Jakab, PhD.

[17] HOMANN, Rouven. *Documentation for NagiosQL 3.x* [online]. 2009, 2009-09-11 [cit. 2010-04-21]. Dostupný z WWW: http://www.nagiosql.org/faq/31-general-documentation.html

[18] GALSTAD, Ethan. *Nagios Voted Monitoring Application Of The Year* [online]. 2010, 2010-02-25 [cit. 2010-04-23]. Dostupný z WWW: http://www.nagios.org/news/77-news-announcements/236-nagios-voted-monitoring-application-of-the-year

[19] The Cacti Group. *Cacti The Complet RRDTool based Graphing Solution* [online]. 2010, 2009-01-20 [cit. 2010-05-03]. Dostupný z WWW: http://www.cacti.net/image.php?image_size=1095x972&image_id=43

[20] IBM Corporation. *Systém pomoci pre IBM Eclipse* [online]. 2005, 2005-11-02 [cit. 2010-04-28]. Dostupný z WWW: http://publib.boulder.ibm.com/infocenter/idshelp/v10/index.jsp?topic=/com.ibm.snmp.doc/snmp34.htm

ZOZNAM POUŽITÝCH SYMBOLOV A SKRATEK

API	Application Programming Interface
BIND	Berkeley Internet Name Domain
CCMS	Computing Center Management System
CGI	Common Gateway Interface
FTP	File Transfer Protocol
GNU	GNU's Not Unix
GPL	General Public Licence
HTTP	Hypertext Transfer Protocol
HTTP/S	Hypertext Transfer Protocol Secure
ICMP	Internet Control Message Protocol
MAC	Media Access Control
MySQL	My Structured Query Language
ISO	International Organization for Standardization
LAN	Local Area Network
NMS	Network Management Software
NRPE	Nagios Remote Plugin Executor
OID	Object Identifier
OpenSSL	Secure Sockets Layer
OSI	Open Systems Interconnection
PERL	Practical Extraction and Report Language
PFCG	Profile Generator
PHP	Hypertext Preprocessor
POP3	Post Office Protocol
RFC	Remote Function Call

SAP	Systemanalyse und Programmentwicklung, System Analysis and Program Development
SDK	Software Development Kit
SMS	Short Message Service
SMI	Structure of Management Information
SNMP	Simple Network Management Protocol
SMTP	Simple Mail Transport Protocol
SSH	Secure Shell
SSL	Secure Sockets Layer
STDOUT	Standard output
TTL	Time To Live
TCP/IP	Transmission Control Protocol/Internet Protocol
UDP	User Datagram Protocol
VPN	Virtual Private Network
WAN	Wide Area Network

ZOZNAM OBRÁZKOV

ZOZNAM TABULIEK

ZOZNAM PRÍLOH

PRÍLOHA P I: KONFIGURAČNÉ SÚBORY PRE NAGIOS.

Konfiguračné súbory sú uložené na CD podľa nižšie uvedenej adresárovej štruktúry.

..\nagios

commands.cfg (definícia spustiteľných príkazov)

contactgroups.cfg (definícia kontaktných skupín)

contacts.cfg (definovanie kontaktov, ktorým bude odoslané oznámenie)

contacttemplates.cfg (definovanie šablony pre kontakty)

hostdependencies.cfg (definovanie závislostí medzi hostiteľmi)

hostescalations.cfg (definovanie poradia a priority oznámení pri nedostupnosti hostiteľa)

hostextinfo.cfg (definovanie doplnkových informácii o hostiteľovi, napr. použitá ikona v „STATUS MAP“)

hostgroups.cfg (definovanie skupín pre hostiteľov)

hosttemplates.cfg (definovanie šablony pre hostiteľov)

servicedependencies.cfg (definovanie závislostí medzi službami)

serviceescalations.cfg (definovanie poradia a priority oznámení pri nedostupnosti služby)

serviceextinfo.cfg (definovanie doplnkových informácii o službe, napr. použitá ikona v „STATUS MAP“)

servicegroups.cfg (definovanie skupín pre služby)

servicetemplates.cfg (definovanie šablony pre služby)

timeperiods.cfg (definovanie času, kedy budú posielané oznámenia)

..\nagios\hosts (definovanie monitorovaných zariadení a hostiteľov)

firewall_NESS.cfg

hplj2605dn.cfg

Laptop_1.cfg

Laptop_2.cfg

Laptop_3.cfg

Laptop_4.cfg

Laptop_5.cfg

Laptop_6.cfg

Laptop_7.cfg

Laptop_8.cfg

Laptop_9.cfg

laptop_NESS.cfg

localhost_UBUNTU.cfg

localhost_UBUNTU_NRPE

router_home.cfg

router_NESS.cfg

server_1.ness.sk.cfg

server_2.ness.sk.cfg

server_3.ness.sk.cfg

SKBTSNTDV64.ness.com.

switch_1.cfg

switch_2.cfg

switch_3.cfg

web.utb.cz.cfg

..**nagios\services** (definovanie monitorovaných služieb bežiacich na hostiteľoch)

check_sap_as.cfg

localhost.cfg

localhost_NRPE.cfg

printer.cfg

SAP.cfg

switch.cfg

windows.cfg