UNIVERZITA PARDUBICE
Fakulta elektrotechniky a informatiky

Dohledové systémy pro počítačové sítě

Vít Pleskot

Bakalářská práce
2012

Univerzita Pardubice
Fakulta elektrotechniky a informatiky
Akademický rok: **2011/2012**

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

(PROJEKTU, UMĚLECKÉHO DÍLA, UMĚLECKÉHO VÝKONU)

Jméno a příjmení:	**Vít Pleskot**
Osobní číslo:	**I09231**
Studijní program:	**B2646 Informační technologie**
Studijní obor:	**Informační technologie**
Název tématu:	**Dohledové systémy pro počítačové sítě**
Zadávající katedra:	**Katedra informačních technologií**

Zásady pro vypracování:

Cílem práce je provést analýzu možností monitoringu a managementu počítačových sítí. Na základě výsledků provedené analýzy pro konkrétní firemní prostředí bude vybrán a prakticky nasazen konkrétní software pro monitoring sítí. Autor práce podrobně představí principy, na kterých pracují softwary pro monitoring a management počítačových sítí, navrhne hodnotící kriteria s ohledem na konkrétní požadavky firmy středního rozsahu. Práce bude obsahovat popis konkrétního nasazení vybraného monitorovacího systému a vyhodnocení jeho reálného chodu po dobu minimálně 2 měsíců.

Rozsah grafických prací:

Rozsah pracovní zprávy:

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

Seznam odborné literatury:

WOLFGANG, Barth. Nagios : System and Network Monitoring. Daly City, California/Un : No Starch Press,US, 2008. 720 s. ISBN 9781593271794.
ERICSSON, Andreas, et al. Nagios 3 Enterprise Network Monitoring : Including Plug-Ins and Hardware Devices. Rockland, MA/US : Syngress Media,U.S., 2008. 376 s. ISBN 9781597492676.
BADGER, Michael . Zenoss Core Network and System Monitoring. Birmingham/US : Packt Publishing Limited, 2008. 280 s. ISBN 9781847194282.
DONDICH, Taylor. Network Monitoring with Nagios . [s.l.] : O'Reilly Media, 2006. 56 s. ISBN 978-0-596-55905-2.
OLUPS, Rihards. Zabbix 1.8 Network Montioring. Birmingham/US : Packt Publishing Limited, 2010. 428 s. ISBN 9781847197689.

Vedoucí bakalářské práce: **Mgr. Josef Horálek**
Katedra softwarových technologií

Datum zadání bakalářské práce: **16. prosince 2011**
Termín odevzdání bakalářské práce: **11. května 2012**

Univerzita Pardubice Česká republika

L.S.

prof. Ing. Simeon Karamazov, Dr.
děkan

Ing. Lukáš Čegan, Ph.D.
vedoucí katedry

V Pardubicích dne 30. března 2012

Prohlášení autora

Prohlašuji, že jsem tuto práci vypracoval samostatně. Veškeré literární prameny a informace, které jsem v práci využil, jsou uvedeny v seznamu použité literatury.

Byl jsem seznámen s tím, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorský zákon, zejména se skutečností, že Univerzita Pardubice má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona, a s tím, že pokud dojde k užití této práce mnou nebo bude poskytnuta licence o užití jinému subjektu, je Univerzita Pardubice oprávněna ode mne požadovat přiměřený příspěvek na úhradu nákladů, které na vytvoření díla vynaložila, a to podle okolností až do jejich skutečné výše.

Souhlasím s prezenčním zpřístupněním své práce v Univerzitní knihovně.

V Pardubicích dne 8. 5. 2012 Vít Pleskot

Poděkování

Rád bych poděkoval Mgr. Josefu Horálkovi za vedení mé bakalářské práce a cenné rady. Dále bych chtěl poděkovat vedení firmy Arit s.r.o., výslovně pak Ing. Tomáši Hruškovi, za možnost uskutečnit praktickou část práce v síti firmy. Závěrem bych rád poděkoval celé své rodině a přátelům za psychickou podporu v těžkých chvílích a neustálé popohánění vpřed. Děkuji!

Anotace

Tato bakalářská práce analyzuje možnosti monitoringu a managementu počítačových sítí. V práci jsou představeny principy monitoringu i open source monitorovací systémy. Výsledkem teoretické části je zvolení vhodného monitorovacího systému pro produkční nasazení. V praktické části bakalářské práce je vybraný monitorovací systém nasazen v reálném firemním prostředí a nakonfigurován pro reálný chod.

Klíčová slova

monitoring, počítačové sítě, dohledové systémy, Nagios, OpsView, dohled, Zabbix, Zenoss, Icinga, správa sítí

Title

Monitoring systems for computer networks.

Annotation

The subject of this thesis is to analyze the possibility of monitoring and management of computer networks. The thesis presents the principles of monitoring and open source systems. The result of the theoretical part is to select an appropriate monitoring system for production using. In the practical part of the thesis, the selected monitoring system is deployed in a real business environment and configured for the real operations.

Keywords

monitoring, computer network, monitoring systems, Nagios, OpsView, Zabbix, Zenoss, Icinga, network management

Obsah

Seznam zkratek

PAN	Personal Area Network
MAN	Metropolitan Area Network
LAN	Local Area Network
CAN	Campus Area Network
WAN	Wide Area Network
OS	Operační Systém
IP	Internet Protocol
IM	Instant Messaging
SMS	Short Message Service
MMS	Multimedia Messaging Service
RSS	Really Simple Syndication
LAMP	Linux Apache MySQL PHP
ICANN	Internet Corporation for Assigned Names and Numbers
IANA	Internet Assigned Numbers Authority
ICMP	Internet Control Message Protocol
RFC	Request For Comments
SNMP	Simple Network Management Protocol
HTTP	HyperText Transfer Protocol
FTP	File Transfer Protocol
IMAP	Internet Message Access Protocol
POP	Post Office Protocol
SMTP	Simple Message Transfer Protocol
SSH	Secure SHell
DNS	Domain Name Server
NTP	Network Time Protocol
RAM	Random Access Memory
HDD	Hard Disk Drive
TCP	Transmission Control Protocol
UDP	User Datagram Protocol
PoE	Power over Ethernet
GPL	General Public License
HW	HardWare
SW	SoftWare
PC	Personal Computer
IT	Informační Technologie
IIS	Internet Information Services

Seznam obrázků

Seznam tabulek

Úvod

Oblast informačních technologii je jednou z nejrychleji rostoucích technologických oblastí na světě. Nové technologie přicházejí každým dnem a možnosti informačních technologií se rozšiřují každým okamžikem. S novými možnostmi přichází také mnohem větší výpočetní náročnost, větší nároky na datový prostor a rychlejší konektivitu mezi jednotlivými komunikačními uzly. Zvyšující se nároky na výkon sítě, ale i na samotný provoz služeb, přímou úměrou se zvyšují i nároky na správu, údržbu a management počítačových sítí. Po provozovatelích služeb se vyžaduje maximální funkčnost a, v ideálním případě, nulová chybovost při provoze. Administrátoři sítí tak v případě problémů musí zasáhnout co nejrychleji a nejefektněji, aby omezili dopad problému na koncové uživatele.

Správně nakonfigurovaný dohledový systém je pravou rukou administrátora sítě. Pomáhá mu analyzovat počítačovou síť a v reálném čase rychle identifikovat problém. To administrátorovi umožní nalézt řešení a problém odstranit dříve, než chybu zaregistruje koncový uživatel. Dohledový systém nemá v popisu práce pouze monitoring aktivních prvků sítě, jako jsou například switche a routry, ale monitoruje také využívané služby, protokoly a provozní parametry serverů a stanic. Nasazením monitorovacího systému získá správce sítě neocenitelného pomocníka, který šetří jeho čas a zaměstnavateli finance vynaložené na jinak zdlouhavé odstranění problému.

Dohledové systémy už nejsou důležitým článkem funkční sítě jen u velkých firem, a tak se velmi navýšila poptávka a nároky na vlastnosti dohledových systémů. Na trhu tak dnes můžeme najít velké množství komerčních i open source produktů, které umožňují sledování od několika počítačů, až po rozsáhlé sítě s velkým množství stanic, serverů a služeb. Komerční nástroje se od open source liší především v poskytování technické podpory, open source nástroje naopak těží z velké flexibility s možností uživatelského přizpůsobení řešení přímo pro potřeby zákazníka.

Cíl práce

Cílem této bakalářské práce je průzkum možností současných nástrojů pro monitoring a management počítačových sítí a získání informací o vlastnostech dostupných dohledových systémů a řešení. Na základě výsledků průzkumu a výsledků analýzy konkrétního firemního prostředí budou stanoveny kritéria pro výběr vhodného dohledového systému. Vybraný dohledový systém bude prakticky nasazen v analyzovaném firemní prostředí a následně bude vyhodnocen jeho reálný chod minimálně po dobu 2 měsíců.

1 Seznámení s problematikou

1.1 Co jsou to počítačové sítě

Vznik počítačové sítě můžeme definovat jako okamžik spojení dvou a více počítačů pomocí telekomunikačního systému za účelem sdílení zdrojů a vzájemné komunikace. V současnosti je v praxi nejrozšířenější síť s technologií ethernet s využitím protokolu TCP/IP.

Sítě můžeme dělit podle velkého množství parametrů. Dle použité technologie komunikace na Ethernet, Token Ring, Frame Relay atd., síťové topologie na sběrnici, hvězdu a další, velikosti na PAN, LAN, CAN, MAN a WAN případně i podle použitého komunikačního protokolu jako jsou například TCP/IP, IPX/SPX, AppleTalk, anebo technologie zapojení na drátové, optické nebo bezdrátové. (1)

1.2 Co je to dohledový systém

Dohledový systém je aplikační nástroj umožňující správci sítě v reálném čase monitorovat parametry sítě nebo zpětně dohledat stav sítě v konkrétní časový okamžik. Dohledové systémy (často také monitorovací systémy) jsou komplexní nástroje pro centralizovanou správu a analýzu sítě. Můžeme monitorovat nejen veškeré parametry serverů, pracovních stanic a síťových prvků, jako jsou například dostupnost, obsazenost datových úložišť, teploty na důležitých hardwarových prvcích, stavy baterií v záložních zdrojích napájení, ale také spuštěné služby a aplikace na serverech i pracovních stanicích. Monitorovací nástroje jsou multiplatformní a můžou tedy běžet na všech dnes známých operačních systémech, jako jsou například UNIX, MacOS, Windows a další. (2) (3)

Monitorovací systémy nám poskytují statistiky a grafy z nasbíraných dat, nabízejí několik způsobů varování (např.: SMS, email, IM, pager) na vyskytlé situace, případně automatické nebo poloautomatické zásahové akce pro napravení vzniklých problémů a zdokumentování situace i její nápravy do souborů či databáze. Všechny tyto vlastnosti dohledových systémů dávají administrátorům sítě možnost snížit čas potřebný pro odhalení problému i celkový downtime[1] zařízení ve spravované síti. Na základě nasbíraných dat pak může administrátor sítě problémovým situacím předcházet, anebo plánovat další využití zařízení. (3)

Očekávaným důsledkem nasazení dohledového systému je maximalizace spolehlivosti sítě, síťových prvků a zařízení v síti, především pak klíčových serverů a páteřních rozvodů, snížení reakčních časů administrátorů a downtime zařízení, díky včasnému varování a automatickým zásahům systému.

[1] Doba, po kterou není server v provozu, nebo neplní svojí funkci.

1.3 Složení monitorovacího systému

Moderní monitorovací systémy využívají tzv. pro-aktivní monitorování. Pro-aktivním monitoringem se rozumí sledování událostí na monitorované stanici agentem. Dohled tedy funguje na principu modelu klient – server. Serverová část, která je centrálním uzlem, se často nazývá jako manager a klientská část se označuje jako agent. (4) (5)

Manager periodicky sbírá data od agentů a zajišťuje jejich další zpracování. Administrátor sítě má k dispozici webové nebo jiné grafické rozhraní, které mu umožňuje nejen zobrazovat nashromážděná data, ale i systém spravovat a provádět konfiguraci managera.

Agent, sloužící jako zdroj informací, běží na monitorovaném zařízení a v reálném čase předává managerovi informace o vlastnostech a stavu zařízení, která si manager vyžádal.

1.4 Vlastnosti monitorovacích systémů

1.4.1 Informace o síti

- Systém poskytuje statistiky (vytížení, přístupů, teplot, a další) s denním, týdenním a měsíčním přehledem. Umožňuje export statistik například do PDF.
- Dále zaznamenává změny v monitorovaných systémech, chyby a varování do log souboru opět s možností exportu do jiných formátů pro další zpracování nebo tisk.
- Nasbíraná data zpracovává do grafů, které nám umožňují snazší orientaci ve statistických datech a mohou správci sítě napovědět při predikci problémů a hledání slabých míst.
- Webové rozhraní pro přehled o síti a její správu. Umožňuje měnit konfiguraci systému, přidávat, editovat, mazat klienty a monitorované informace o jejich činnosti. Poskytuje přehled statistik, grafů a logů na centralizovaném místě dostupném přes webový prohlížeč.

1.4.2 Inventarizace sítě

- Monitorovací systém je schopen zjistit všechny aktivní prvky v síti s vlastní IP adresou.
- U zjištěných prvků detekuje OS nebo verzi firmwaru, u počítačů zjistí nainstalované hotfixy a aktualizace, případně další software.
- Ke každému zařízení v síti přiřadí jeho MAC adresu případně i jeho síťový název pro snadnou identifikaci.
- Ze zjištěných informací vytvoří grafickou podobu sítě s popisy jednotlivých prvků.
- Všechny výše popsaná data může automaticky obnovovat.

1.4.3 Monitoring sítě

- Monitorujeme stavy TCP/UDP portů.
- Monitoring služeb zajišťuje kontrolu běhu důležitých aplikačních částí serveru, jako mohou být Apache a přístup k databázi u webového serveru nebo stav emailové fronty mail serveru.
- Monitoring serverů a stanic nás informuje o základním stavu serveru nebo stanice (dostupná/nedostupná).
- Monitoring hardwarových částí a jejich vlastností nám poskytuje informace o teplotách procesoru, obsazenosti disků, velikosti volné operační paměti nebo rychlosti otáček větráků.

1.4.4 Řešení problémů na síti

- Při identifikaci nedostatku místa na discích monitorovaných stanic spustí čištění konkrétního disku a vše řádně zaprotokoluje.
- Porovnáním seznamu monitorovaných stanic s aktuálním stavem sítě identifikuje neautorizovaný vstup do sítě. Reakcí může být automatické odpojení stanic, nebo pouze varování administrátora.
- Systém kontroluje bezpečnostní aplikace, jako je například firewall nebo antivirus, v případě, že není služba aktivní, pokusí se ji spustit. Zajišťuje také aktualizaci antivirových softwarů.

1.5 Kategorie monitorovacích systému a nástrojů

Monitorovací systémy a nástroje se rozdělují do kategorií podle funkcí, kterými disponují. Toto rozdělení slouží k lepší orientaci a rychlejšímu nalezení potřebného nástroje nebo systému. Kategorie se často prolínají a doplňují, a proto téměř žádný nástroj nebo systém nespadá pouze do jedné kategorie. Kategorií je velké množství a některé se liší pouze nepatrně, jiné naopak sdružují systémy, které obsahují vlastnosti několika dalších kategorií. Podrobněji se budeme věnovat jenom několika vybraným kategoriím, které jsou pomyslnými stavebními kameny monitoringu. Nezmíněné kategorie jsou spíše podmnožinou popsaných kategorií. Jedná se například o Unix/Linux Monitoring, SNMP monitoring, Web monitoring nebo VMware Monitoring. (6) (5) (7)

1.5.1 Alerting

Úkolem alerting systémů je co nejrychleji a nejefektivněji informovat administrátora o vzniklém problému nebo chybě na monitorované síti. K informování administrátora využívají různé komunikační služby. Nejčastěji využívají varování SMS zprávami nebo emaily, ale velmi oblíbené jsou také IM komunikátory nebo RSS kanály. Alerting systémy mohou být integrované v monitorovacích systémech, anebo operují jako samotné systémy rozšiřující možnosti monitoringu. (6) (5) (7)

Způsoby varování administrátora:

- SMS/MMS zprávami na mobilní telefon,
- zasíláním zpráv přes IM komunikátory,
- využitím RSS kanálů,
- emailovými zprávami,
- tvorbou log souborů,
- přes webové rozhraní monitorovacího systému,
- využitím Twitter.com,
- varováním na pager.

1.5.2 Application monitoring

Application monitoring nám umožňuje monitorovat stav a funkce aplikací v reálném čase. Vytvoří kompletní přehled o chování nejen business aplikací, pomáhá rychle detekovat a diagnostikovat problémy aplikačních serverů a jejich služeb, které by mohli mít dopad i na koncové uživatele. U monitorovaných aplikací tak můžeme získat přehled například o velikosti alokované paměti, počtu běžících vláken nebo počtu otevřených síťových spojení. (6) (5) (7)

Možnosti application monitoringu:

- monitoring web-serverových služeb (LAMP),
- monitoring všech majoritních Java serverových aplikací jako například Oracle WebLogic, GlassFish, Apache Tomcat, JRun a další,
- monitoring Microsoft .Net,
- monitoring databázových systémů s podporou nejznámějších řešení jako jsou Oracle, MS SQL, MySQL, PostgreSQL, D2 a další. Monitoruje se například dostupnost databáze, konektivita databázového klienta nebo velikost datového prostoru.

1.5.3 Cloud monitoring

U cloudových řešení je důležité monitorovat nejen jednotlivé instance virtuálních serverů, ale také HW, nad kterým cloud provozujeme. U jednotlivých instancí nás zajímají především hodnoty aktuální alokované paměti, vytížení procesoru v čase nebo stav vláken v průběhu času. Naopak u HW monitorujeme především celkový stav paměti a zatížení procesoru a celkový počet spuštěných virtuálních instancí. Získaná data nám pomáhají plánovat vytížení HW a predikovat trendy v jeho využívání. To nám umožní zefektivnit nejen chod samotného cloudu ale i investice do jeho rozvoje a udržování. Nástroje pro

cloud monitoring podporují většinu nejrozšířenějších platforem, jako jsou VMWare, Xen, Hyper-V nebo Amazon EC2. (6) (5) (7)

1.5.4 Database monitoring

Database monitoring je specializovaná část application monitoringu, která monitoruje výhradně databázové systémy a jejich chod. Aktivní monitoring databáze nám pomáhá včas detekovat a odstranit problémy, dříve než se projeví v chování samotné databáze u koncového uživatele. Monitorovací systém optimalizuje strukturu databáze pro lepší využití HW zdrojů a pomáhá nám v plánování dalšího zatížení serveru. Dále měří výkonové zatížení jednotlivými databázovými dotazy nad tabulkami a sloupci, sleduje propustnost transakcí a aktivních spojení a analyzuje možnosti zlepšení výkonu databáze agregováním často volaných tabulek. Výhodou těchto systému je také schopnost opravit, případně obnovit, poškozenou databázi do původního stavu. Samozřejmostí je podpora nejrozšířenějších databázových řešení (MySQL, Oracle, PostgreSQL, D2, MS SQL, atd.). (6) (5) (7)

1.5.5 Enterprise monitoring

Enterprise monitoring systémy jsou komplexní nástroje pro real-time monitoring velkých podnikových sítí. Kombinují vlastnosti všech ostatních kategorií pro dosažení maximální efektivity dohledu. Sjednocení všech druhů monitoringu do jednoho nástroje nám dává mnohem lepší přehled o stavu monitorované sítě, usnadňuje konfiguraci a poskytuje ucelená data pro další analýzy a plánování využití sítě. Samozřejmostí je grafické uživatelské rozhraní pro konfiguraci a správu sítě s možností exportu monitorovaných hodnot do souborů XML, PDF nebo databáze a systém včasného varování pověřených osob na vzniklé problémy. (6) (5) (7)

Enterprise systémy jsou často poskytovány velkými společnostmi jako komerční produkty. Komerční produkty, na rozdíl od open source řešení, poskytují zákazníkům kvalitní technickou podporu pro řešení problémů a technickou pomoc při zavádění systému.

1.5.6 Environment monitoring

Pro provoz serverů je velmi důležité prostřední, ve kterém jsou umístěny. Místnosti, kde se servery nacházejí, by měli být klimatizované pro udržování stálé provozní teploty. Důležitá je také relativní vlhkost vzduchu v místnosti. Environment monitoring se zabývá právě sledováním prostředí v serverovnách. (6) (5) (7)

Jedná se převážně o HW řešení, která se skládají ze senzorů a centrální jednotky, která vyhodnocuje měřená data. Senzory jsou většinou napájeny PoE. Centrální jednotky jsou často upraveny pro umístění do rackových skříní jako 1U rack, napájeny bývají ze sítě a doporučuje se využití záložního zdroje z důvodů včasného varování při výpadcích energie. Konfigurace centrálních jednotek probíhá přes webové rozhrání, které nám umožní nastavit všechny senzory, ale také způsob varování (nejčastěji email, nebo SMS).

Možnosti environment monitoringu:

- monitoring teploty v místnosti,
- monitoring vlhkosti vzduchu,
- sledování proudění vzduchu a klimatizace,
- detekce kouře,
- detekce kapalin,
- detekce pohybu a neoprávněného vstupu do místnosti,
- sledování úrovně napětí (přepětí, podpětí),
- detekce vibrací a nárazů.

1.5.7 Event Log monitoring

Event log je jeden z nejjednodušších způsobů monitoringu. K získávání informací nepoužívají event log systémy žádné agenty, ale jsou založeny na přístupu k log souborům na monitorovaných zařízeních. Rozborem log souboru systém získá přehled o proběhlých událostech na sledovaném zařízení. Jednotlivé události může uložit do databáze, vytisknout, a v případě problémů varovat administrátora emailem s podrobnými informacemi. Administrátor situaci vyhodnotí a případně podnikne kroky k odstranění problémů. (6) (5) (7)

Druhy monitorovatelných log souborů:

- log soubory platformy Windows,
- Unix SysLog,
- Solaris, HP-UX, IBM AIX logy,
- logy routrů a switchů (např. Cisco zařízení),
- Microsoft ISS Web a FTF server application,
- Apache web server,
- DHCP Windows application,
- logy databází (MS SQL, Oracle, atd.),
- logy síťových tiskáren.

1.5.8 Network monitoring

Monitoring sítě zajišťuje dohled nad síťovými zařízeními, kontroluje propustnost sítě a dostupnost všech prvků. Nejčastěji monitorujeme datový tok na konkrétním síťovém rozhraní serverů, routrů nebo switchů, propustnost a latency celé infrastruktury nebo stav aktivních síťových prvků, které podporují SNMP protokol. Z naměřených dat sestavujeme analýzy vytížení sítě a zaměřujeme se především na efektivní využití páteřních rozvodů. Network monitoring nástroje umí vytvořit grafickou reprezentaci síťových zařízení včetně jejich označení a IP adres. (6) (5) (7)

Do této kategorie zahrnujeme i specializované nástroje pro měření síťového provozu (trafficu), které se někdy uvádí samostatně. Tyto nástroje monitorují veškerý datový provoz mezi uzly v síti a umožňují nám ho filtrovat a analyzovat v reálném čase. Monitoring umí provádět pro jednotlivá síťová zařízení nebo rozhraní, ale i pro různé síťové protokoly. V rámci filtrování mohou zasahovat i do nastavení síťového HW a SW.

1.5.9 PC monitoring

PC monitoring sleduje využívání jednotlivých stanic v síti. Sleduje veškeré operace, které počítač provádí sám nebo z vůle uživatele. Základními sledovanými vlastnostmi jsou například jména přihlášených uživatelů, spuštěné programy a doby jejich využívání, využívání webového prohlížeče a IM komunikátorů a samozřejmě vlastnosti HW.

Monitorováním programů získáváme přehled o počtu spuštění, aktivním využití a výkonové zátěži jednotlivých programů. Sledování výkonového zatížení jednotlivých programů pomáhá včas detekovat viry a malware. Monitoring webových služeb nám nejen zpřístupní kompletní historii prohlížení webového obsahu, ale také nám umožní filtrovat přístup na webové stránky dle klíčových slov nebo časových pravidel. Nástroje pro PC monitoring poskytují také vzdálené sledování plochy, pohybu kurzoru myši a stisknutých kláves (tyto nástroje nezaznamenávají stisknuté klávesy v přihlašovacích polích) a historie tisku. Běžnou vlastností je podpora vzdálené správy PC s možností ovlivnit i samotný chod počítače (spouštět a ukončovat programy a procesy, restartovat/vypnout PC, odhlásit uživatele). V případě problémů nástroje upozorňují nejen uživatele sledovaného systému, ale i administrátora sítě. (6) (5) (7)

Specializované nástroje, využívající tento způsob monitoringu, často využívají školy a školicí střediska, protože umožňují lektorovi monitorovat ze svého PC činnost všech studentů. Velmi často se s ním také setkáváme ve firmách, které ho nasazují pro zvýšení pracovní morálky svých zaměstnanců.

1.5.10 Performance monitoring

Performance monitoring jsou jednoduché nástroje, které se specializují pouze na měření výkonu serveru a stanic. Velmi často jsou tyto nástroje obsaženy standardně ve větších monitorovacích systémech, ale vzhledem k důležitosti monitoringu výkonu jsou často nasazovány samostatně. U serverů monitorují tři základní parametry: využití procesoru, obsazení operační paměti a volnou kapacitu disků. U pracovních stanic se navíc oproti

serverům může sledovat využití grafického čipu. Samozřejmostí je varování administrátora při dosažení kritických čísel u sledovaných hodnot.

1.5.11 Protocol Analyzing and Packet Capturing

Nástroje z této kategorie zachycují a zaznamenávají veškerý provoz na síti v reálném čase, vzhledem k velkému množství podporovaných síťových protokolů jsou následně schopny analyzovat téměř veškerou síťovou komunikaci. Spolupracují s aktivními prvky sítě, jako jsou switche a routry, a monitorují aktivitu na jejich portech. Umožňují nám sestavovat filtry a monitorovat tak pouze síťový provoz, který nás zajímá.

Některé z těchto nástrojů umí přepnout síťové rozhraní do promiskuitního módu a sledovat tak i komunikaci určenou jiným zařízením. Tyto nástroje slouží především k ladění a vývoji programů komunikujících po síti, pro pochopení síťové komunikace, anebo jsou zneužívány hackery pro přípravu a provedení útoku. (6) (5) (7)

1.5.12 Security monitoring

Bezpečnost je v současné době nejvíce diskutované téma v oblasti IT. Pro počítačové sítě je bezpečnost důležitým faktorem, který rozhodne o úspěchu řešení nebo softwaru. Proto je důležitý i monitoring bezpečnostních prvků sítě. V případě narušení bezpečnosti sítě, hrozí narušení celé síťové infrastruktury a v nejhorším případě i únik nebo ztráta nejen citlivých dat.

Bezpečnost v síti by měla být zabezpečena na několika na sobě nezávislých úrovních. Od úrovně mechanického zabezpečení přístupu k serverům a klíčovým prvkům sítě, přes ochranu stanic a serverů antivirovými softwary a firewally, až po šifrování samotné komunikace, bychom měli bezpečnosti věnovat maximální úsilí.

Nástroje pro bezpečnostní monitoring pomáhají správcům sítě udržet bezpečnost na vysoké úrovni. Sledují stav antivirových řešení a hlídají jejich aktualitu, spolupracují s firewally a monitorují otevřené porty a síťová rozhraní a kontrolují bezpečnostní aktualizace pro operační systém sledovaných stanic. Větší bezpečností systémy disponují i kontrolou zabezpečení přístupu k serverům a klíčovým prvkům sítě například kamerovým systémem nebo pohybovými čidly.

1.6 Způsoby monitorovaní

1.6.1 Ping

Základní diagnostický nástroj umožňující ověřit dostupnost hosta v TCP/IP síti. Funguje na principu zasílání ICMP zpráv a slouží jako sonar. Vyšle dotaz (echo-request) a čeká na odpověď (echo-reply), měřením doby mezi odesláním požadavku a přijmutím odpovědi zjišťuje zpoždění (latency) sítě měřenou v milisekundách. Pokud paket s odpovědí nedorazí, považuje ho ping za ztracený. V případě většího množství ztracených paketů a větší latence mezi měřeným a zdrojovým zařízením můžeme předpokládat problém v rychlosti sítě a dostupnosti zařízení a je třeba odhalit problémové místo. (8)

1.6.2 SNMP

SNMP je standardizovaný jednoduchý protokol aplikační vrstvy sloužící k periodickému získávání informací o stavu sítě. První standard pochází již z roku 1988 a od té doby protokol prošel změnami až do současné verze tři. *„Součástí standardu je i definice databázové struktury a datových objektů, do nichž se data ukládají“* (9)

Přehled verzí

- SNMPv1 – první verze, slabé zabezpečení (heslo, tzv. community string, přenášeno jako plain-text), (10) (11)
- SNMPv2c – vychází z verze 1, přidány nové datové typy a struktury (např. INFORM, GET BULK), vylepšen výkon, není zpětně kompatibilní s první verzí, v současné době asi nejrozšířenější varianta protokolu SNMP,
- SNMPv3 – výrazně zlepšeno zabezpečení přidáním šifrování přenášených dat (na výběr mezi DES a AES (12)) a autentizace (pomocí jména a hesla s možností šifrování hesla pomocí MD5 nebo SHA1 (12)), kontrola integrity paketů.

Protokol SNMP funguje na principu klient – server respektive agent – manager. Agent je typicky realizován démonem snmpd naslouchajícím na portu UDP 161 a běží na sledovaných zařízeních, jako jsou servery, switche, stanice, tiskárny a další aktivní prvky sítě. Manager používá démona snmptrapd, který přijímá zprávy Trap obvykle na UDP portu 162 a běží na monitorovacím serveru. Manager žádosti zasílá z dynamického portu, na který pak agent odpovídá. Pro každý požadavek tak může být využit jiný port. Vzhledem k využití UDP protokolu je SNMP komunikace velmi rychlá, hrozí ale ztráta informací kvůli nespojitému charakteru přenosu dat protokolu UDP. Proto od verze dva je v SNMP implementována kontrola pro doručování dat. (13) (10)

Komunikace v SNMP protokolu probíhá dvěma způsoby. Typicky manager zasílá request (požadavek/dotaz) agentovi a ten na jeho dotaz odpovídá. Ve specifických situacích ovšem může agent zaslat tzv. Trap i bez předchozího dotazu managera. Příkladem takové situace může být například překročení sledovaných hodnot, jako je například teplota procesoru, nebo otáčky větráku. (14)

SNMP operace

- GetRequest – posílá manager, úvodní zpráva komunikace, specifikuje vyžadované informace,
- SetRequest – posílá manager, určuje agentovi, aby nastavil některou z hodnot,
- GetNextRequest – posílá manager, vyžaduje následující záznam ze seznamu,
- GetBulkRequest – posílá manager, umožňuje získat více hodnot jedním dotazem (od verze 2),

- Response – posílá agent, odpověď na dotazy managera,
- Trap – posílá agent, speciální typ, zasílá se při vyskytnutí specifických událostí, není třeba dotaz managera,
- InformRequest – posílá manager, slouží pro výměnu informací mezi managery (od verze 2).

MIB

„MIB (managent information base) je hierarchicky strukturovaná databáze, která se skládá z objektů, které definují vlastnosti zařízení.“ (15)

MIB je hierarchická datová struktura, která reprezentuje strom. A odpovídá konkrétnímu zařízení. MIB se vytváří podle pravidel Structure of Management (SMI) dle specifikací v RFC 2578,2579,2580. (16) (13)

Struktura začíná nepojmenovaným kořenem (označován tečkou) a pod ním se nacházejí uzly. Uzly mohou tvořit kořeny dalším podstromům. Každý uzel má své označení, skládající se z textového popisu a číselné hodnoty. Tato označení by měla být v rámci MIB stromu unikátní, protože slouží k směrování SNMP dotazů. (16) *„MIB se dělí na dvě hlavní části. Standard MIB, která je definována organizací IETF a popisuje univerzální vlastnosti, které většina výrobců implementuje. A Enterprise MIB, kde organizace IANA přiděluje unikátní podstromy jednotlivým společnostem, které si následně sami spravují.“* (2)

Pro jednoznačnou identifikaci objektu v MIB slouží OID (object identifier). OID kopíruje hierarchickou strukturu MIB a se skládá z čísel, všech rodičovských-nadřazených objektů až ke kořenu stromu, oddělených tečkami. Adresu můžeme zapsat také pomocí textového označení jednotlivých uzlů. Příkladem OID může být například **.1.3.6.1.2.1.2.2.1.6.1**, které v textové podobě reprezentuje zápis **.iso.org.dod.internet.mgmt.mib-2.interfaces.ifTable.ifEntry.ifPhysAddress**. (2) (13)

1.7 Sledování portů

Pro navázání úplného síťového spojení musíme znát typ protokolu, IP adresu odesílatele a příjemce, a číslo portu odesílatele a příjemce. K identifikaci portu slouží dvou bajtové číslo, které nám umožňuje adresovat porty v rozsahu 0 – 65535. Můžeme tedy říct, že síťový port slouží při komunikaci pomocí protokolů TCP a UDP k rozlišení jednotlivých služeb. Porty můžeme rozdělit do tří skupin podle rozsahu portů.

- 0 – 1023 – známé porty (well known ports) – definovány v RFC 1700 se používají pro často využívané aplikace (17)
- 1024 – 49151 – registrované porty (registered ports) – pro méně známe aplikace, jejichž použití se může registrovat u ICANN (dříve IANA)

- 49152 – 65535 – dynamické a privátní porty (dynamic and private ports) – používají se pro dynamické přidělování a soukromé využití, nejsou přiděleny žádné konkrétní službě ani aplikaci

Tabulka 1 - přehled nejznámějších portů

port	protokol	služba	popis
4	udp	NTP	Network Time Protocol - protokol pro synchronizaci času
20	tcp / udp	FTP	File Transfer Protocol - datový přenos
21	tcp / udp	FTP	File Transfer Protocol - režijní přenos
22	tcp / udp	SSH	Secure Shell - zabezpečený komunikační protokol pro vzdálenou správu
23	tcp / udp	Telnet	Telnet - nezabezpečený komunikační protokol pro vzdálenou správu
25	tcp / udp	SMTP	Simple Mail Transfer Protocol - protokol pro přenos elektronické pošty
53	tcp / udp	DNS	Domain Name System - protokol sloužící pro výměnu informací o převodu doménových jmen a IP adres
80	tcp / udp	HTTP	HyperText Transfer Protocol - protokol pro přenos hypertextových dokumentů
110	tcp / udp	POP3	Post Office Protocol - protokol používaný pro stahování elektronické pošty
115	tcp	SFTP	SSH File Transfer Protocol - protokol pro zabezpečený přenos dat
143	tcp / udp	IMAP	Internet Message Access Protocol - protokol pro vzdálenou správu emailové schránky
161	tcp / udp	SNMP	Simple Network Management Protocol - protokol pro získávání informací o síťových prvcích
443	tcp / udp	HTTPS	Hypertext Transfer Protocol Secure -nadstavba protokolu HTTP umožňující zabezpečit spojení
3306	tcp	MySQL	MySQL - databázový systém

Rozlišujeme tři stavy portu – otevřený, filtrovaný, uzavřený. Pokud je port otevřený, znamená to, že stanice na tomto portu přijme spojení. Filtrovaný port je takový port, který je chráněn firewallem nebo jiným filtrem zabraňujícím ve spojení s tímto portem. Uzavřený port značí, že na stanici služba využívající tento port není spuštěna.

Pro pochopení způsobu skenování portů je důležité vědět, jak probíhá navázání a ukončení TCP spojení. Pro navázání spojení se používá třícestný handshake, kdy se posílá TCP segment s nastavenými příznaky, číslem sekvence a odpovědi uvedenými v TCP hlavičce. (18)

Navázání probíhá v těchto třech krocích:

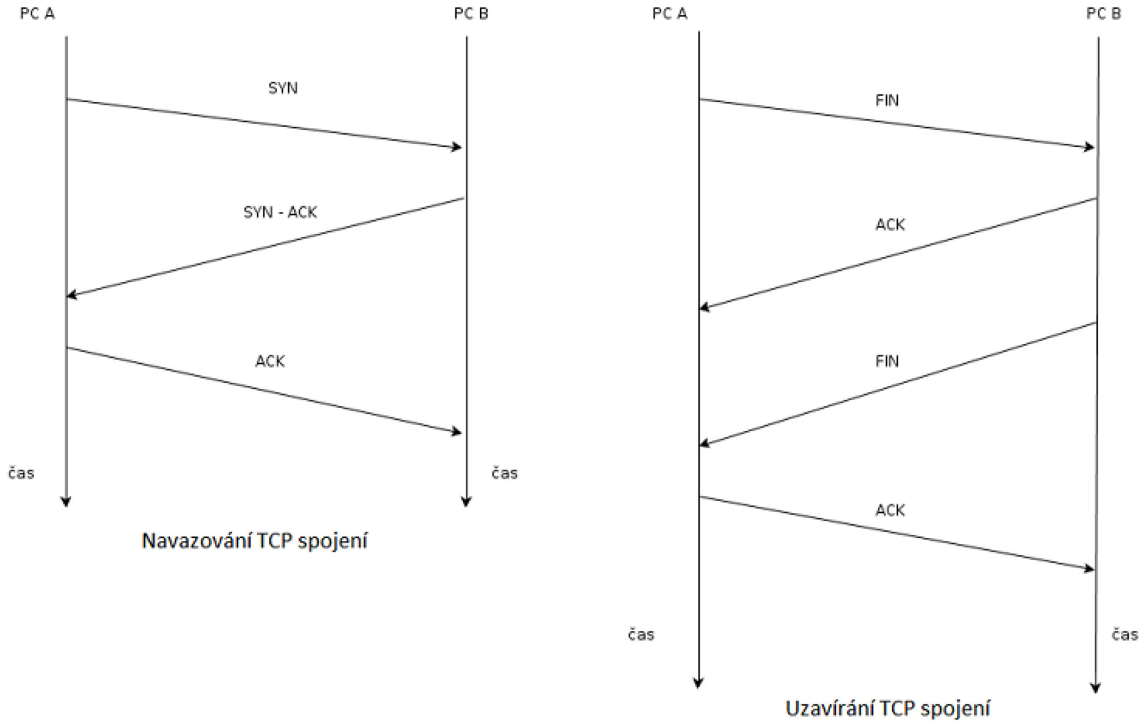
- klient pošle SYN paket,

- druhá strana odpoví paketem SYN-ACK,
- klient odpoví paketem ACK.

Ukončení komunikace se podobá navazování, využívá čtyřcestný handshake, kdy obě komunikující strany samostatně uzavřou spojení.

Ukončení tedy probíhá v těchto krocích:

- klient pošle FIN paket,
- druhá strana odpoví paketem ACK,
- druhá strana pošle FIN paket,
- klient odpoví paketem ACK.



Obrázek 1 - Navazování a uzavírání TCP spojení ***zdroj: vlastní***

Způsobů jak skenovat porty existuje mnoho, některé z nich jsou pro administrátory sítě lehce odhalitelné, některé nikoliv. Skenování portů bývá často předzvěstí útoku a proto je v zájmu administrátora sítě, aby je co nejdříve rozpoznal. Při budování sítě se ovšem skenování portů často používá pro určení slabých míst, která jsou potřeba zabezpečit. Mezi nejčastější způsoby skenování portů patří TCP Connect scan, TCP SYN scan, FIN scan a ACK scan, UDP scan.

TCP Connect Scan

Nejjednodušší způsob skenování portů. Provádí se klasické navázání komunikace s cílovým portem. Pokud se navázání podaří, port je otevřen, v opačném případě je uzavřen. Toto skenování je v případě úspěšného navázání spojení zalogováno, a proto se většinou nepoužívá pro přípravu útoku. (19) (20)

TCP SYN scan

Neotevírá celé TCP spojení, ale proběhne pouze první část spojení. Klient pošle pouze SYN paket. Pokud druhá strana odpoví paketem SYN-ACK, port je otevřený, pokud odpoví paketem RST, port je uzavřen. Tyto pokusy o navázání spojení nejsou standardně logovány. Logování můžeme zajistit doinstalováním softwaru, který s takovými pokusy počítá (například firewall). (19) (20)

FIN scan

Posílá paket, který má nastaven pouze příznak FIN používaný pro ukončení spojení. Pokud na takový paket druhá strana odpoví paketem RST, port je uzavřen. Pokud druhá strana paket ignoruje (klient, iniciující spojení, neobdrží žádnou odpověď) je port otevřen. Některé operační systémy (například Windows XP) odpovídají paketem RST i v případě, že port je otevřen. Toto chování je nestandardní a v rozporu s RFC 793 (16) (19)

ACK scan

Slouží pro zjištění filtrovaných portů. Klient zasílá paket s příznakem ACK a náhodně generovaným číslem sekvence. Pokud druhá strana odpoví paketem RST, port je uzavřen a není filtrován. Pokud neobdrží klient žádnou odpověď nebo odpověď jinou, znamená to, že port je filtrovaný. (19) (20)

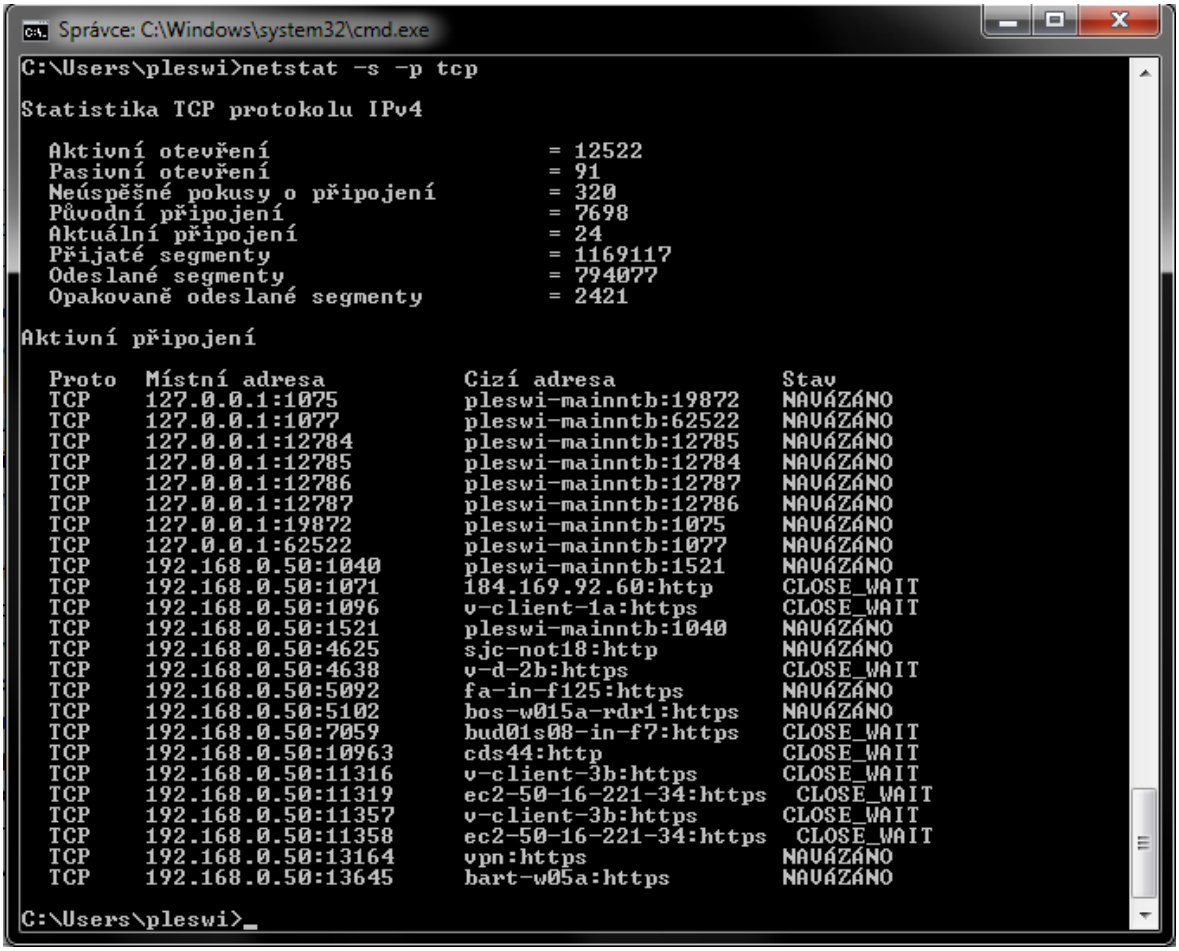
UDP scan

UDP spojení na rozdíl od TCP spojení nenavazuje spojení třícestně, ale již od prvního paketu posílá data. Tento rozdíl je dán rozdílnou filozofií UDP protokolu. Skenování probíhá zasíláním prázdných UDP paketů. Pokud na portu žádná služba neběží, obdržíme jako odpověď paket ICMP s informací, že port je nedostupný. Pokud neobdržíme nic nebo obdržíme nějaká data odeslaná z tohoto UDP portu, port je otevřen. Nevýhodou tohoto způsobu je časté filtrování ICMP zpráv na síti, proto nemůžeme jednoznačně určit, zda je port uzavřen. (19) (20)

1.8 Netstat

Nástroj Netstat, běžící v příkazové řádce, poskytuje řadu informací o stavu sítě. Umožňuje zobrazit aktivní síťová připojení, porty na kterých jsou spojení navázána a protokol na kterém probíhají. Zobrazuje také směrovací tabulku, anebo statistiky síťových rozhraní a protokolů. Netstat můžeme využít jak na platformách Windows, tak na Unixových

systémech. O všech možnostech nástroje se můžeme dočíst například v nápovědě nebo manuálových stránkách. (21) (19)



Obrázek 2 - Výstup programu netstat *zdroj:vlastní*

2 Výběr a implementace vhodného systému

2.1 Analýza firemního prostředí

Nasazení monitorovacího systému musí předcházet důkladná analýza prostředí, které má systém monitorovat. Při vypracování této bakalářské práce mi vyšla vstříc pražská firma Arit s.r.o., která se zabývá outsourcingem informačních technologií. Firma Arit s.r.o. se již nějakou dobu o možnosti monitorovacích systémů zajímala, a proto mi umožnila provést praktickou část bakalářské práce ve své síti.

Firemní servery jsou umístěny ve stejné budově jako kanceláře. Servery a stanice zákazníků jsou rozmístěny po celé Praze, několik firemních zákazníků se nachází i mimo území Prahy.

Vedení firmy rozhodlo, že do samotného dohledového systému nebude investovat žádné finance a využije některé z open source řešení. Vzhledem k povaze poskytovaných služeb firmou Arit s.r.o. by systém měl být pružný a snadno modifikovatelný pro monitoring specifických požadavků zákazníka. Pro konfiguraci je vyžadováno webové rozhraní s možností plné konfigurace systému. Systém by měl mít možnost varovat techniky a administrátora na vzniklé problémy emailovou zprávou.

Motivací pro zavedení dohledového systému bylo v první řadě zvýšení informovanosti servisních techniků o stavu spravovaných serverů. Doposud veškeré informace pocházeli pouze z pravidelných servisních návštěv u zákazníků a byli nekomplexní a neaktuální. To byl velmi nevyhovující při plánování údržby a servisu spravovaných serverů. Očekává se i snížení mimořádných výjezdů servisních techniků a snížení downtime serverů pomocí včasných zásahů.

2.1.1 Server pro monitoring

Firma disponuje čtyřmi fyzickými servery, na kterých provozuje ve virtuálním prostředí několik serverů pro různé služby. Pro monitoring server byl vyčleněn jeden virtuální server s parametry 2x jádro 2,5 Ghz, 3GB RAM, 50GB HDD pro systém a 100GB HDD pro databázi v prostředí VMware Server. Všechny firemní servery běží na platformě Linux s distribucí CentOS ve verzi 5.6 x64 nebo 6 x64. Databázová část dohledového systému může využívat MySQL nebo PostgreSQL databáze.

2.2 Analýza monitorovaného prostředí

Firma poskytuje služby přibližně třem desítkám stálých zákazníků a pro desítku zákazníků pracuje pouze nárazově. Vedení firmy by chtělo využít dohledový systém pro obě skupiny zákazníků i pro své vlastní servery. V plánu je tedy monitorovat zhruba 80 serverů a zařízení.

Monitoring bude probíhat na různých platformách a zařízeních. Pro servery bude využito pro-aktivní monitorování pomocí agenta umístěného na každém z monitorovaných serverů.

U zařízení bez možnosti instalace agenta, ale s podporou SNMP protokolu, bude využito pro monitoring SNMP zpráv. Zařízení, bez podpory SNMP protokolu a možnosti instalace agenta bude monitorováno pouze využitím nástroje ping.

2.2.1 Monitorované platformy

Většina monitorovaných serverů běží na platformě Linux s instalovanými distribucemi CentOS 5.4 x64 a CentOS 5.6 x64. Několik serverů využívá distribuci OpenSuse 11 x64. Monitoring bude probíhat i na serverech využívajících platformou Windows s operačním systémem Windows Server 2003 a Windows Server 2008 v 64 bitové verzi. Některé ze serverů běží ve virtuálním prostředí. V monitorované síti se v současné době nevyskytují žádná zařízení na platformě Mac OS-X nebo Solaris.

Mezi monitorovanými zařízeními se nachází síťové tiskárny značky HP a Xerox a routery značky AirLive a Mikrotik.

2.2.2 Monitorované parametry a služby

Vzhledem k počtu serverů a rozmanitosti poskytovaných služeb zákazníky firmy Arit s.r.o. je důležité specifikovat, jaké parametry a služby by dohledový systém měl sledovat. Samozřejmostí je základní monitoring důležitých HW částí serverů, do kterého zahrnujeme měření teplot CPU a HDD, měření zatížení CPU, velikost volné operační paměti a volného místa na discích, měření rychlostí otáček větráků. Většinu těchto parametrů budeme sledovat na všech serverech.

Na back-up serverech, u kterých jsou velmi exponované datové disky, na které se ukládají zálohy dat z jiných serverů, budeme monitorovat stav RAID pole, počet transakcí za sekundu a počty realokovaných sektorů.

U webových serverů je důležité monitorovat služby potřebné pro běh webové prezentace. Mezi takové služby patří například Apache nebo IIS a databáze. Další důležitým parametrem je dostupnost protokolů HTTP/S a FTP. Sleduje se také počet otevřených TCP spojení těchto protokolů.

Mezi sledované parametry emailový serverů patří stavy protokolů IMAP, POP3 a SMTP. Důležité je sledovat také stav emailové fronty a dostupnost připojení k síti internet.

U serverů se specifickými funkcemi se dohled oproti výše zmíněným skupinám bude lišit především ve sledování funkčnosti specifických služeb a činnosti konkrétních protokolů. Monitorovat se budou například vizualizační nástroje a v nich vizualizované systémy nebo výstupní logy automatizovaných činností specializovaných programů zákazníka.

2.3 Kritéria pro výběr systému

Na základě analýzy firemního a monitorovaného prostředí byly sestaveny následující kritéria pro výběr vhodného dohledového systému, který bude ve firmě Arit s.r.o. implementován do provozu.

Kritéria pro výběr:

- open source řešení,
- možnost sledovat vlastní parametry,
- serverová část s podporou distribuce CentOS,
- databázová část s podporou MySQL nebo PostgreSQL,
- podpora SNMP protokolu,
- webové konfigurační rozhraní,
- monitoring všech zmíněných platforem a distribucí,
- varování na vzniklé problémy emailem, případně i RSS kanálem,
- aktivní vývoj aplikace.

2.4 Vybraní kandidáti

2.4.1 Icinga

Icinga je relativně mladý dohledový systém, který vznikl v roce 2009 jako fork projektu Nagios a je distribuován pod licencí GPL V2. Jako fork projektu Nagios umožňuje využít jeho konfigurací, pluginů a rozšíření. Tato vlastnost umožňuje velmi rychlý a pohodlný přechod z Nagios na Icinga. (22)

Již první verze změnila architekturu systému (23) rozdělením jednotlivých části systému do samotných instancí a pro komunikaci zavedla API. Další vývoj směroval k podpoře různých databázových řešení. To si vyžádalo změnu modulů pro komunikaci s databázemi. V dnešní době Icinga již podporuje MySQL, Oracle i PostreSQL. Velkou změnou prošlo i webové rozhraní, které ovšem u mnoha uživatelů, kteří přešli z Nagiosu nesklidilo úspěch. Proto se vývojáři rozhodli umožnit využívat i „klasické“ webové rozhraní se strukturou zobrazování jako má Nagios. Nové i klasické webové rozhraní jsou vyvíjeny tak, že klasické rozhraní reflektuje důležité vlastnosti nového a prezentuje je v jiné podobě. Projekt Icinga poskytuje také mobilní aplikaci pro platformy iOS, Android, BlackBerry a webOS.

Na webu projektu se zatím bohužel nedá zjistit, jaké jsou hardwarové nároky na monitorovací server a velmi obtížně se dohledává seznam podporovaných platforem pro server i klienta, který se zatím nachází hluboko ve wikipedii projektu. Na dotaz o podporovaných platformách support odpověděl během 14 dní. Icinga Agent podporuje všechny Linuxové platformy ovšem platforma MacOS není vůbec podporována a platforma Windows se monitoruje přes agenta NSClient++, který byl vyvinut pro Nagios a v Incinga je podporován díky kompatibilitě pluginů. (24)

Při testování byla využita virtuální verze postavena na distribuci CentOS 6 a standardní instalace systému ve verzi 1.6.1 na distribuci CentOS 5.6 ve virtuálním prostřední. Pro chod virtuální verze byl využit VMWare Player 4 s parametry: 1 jádro, 2,2Ghz, 2GB RAM, 40 GB HDD. Virtuální verze zatím není určena pro ostrý provoz, slouží pro otestování systému v reálné síti a pro případné testování nových nastavení. Při klasické instalaci bylo postupováno dle oficiálního návodu.

2.4.2 Nagios

Nagios je open source nástroj pro monitoring sítí. Jeho počátky se datují do roku 1996 a jeho původní název „NetSaint" se v roce 2002 změnil na dnešní název „Nagios". Projekt několikrát získal ocenění od LinuxQuestions.org v kategorii „Monitoring Application of the Year" (25).

Nagios funguje na principu klient - server, kde klient je tzv. agent na monitorovaném systému a server je tzv. manager, běžící na monitorovacím serveru. Komunikaci mezi managerem a agentem řídí manager, který periodicky zažádá o konkrétní informace, které mu agent následně zašle. Samotné jádro Nagios je celkem holé a mnoha funkcemi (oproti jiným systémům) neoplývá. Síla Nagiosu ale spočívá v pluginech, nadstavbách a add-on které rozšiřují možnosti monitoringu a umožňují tak administrátorům přizpůsobit dohled přesně na míru konkrétní síti. (26)

Pluginy a addony jsou převážně tvořeny komunitou uživatelů a shromažďovány na webu Nagios Exchange. Pokud by administrátor nenašel rozšíření, které potřebuje, může si napsat vlastní, a to hned v několika jazycích, jako jsou například BASH, Python, C++ a další. Na webu Nagios Exchange pak nalezneme i informace o nadstavbách, které jsou převážně vyvíjeny jako samostatné produkty jinými společnostmi nebo komunitami. Naopak nevýhodou Nagiosu, kterou mu uživatelé často vytýkají, jsou vysoké režijní náklady na síťový provoz podpora pouze MySQL databáze a relativně složitá konfigurace.

2.4.3 OpsView

OpsView je původně open source nadstavba nad Nagios Core. Vznik projektu se datuje do roku 2003, v roce 2005 byla založena společnost Opsera Limited. OpsView podporuje všechny známé serverové systémy Linux, jako jsou Debian, CentOS, Ubuntu, RHEL a SUSE. OpsView, stejně jako Nagios, podporuje pouze MySQL databázi, ale oproti Nagiosu má mnohem více uživatelsky přívětivé rozhraní pro konfiguraci a přehledněji zpracovaná data získaná data od klientů. (27)

V roce 2008 se začal vývoj projektu dělit na dvě verze – Community a Enterprise. Obě verze byly postaveny na Nagios jádře a byli tak plně závisle na jeho možnostech. Verze Community byla poskytována zdarma pod licenci GPL, na rozdíl od Enterprise verze neměla technickou podporu a podporu varování pomocí SMS zpráv (tato funkce byla dostupná jen ve vybraných zemích světa). Vzhledem k silné komunitě používající kombinaci Nagios jádra a OpsView Community edition vzniklo komunitní fórum s velkým množstvím zkušených přispěvatelů a step-by-step návodů.

V prvním čtvrtletí roku 2012 se projekt OpsView znovu transformoval a zvolil jinou ekonomickou a vývojovou politiku. V současné době poskytuje OpsView Core, OpsView PRO, OpsView Enterprise a OpsView MSP Zone. Zdarma se nadále poskytuje pouze OpsView Core, který nemá technickou podporu, a všechny funkce si musí administrátoři nakonfigurovat od začátku sami. Ostatní verze jsou komerční a liší se technickou podporou, maximálním počtem monitorovaných zařízení.

Nové jádro je už vyvíjeno nezávisle na Nagios jádře a je u něj deklarována 100% zpětná kompatibilita. Oproti Nagios jádru přibyl Cloud monitoring a Virtual Monitoring, který se u Nagiosu musel řešit rozšířeními a pluginy. Instalaci je možné umístit do cloudu, nebo je možnost využít připravených instalací na vybraných distribucích Linuxu, které stačí nahrát do VMware a spustit. Samozřejmě je i možnost čisté instalace na HW server. (28)

Při testování byla dostupná pouze Community Edition s Nagios jádrem. Využita byla virtuální verze postavena na distribuci CentOS 5.6 s Community Edition verze 3.11. Pro chod virtuální verze byl využit VMWare Player 4 s parametry: 1 jádro, 2,2Ghz, 2GB RAM, 40 GB HDD. Standardní instalace systému byla prováděna na distribuci CentOS 6 s parametry 1 jádro 2,5Ghz, 3GB RAM a 100GB HDD. Při instalaci bylo postupováno dle oficiálního návodu a instalace proběhla na čistou instalaci distribuce.

2.4.4 Zenoss

Počátky vývoje projektu Zenoss se datují do roku 2002, o tři roky později vznikla společnost Zenoss Inc a byla vydána první verze Zenoss Core. Projekt vznikl na popud nespokojenosti autorů s nabídkou a možnostmi monitorovacích systému v době vzniku. Zenoss Core je vyvíjen jako open source pod licencí GPL a za projektem stojí silná komunita. Jádro Zenoss je využito i v komerčním řešení Zenoss Enterprise, které navíc oproti bezplatné verzi přináší plnou technickou podporu a několik rozšiřujících funkcí a financuje tak vývoj Zenoss Core. (29)

Zennoss Core v současné době, jako většina monitorovacích systémů, podporuje všechny často používané serverové distribuce Linuxu jako například Centos, RHEL, Fedora, Debian, SUSE a v posledních verzích přibyla i podpora VMware a MacOS. Monitoring může probíhat na platformách Linux, Mac i Windows. Databáze je postavena pouze pro MySQL řešení. Vývojáři přistoupili také k podpoře Nagios pluginů, které výrazně rozšiřují možnosti systému. (30)

Pro testování byla využita virtuální verze postavena na distribuci CentOS 5.6 s Zenoss Core verze 3.1. Pro chod virtuální verze byl využit VMWare Player 4 s parametry: 1 jádro, 2,2Ghz, 2GB RAM, 40 GB HDD.

2.4.5 Zabbix

Zabbix začal vývoj v roce 1998 jako interní projekt banky. V roce 2001 byla uvolněna první verze pod licencí GPL, ale trvalo tři další roky do vydání první stable verze. V květnu roku 2012 byla uvolněna stable verze 2.0. Komerční verze produktu poskytuje

technickou podporu v několika úrovních například podle maximálního času servisního zásahu. (31)

Zabbix agent podporuje monitoring všech Linuxových platforem, MacOS i Windows. Serverová část nepodporuje pouze Windows platformu. Zabbix podporuje velké množství databázových řešení. Na výběr je MySQL, PostgreSQL, Oracle, SQLite a IBM DB2. Na rozdíl od většiny open source dohledových systémů Zabbix podporuje i komerční databázová řešení. To je dáno tím, že Zabbix vznikal jako projekt na bankovní půdě, kde se tato řešení často používají. Zabbix poskytuje rozsáhle API, díky kterému vznikají například mobilní aplikace třetích stran. (32)

Pro testování byla využita virtuální verze postavena na distribuci OpenSUSE se Zabbix verze 1.8. Pro chod virtuální verze byl využit VMWare Player 4 s parametry: 1 jádro, 2,2Ghz, 2GB RAM, 40 GB HDD. Standardní instalace systému byla prováděna na čistou instalaci distribuce CentOS 6 s parametry 1 jádro 2,5Ghz, 3GB RAM a 100GB HDD. Při instalaci bylo postupováno dle oficiálního návodu produktu.

2.5 Zhodnocení kandidátů a volba řešení

Tabulka 2 - Přehled vlastností vybraných dohledových systémů

	Icinga	Nagios	OpsView	Zenoss	Zabbix
Open source	X	X	X	X	X
Sledování vlastních parametrů	Nagios Plugins	Plugins	Nagios Plugins	Nagios Plugins	Plugins
MySQL	X	X	X	X	X
PostgreSQL	X	O	O	O	X
Webové konfigurační rozhraní	X	X	X	X	X
Serverová část s podporou Centos 5.6+	X	X	X	X	X
Aktivní vývoj	X	X	X	X	X
Podpora SNMP	Pouze přes plugin	Pouze přes plugin	X	X	X
Varování emailem	X	X	X	X	X
Síťové mapy	X	X	X	X	X
Podpora komunity	Mladý projekt, malá komunita	Dlouholetý projekt s velkou základnou uživatelů	Do změny politiky velmi těžil z komunity nagiosu	Dlouhodobý projekt, komunita ovšem není tak činorodá a velká jako u Nagiosu	Dlouholetý projekt, velká komunita
Bonusy	Mobilní aplikace		Mobilní aplikace, podpora běhu v Cloudu		Mobilní aplikace

Všichni vybraní kandidáti splnili základní požadavky nastavené firmou Arit s.r.o. a tabulka (Tabulka 2) dokazuje jejich vyrovnanost v nabízených možnostech. Mezi kandidáty si můžeme povšimnout časté vazby na Nagios, který jako první přišel s možností rozšíření monitorovacího systému pomocí pluginů, které si navíc můžou uživatelé sami naprogramovat. Mnoho produktů ze současné nabídky proto využívá velké oblíbenosti a rozmanitosti pluginů a implementuje jejich podporu.

Mezi kandidáty byly zařazeny převážně dlouholeté projekty, výjimku tvoří mladý nadějný projekt Icinga, který sice vznikl jako fork Nagios, ale úplně přepracoval vnitřní strukturu systému. Implementuje plnou podporu Nagios pluginů a dokonce nabízí i nástroje pro snadnou migraci právě z Nagios systému.

Projekt Nagios, nejstarší z kandidátů, nabízí širokou škálu uživatelských pluginů, které jsou shromažďovány na serveru Nagios Exchange a jsou tak snadno dohádatelné. Vytvořit skript si může snadno každý uživatel v řadě programovacích a skriptovacích jazyků. Systém je oblíbený právě pro svoji pružnost, modifikovatelnost a možností využít nadstaveb, které potlačí jeden z hlavních nedostatek Nagios, kterým je špatné zpracování výstupů dat a nedostatečné webové konfigurační rozhraní. Nespornou výhodou ovšem je velká, silná a činorodá komunita.

OpsView, nadstavba nad jádrem Nagios, která přináší moderní a komplexnější webové rozhrání, s lepšími výstupy včetně přehledný grafů. V kombinaci s Nagios Core vzniká velmi silný nástroj pro monitoring, jehož nejslabší stránkou jsou, v porovnání s konkurencí, vyšší režijní náklady na provoz v síti.

Výjimku v kandidátech tvoří i systém Zabbix, který jako jediný z nich nevyužívá žádné z vlastností Nagios systému, je od začátku vyvíjen samostatně a uživatelům poskytuje API rozhraní pro snadnější tvorbu vlastních skriptů. Výhodou pro nasazení Zabbixu ve velkých sítích může být i podpora komerčních řešení databázové části systému. Bohužel projekt nemá žádnou snadno dostupnou centralizovanou správu rozšíření jako je Nagios Exchange a uživatelská rozšíření jsou rozeseta na diskusních fórech a případně osobních stránkách autorů. Během dlouholetého vývoje projektu vyšlo několik knižních publikací a výrazně se rozrostla komunita uživatelů.

Jádro monitorovacího systému Zenoss je sice vyvíjeno jako samostatný produkt, ale do systému byla implementována podpora pluginů systému Nagios. K dispozici je uživatelům API, které umožňují vytvářet vlastní pluginy v jazyce Python. Podporované databázové řešení je pouze MySQL a systém se tedy nijak výrazně neodlišuje od ostatních kandidátů.

Po otestování všech systémů a výše uvedeném vyhodnocení jejich vlastností jsem se rozhodl doporučit firmě Arit s.r.o. dva kandidáty. Prvním doporučeným kandidátem je systém Nagios s nadstavbou OpsView. Důvodem k doporučení byla právě modifikovatelnost systému, podpora více jazyků pro tvorbu pluginů, centralizované úložiště pluginů a nadstavbou vylepšené zpracování výstupů. Druhým doporučeným kandidátem byl systém Icinga, který mě zaujal novou vnitřní strukturou, která je podle

mého názoru lépe řešená oproti systému Nagios a poskytuje systému zatím nevyužitý potenciál. Zaujal mě také nástroj pro snadnou migraci z Nagios systému a podpora i pro PostreSQL databázi. Bohužel komunita a dokumentace projektu je zatím velmi slabá a pro řešení složitějších problémů se mi zdá nedostatečná.

Firma Arit s.r.o. se na základě výsledků hodnocení a mých doporučení rozhodla pro systém Nagios s nadstavbou OpsView. Projekt Icinga se vedení firmy také líbil a rozhodli se jeho vývoj nadále sledovat s možností budoucí migrace systému ze systému Nagios.

2.6 Instalace a konfigurace dohledového systému a agentů

Vedení firmy Arit s.r.o. se rozhodlo pro instalaci dohledového systému do virtuálního prostřední s čistou instalací OS. Pro instalaci byl vyčleněn server s parametry:

- CPU 2x jádro 2,5Ghz,
- RAM 3GB,
- systémové HDD 50GB,
- databázové HDD 100GB,
- OS CentOS 5.6 x64.

Před instalací samotného OpsView je důležité nainstalovat nebo nakonfigurovat tzv. předpokládané části systému. U OpsView se jedná o instalaci MySQL, vypnutí security enhanced Linux, vytvoření skupin nagios a nagcmd a uživatele nagios spadajícího do těchto skupin. Další software se nainstaluje a nakonfiguruje během instalace OpsView a to včetně Nagios Core. (33)

2.6.1 Instalace serveru

Pro instalaci serveru je na webu OpsView podrobný návod (34), který nás provede instalací krok za krokem. Není účelem této práce tento návod přepisovat nebo kopírovat, a proto popis instalace bude zestručněn tak, aby pouze přiblížil postup a nutné kroky k nainstalování dohledového systému.

Instalaci OpsView sytému můžeme provést stažením balíčků přímo ze serveru projektu a lokální instalací nebo můžeme využít některého z rozšířených repozitářů systému. Během instalace systém stáhne a nainstaluje všechny důležité součásti včetně Nagios Core.

Po dokončení instalace je nutné systém dokonfigurovat a připravit k používání. V konfiguračních souborech je nutné nastavit uživatele a heslo pro přístup k databázi a systému. Pro další konfiguraci jsou připraveny skripty, které nakonfigurují ostatní součásti dohledového systému a vytvoří databázovou strukturu. Na závěr je důležité vygenerovat konfigurační soubor pro Nagios Core, ověřit spuštění MySQL. Pokud na serveru využíváme firewall musíme upravit jeho nastavení tak, aby byl server dostupný z požadovaných sítí. Server OpsView naslouchá na portu 3000. Po splnění všech kroků

návodu můžeme server spustit. Webové konfigurační rozhraní najdeme rovněž na portu 3000 na příslušném serveru a pro přístup využijeme defaultní účet: admin, heslo: initial. Po přihlášení do webového rozhraní okamžitě změníme heslo.

2.6.2 Instalace agenta na systému Linux

Následující postup pro instalaci a úvodní konfiguraci agenta na monitorovaných serverech se systémem Linux je již výsledným postupem, který vznikal v průběhu implementace dohledového systému pro firmu Arit s.r.o. Postup je vytvořen pro distribuci CentoOS a při konfiguraci na jiných distribucí se mohou příkazy mírně lišit. Před jakýmikoli úpravami konfiguračních souborů platí pravidlo zálohovat původní soubor.

Agenta a základní rozšíření nainstalujeme příkazem:

```
yum -y install nagios-plugins-* nrpe
```

Pro stažení a rozbalení souboru s dalšími rozšířeními, které nejsou obsaženy ve standardním balíku a rozšířeními které jsou vytvořeny firmou Arit s.r.o., použijeme následující příkazy:

```
wget -P /usr/lib64/nagios/plugins/ \
mail.arit.cz/downloads/nagios/plugins_arit.tar

cd /usr/lib64/nagios/plugins/

tar xf plugins_arit.tar
```

Některé z pluginů spouštějí software nebo služby systému vyžadující oprávnění root. Pro správnou funkčnost těchto pluginů je nutné přidat uživatele nrpe[2] do souboru sudoers. Pro úpravu sudoers použijeme příkaz:

```
visudo
```

Nyní provedeme následující změny:

```
#Defaults requiretty
nrpe  ALL=(ALL)   NOPASSWD:/usr/lib64/nagios/plugins/
```

Jedna z důležitých monitorovaných vlastností je informace o datu a čase restartu serveru. Pro zjištění této informace se využívá jednoduchý skript, který čte data ze souboru vytvořeného při startu systému. Pro vytvoření tohoto souboru postupujeme následovně:

```
echo 'echo "Cas startu `date`" >> /var/log/restart.log' >>/etc/rc.local
```

Pro nastavení komunikace agenta se serverem, úrovně logů a dalších parametrů je nutné editovat soubor /etc/nagios/nrpe.conf. Direktivou Allow_hosts[3] nastavujeme IP adresy serverů, které můžou spouštět agenta. Povolení spouštění agenta se vzdálenými parametry

[2] Uživatel nrpe se vytvoří automaticky při instalaci a spouští všechny operace vykonávané agentem.
[3] Více adres oddělujeme čárkou. Pro testování se využívá adresa localhost 127.0.0.1, který umožní spouštět agenta lokálně.

provedeme nastavením direktivy Dont_blame_nrpe[4] a direktivou Debug nastavujeme logování chybových výstupů. Další důležitou direktivou je server_port, která nám udává, na jakém portu agent komunikuje se serverem. Tento port je nutné povolit ve firewallu monitorovaného serveru nebo stanice pro komunikaci s dohledovým serverem. Na konci souboru je prostor pro nastavení jednotlivých příkazů, které má agent možnost spouštět, ale doporučuje se tento seznam vést v externím souboru a ten do konfiguračního připojit pomocí direktivy include. Konfigurační soubor je komentovaný a obsahuje samozřejmě více direktiv. Nyní si uvedeme konkrétní změny v nastavení agenta na monitorovém serveru, celý konfigurační souboru se nachází v příloze A.

```
Allowed_hosts=217.11.226.6
Dont_blame_nrpe=1
Debug=1

include=/etc/nrpe.d/command-arit.cfg
```

Úpravy nastavení firewallu provedeme v souboru /etc/shorewall/params přidáním řádku:

```
Arit=217.11.226.6
```

A v souboru /etc/shorewall/rules přidáním řádku:

```
ACCEPT          net:$Arit       fw          tcp         5666
```

Zkontrolujeme syntaxi nastavení firewallu a restartujeme:

```
shorewall check
shorewall restart
```

Nyní je vše nakonfigurované a zbývá službu pouze spustit. Abychom nemuseli službu spouštět ručně po každém restartu počítače, přidáme ji do automaticky spouštěných služeb.

```
service nrpe start
chkconfig --level 3 nrpe on
```

Následující nastavení příkazů je ukázkou ze souboru command-arit.cfg, celý soubor se nachází v příloze C. Příkazy jsou uvozeny slovem command a následuje, v rámci soubor u, unikátní název v hranatých závorkách, za rovnítkem se uvádí cesta k pluginu a argumenty. První příkaz je bez argumentů a spouští plugin pro kontrolu RAID pole. Druhý příkaz slouží pro kontrolu místa na disku a má tzv. vzdálené parametry, které se nastavují na dohledovém serveru a zde jsou zástupnými proměnnými předávány skriptu. Třetí příkaz zjišťuje obsazenost oddílu swap a používá lokální parametry, které mají hodnoty uvedeny v konfiguračním souboru. Poslední ukázkový příkaz zjišťuje počet realokovaných sektorů na discích. Volá se s prefixem sudo, protože plugin volá program smarctl, který vyžaduje root opravnění.

```
command[check_raid]=/usr/lib64/nagios/plugins/check_linux_raid
command[check_home]=/usr/lib/nagios/plugins/check_disk -w $ARG1$ -c
$ARG2$ -p $ARG3$
```

[4] Defaultní nastavení umožňuje spouštět vzdáleně příkazy pouze s lokálními parametry.

```
command[check_swap]=/usr/lib64/nagios/plugins/check_swap -w 75% -c 50%
command[check_smart_reallocated_sector]=sudo
/usr/lib64/nagios/plugins/check_smart_reallocated_sector -d /dev/sda
/dev/sdb -w 1 -c 10
```

2.6.3 Instalace agenta na systému Windows

Pro monitoring Windows serverů nebo stanic se využívá software nrpe-nt. Pro potřeby firmy Arit s.r.o. byly vytvořeny skripty pro snadnější instalaci a konfiguraci. Poslední stabilní verze i s upravenými skripty se nachází na firemním serveru. Při stažení softwaru z domácí stránky je potřeba doplnit balík pluginů, například Nagios Plugin Collection. Po stažení archivu soubor rozbalíme do adresáře C:\Program Files\nrpe-nt. Nainstalování provedeme spuštěním nrpe_nt.exe nebo skriptu install.bat. Po instalaci přistoupíme ke konfiguraci. Úpravy provádíme v souboru nrpe.cfg, který obsahuje všechny direktivy jako konfigurační soubor na systému Linux. Oproti Linuxu obsahuje soubor ještě direktivu loglevel pro nastavení úrovně logování událostí. Soubor je opět komentovaný a celý se nachází v příloze B. V nastavení firewallu musíme povolit port, na kterém komunikuje agent s dohledovým serverem. Ukázka změn v konfiguračním souboru, které jsou nutné pro fungování agenta v monitorované síti firmy Arit s.r.o., ostatní nastavení je ponecháno:

```
allowed_hosts=217.11.226.6
dont_blame_nrpe=1
debug=1
include=C:\Program Files\nrpe-nt\command-arit.cfg
loglevel=4
```

Soubor command-arit.cfg stejně jako na systému Linux obsahuje příkazy, které může agent spouštět. První ukázkový příkaz je bez parametrů a slouží pro zjištění uptime serveru. Druhý příkaz s lokálními parametry slouží pro zjištění využití paměti RAM. Celý ukázkový soubor command-arit.cfg pro Windows nalezne v příloze D.

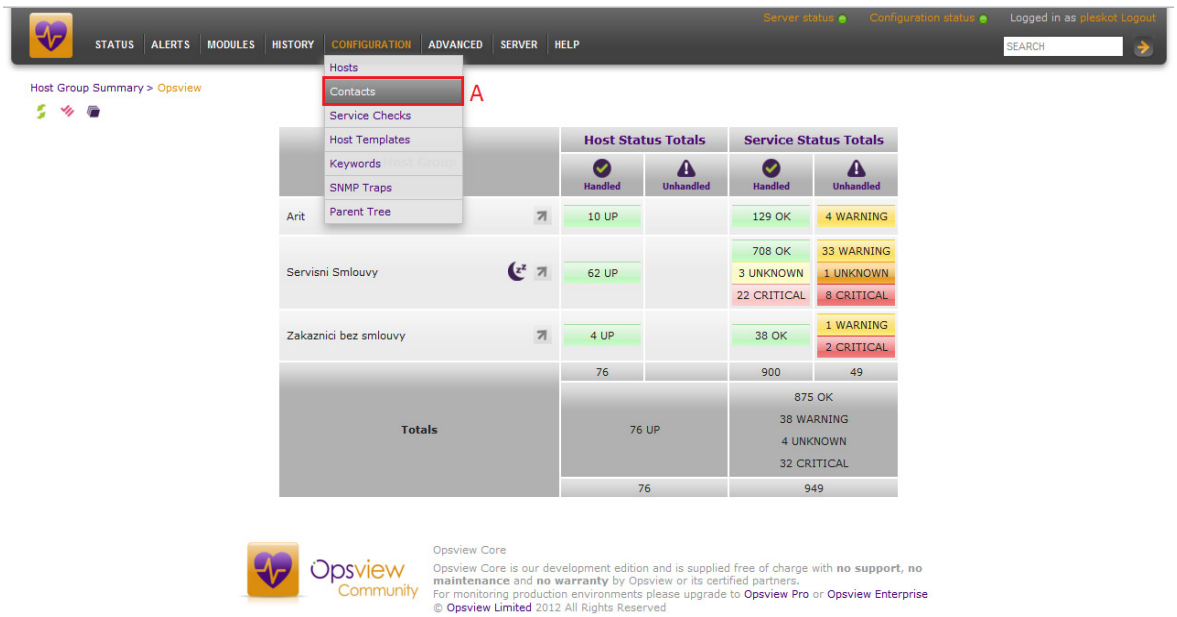
```
command[nt_uptime]=C:\Program Files\nrpe-nt\bin\check_uptime.exe
command[nt_memory]=C:\Program Files\nrpe-nt\bin\memload_nrpe_nt.exe 80 90
```

2.6.4 Webové konfigurační rozhraní

Po instalaci a konfiguraci agentů na monitorovaných serverech můžeme nastavit dohledový systém pro jejich monitoring. Veškerá konfigurace probíhá přes webové rozhraní, které standardně nalezneme na http://ip_monitoring_serveru:3000. Rozhraní je intuitivní, ale velmi rozsáhlé a nabízí mnoho možností ke konfiguraci. Na stránkách projektu OpsView nalezneme ve wikipedii projektu obsáhlou dokumentaci včetně popisu webového rozhraní. Proto je popis rozhraní v této práci omezen pouze na elementární nastavení funkcí pro zprovoznění dohledu.

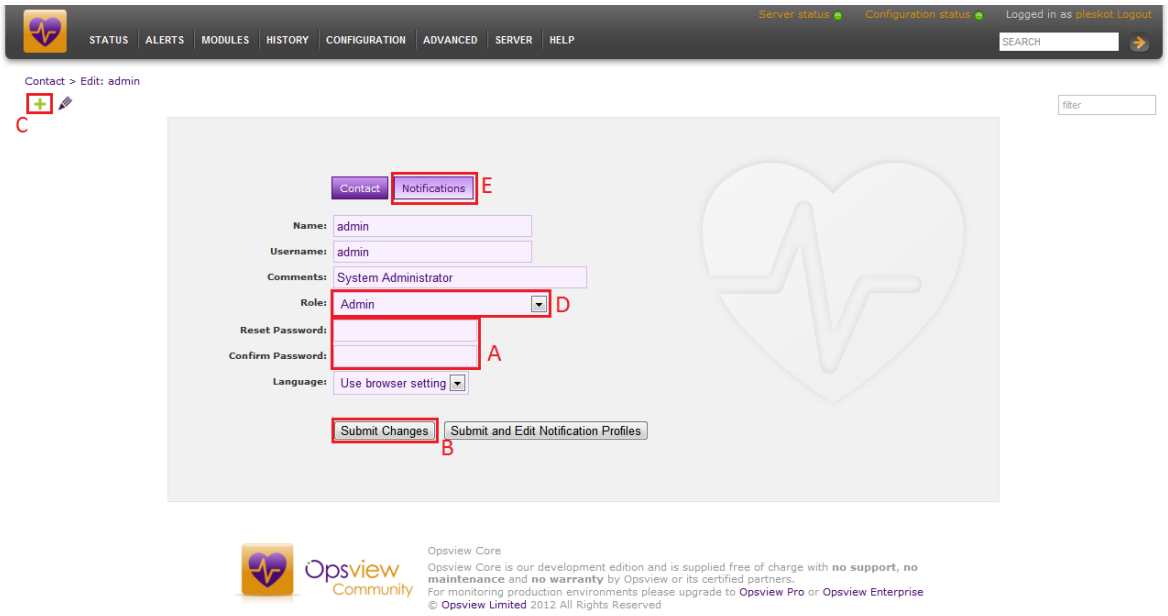
Obrázek 3 - Přihlašovací obrazovka *zdroj:vlastní*

Po přihlášení do systému vidíme tabulku s přehledem monitorovaných zařízení. Pokud jsme se přihlásili poprvé, uvidíme pouze dohledový server, který se automaticky přidá do monitoringu během instalaci serveru. Při prvním přihlášení do systému s defaultními údaji nejprve změníme heslo uživatele admin. Správu uživatelů nalezneme v menu Configuration - Contacs , pozice A na obrázku (Obrázek 4)



Obrázek 4 - Přehled monitorovaných zařízení, konfigurace účtů *zdroj:vlastní*

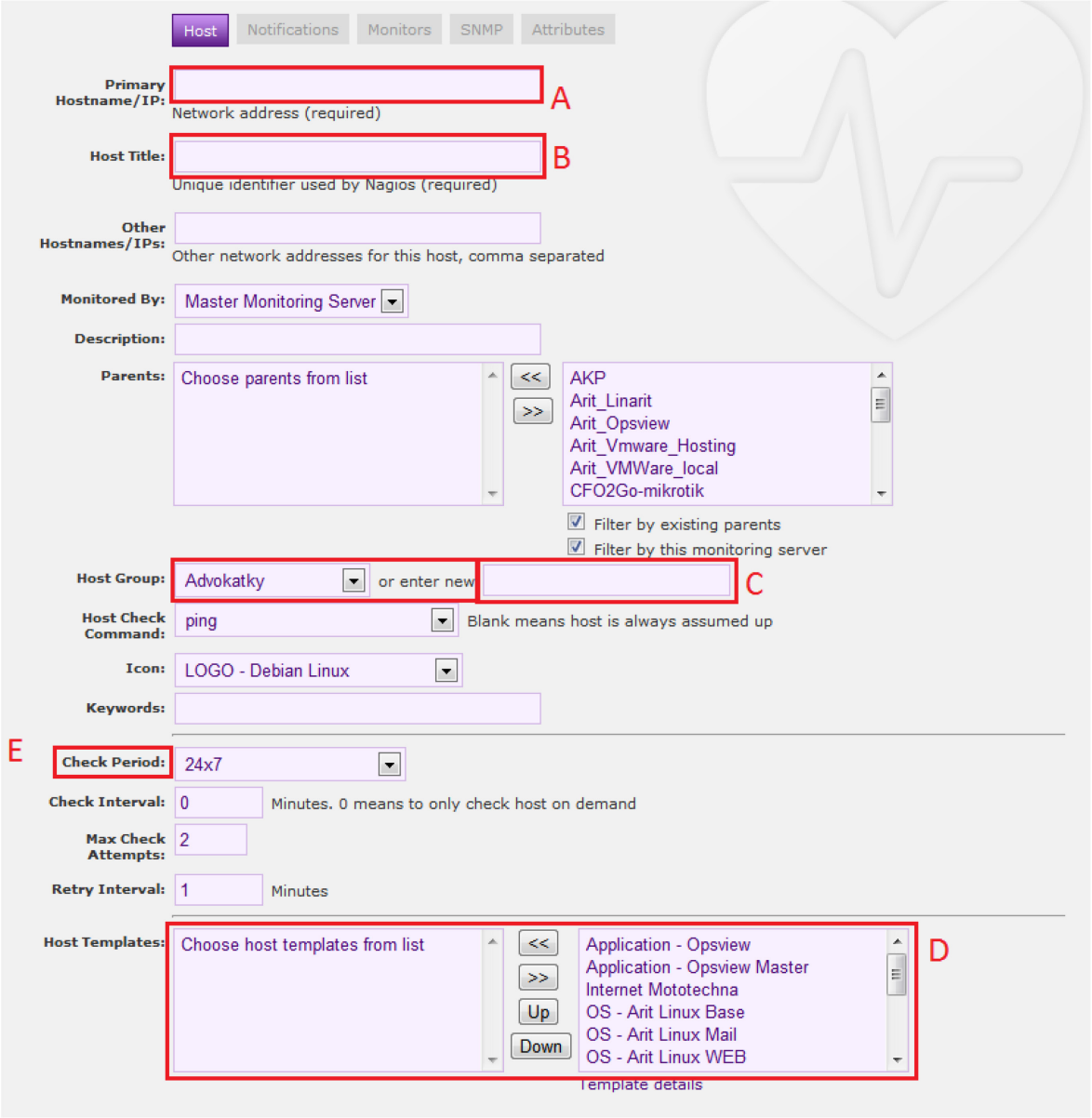
Pro změnu hesla vybereme požadovaného uživatele a ve formuláři (Obrázek 5) vyplníme nové heslo na pozici A. Změny potvrdíme tlačítkem pod formulářem na pozici B. Formulář pro přidání nového uživatele nám zpřístupní tlačítko na pozici C.



Obrázek 5 - Formulář pro editaci nebo vytvoření nového uživatele ***zdroj:vlastní***

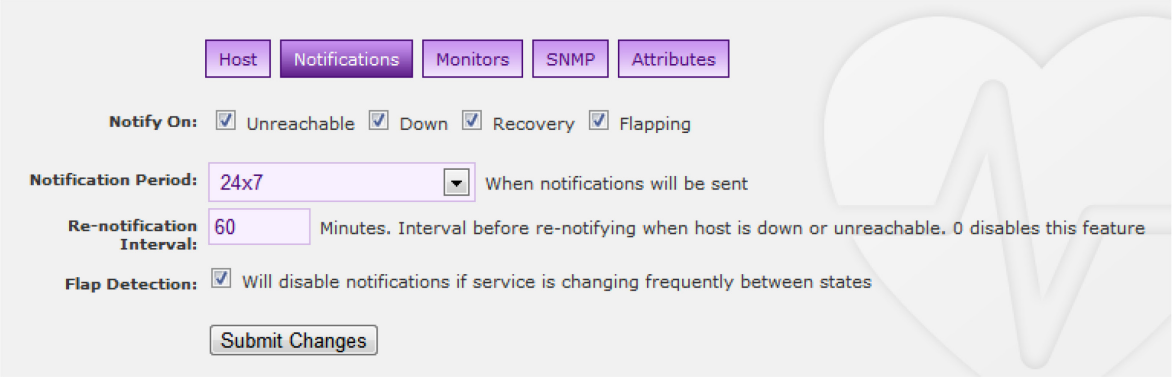
Přidání nového uživatele probíhá přes stejný formulář. Uživatelům můžeme přidělovat různá oprávnění pomocí volby Roles, pozice D. V kartě Notifications, pozice E, můžeme nastavit emailový kontakt, na který budou zasílány emaily ze systému.

Pro přidání nového monitorovaného zařízení použijeme menu Configuration – Hosts a opět tlačítko + v levém horním rohu a vyplníme formulář (Obrázek 6).



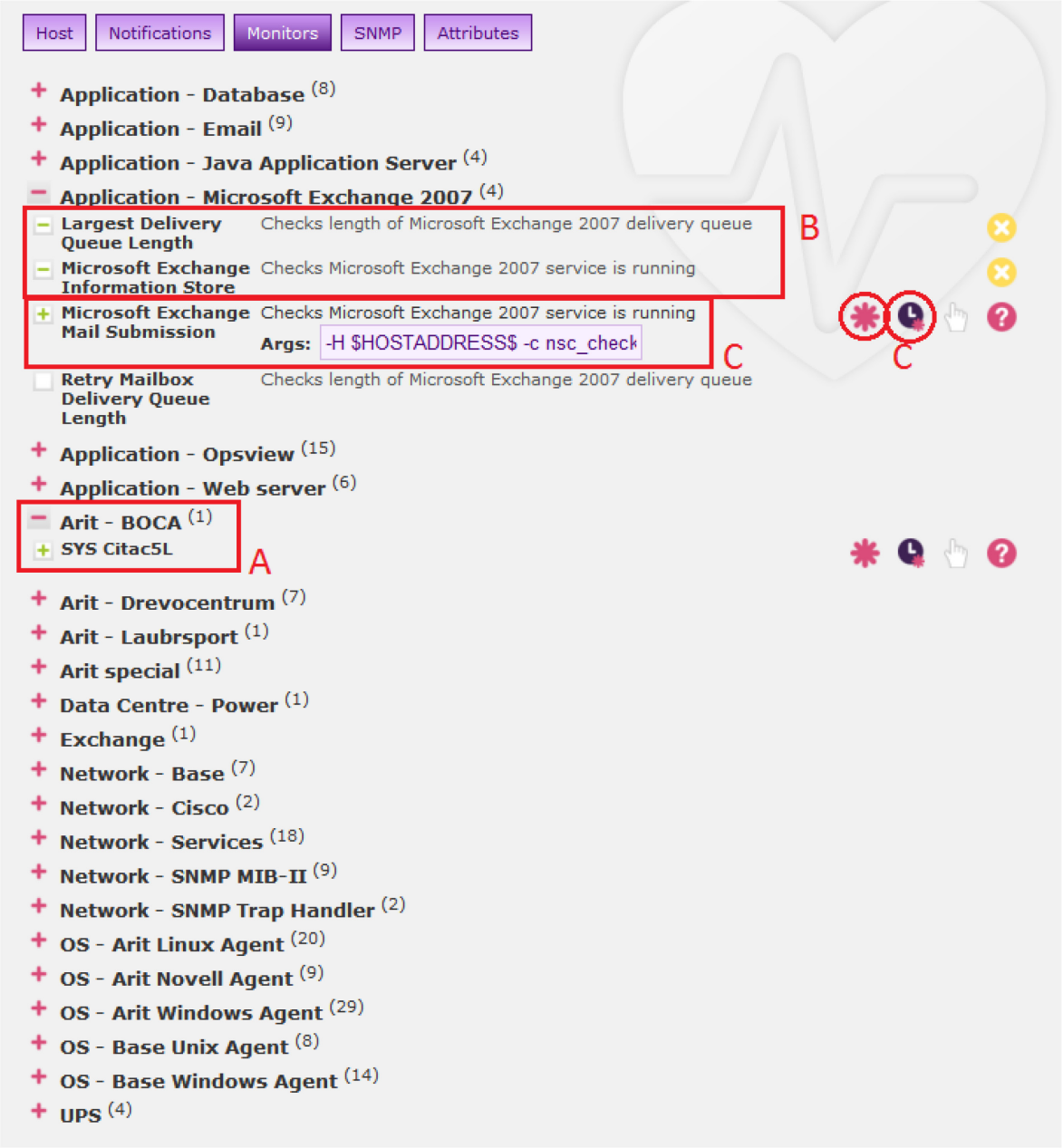
Obrázek 6 - Formulář pro přidání nového monitorovaného zařízení ***zdroj:vlastní***

Na pozici A vyplníme veřejnou IP adresu nebo doménový název monitorovaného serveru. Do pole B uvedeme v rámci dohledového systému unikátní označení monitorovaného serveru. Pro přiřazení serveru do skupiny použijeme menu nebo pole C. Využití skupin nám usnadňuje orientaci v systému a umožňuje například zobrazit servery pouze jednoho konkrétního zákazníka. Pomocí seznamů D přiřadíme serveru některé z připravených monitorovacích vzorů. Vzory si můžeme vytvářet i vlastní a usnadňují nám konfiguraci monitorovaných parametrů a služeb. Ve formuláři můžeme dále nastavit zobrazované logo u serveru, přiřadit klíčová slova, přidat popisek serveru nebo nastavit jak často a jakým způsobem se bude provádět kontrola dostupnosti serveru. Dokumentace k jednotlivým polím je dostupná online po kliknutí na název nastavovaného pole E. Na následující kartě můžeme nastavit vlastnosti upozorňování na stav serveru. (Obrázek 7)



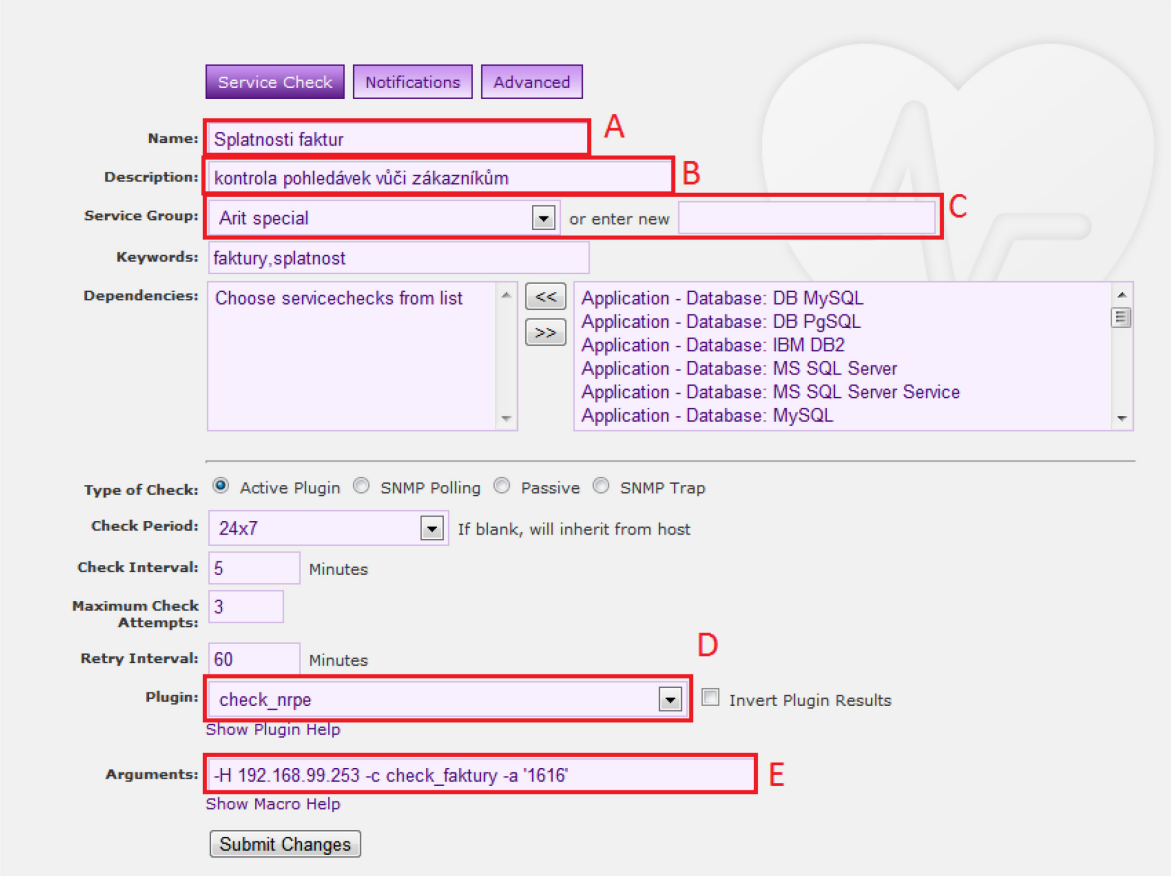
Obrázek 7 - Nastavení upozornění na stav zařízení ***zdroj:vlastní***

Karta Monitors (Obrázek 8) slouží k nastavení monitorovaných parametrů mimo přiřazený vzor. Můžeme tak přidat, pozice A, nebo odebrat, pozice B, některé monitorované parametry oproti vzorovému nastavení. Příkazy k vykonání můžeme také volat s jinými argumenty C.



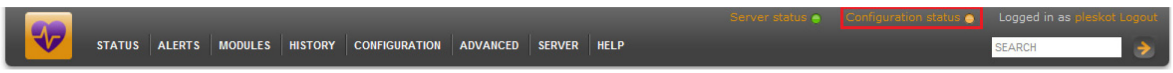
Obrázek 8 - Nastavení monitorovaných vlastností nad rámec skupiny ***zdroj:vlastní***

Pokud chceme monitorovat parametr, který se nenachází v defaultním výběru, musíme ho nejdříve vytvořit. Pro vytvoření nového monitorovaného parametru využijeme menu Configuration – Service Checks a tlačítko + v levém horním rohu Ve formuláři (Obrázek 9) na pozici A vyplníme v rámci systému unikátní název, do pole B vyplníme popis. Pozice C nám umožní náš check zařadit do některé ze skupin nebo vytvořit skupinu novou. V seznamu D vybereme plugin, který bude náš dotaz vykonávat. V případě, že nám nevyhovuje žádný z nabízených pluginů, využije plugin check_nrpe, který je určen pro spouštění uživatelských skriptů a pluginů. Následně do pole E vyplníme argumenty pro spuštění našeho skriptu. Argument -c uvozuje název příkazu v konfiguračním souboru command-arit.cfg na monitorovaném serveru. Změny uložíme.

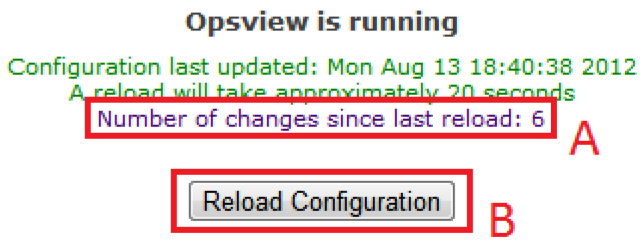


Obrázek 9 - Formulář pro vytvoření nové monitorované vlastnosti *zdroj:vlastní*

Všechny námi provedené změny ve webovém rozhraní jsme sice uložili, ale aby se promítli do provozu dohledového serveru, je nutné provést znovunačtení konfigurace. O tom, že proběhly nějaké změny, které čekají na znovunačtení, nás informuje ikona v záhlaví stránky (Obrázek 10). Po kliknutí na informativní ikonu nám systém nabídne provedení znovunačtení (Obrázek 11) a předkládá k zobrazení seznam provedených změn A. Po potvrzení akce B, nás informuje o čase dokončení znovunačtení konfigurace a zobrazí odpočet do tohoto času.



Obrázek 10 - Upozornění na uložené, ale nenačtené změny *zdroj:vlastní*



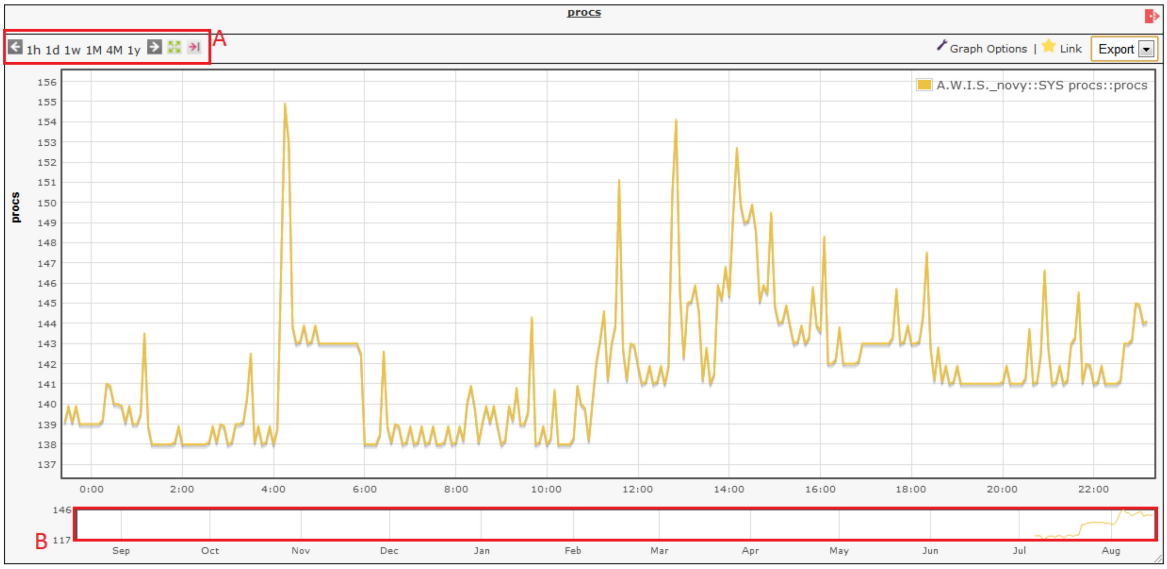
Obrázek 11 - Znovunačtení konfigurace *zdroj:vlastní*

Pro zobrazení informací o monitorovaných serverech vybereme v úvodním přehledu příslušnou skupinu, zákazníka a server. Na obrázku (Obrázek 12) vidíme přehled všech monitorovaných údajů pro jeden server. Po kliknutí na ikonu A se nám zobrazí interaktivní graf (Obrázek 13), který nám umožní vyhledat historické záznamy o průběhu chování sledovaného parametru. Kliknutím na název B se nám zobrazí detail příkazu a jeho nastavení. V grafu (obr) se můžeme pohybovat pomocní záhlaví A nebo vyznačením úseku na časové ose B. Hodnoty ze zobrazeného grafu můžeme exportovat do souboru. Detailní pohled (Obrázek 14) nám zobrazí tabulku A s informacemi o stavu příkazu, jeho výsledcích a času posledního a následujícího spuštění. Tabulka B poskytuje nástroje pro správu příkazu, jeho dočasné vypnutí nebo naplánování odstávky. K jednotlivým příkazům je možné psát na pozici C komentáře, aby ostatní zaměstnanci věděli o plánovaných odstávkách nebo problémech s monitorovanou vlastností.

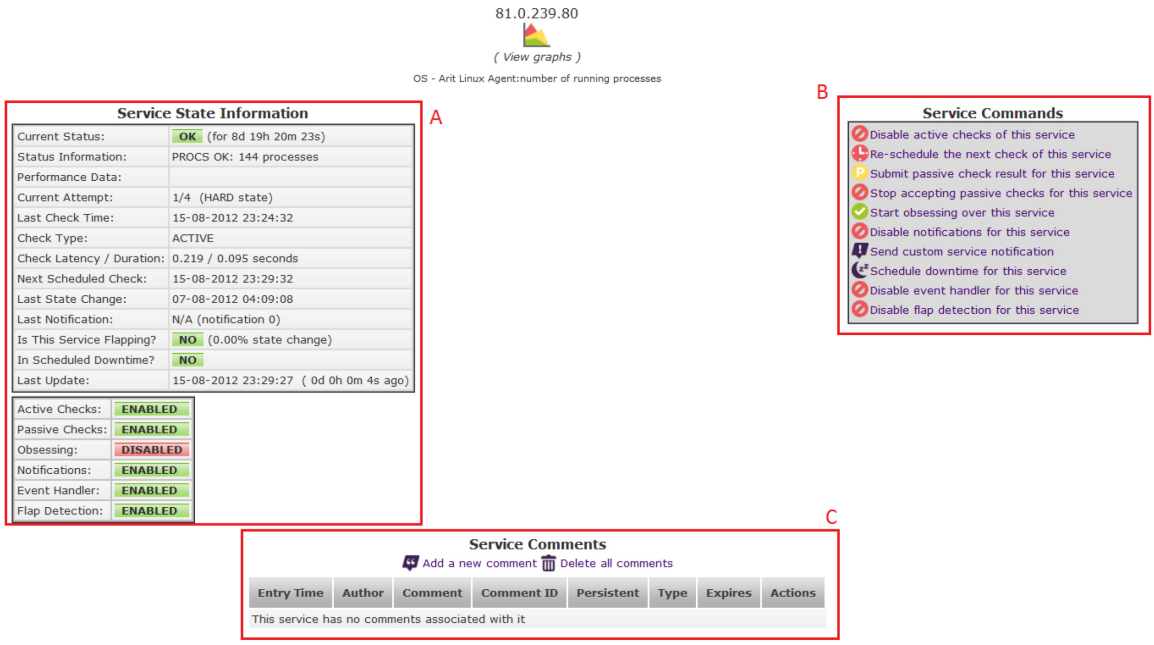
Host	Service	Status	Last Check	#	Status Information
A.W.I.S._novy	FS /	WARNING 10h 22m 36s	2012-08-15 23:11:23	4/4	DISK WARNING - free space: / 23638 MB (18% inode=87%):
	HDD bsec	UNKNOWN 8d 19h 7m 9s	2012-08-15 23:11:03	4/4	SMART_CHECK ERROR: Nelze ziskat potrebne informace
	HDD temp	OK	2012-08-15 23:13:04	1/4	SMART_CHECK OK: /dev/sda teplota: 29 Â°C: /dev/sdb teplota: 30 Â°C:
	MAIL fronta	OK	2012-08-15 23:14:23	1/4	OK: mailq is empty
	NET ping	OK	2012-08-15 23:15:00	1/4	OK - 81.0.239.80: rta 4,290ms, lost 0%
	Splatnost faktur	OK	2012-08-15 22:59:09	1/4	OK: Vsechny pohledavky radne zaplaceny !
	SSH 1300	OK	2012-08-15 23:15:03	1/4	TCP OK - 0,011 second response time on port 1300
	SYS CpuLoad	OK	2012-08-15 23:14:18	1/4	OK - load average: 0.00, 0.00, 0.00
	SYS procs	OK	2012-08-15 23:14:32	1/4	PROCS OK: 145 processes
	SYS restart	OK	2012-08-15 23:14:20	1/4	OK
	SYS swap	OK	2012-08-15 23:14:20	1/4	SWAP OK - 96% free (1916 MB out of 2015 MB)
	SYS users	OK	2012-08-15 23:14:33	1/4	USERS OK - 1 users currently logged in
	WEB http	WARNING 1d 11h 32m 5s	2012-08-15 23:13:53	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 546 bytes in 0,010 second response time
	WEB https	WARNING 8d 19h 6m 58s	2012-08-15 23:14:33	4/4	HTTP WARNING: HTTP/1.1 403 Forbidden - 5221 bytes in 0,042 second response time
Totals	14	10 OK 3 WARNING 1 UNKNOWN			

B A

Obrázek 12 - Přehled monitorovaných vlastností serveru ***zdroj:vlastní***



Obrázek 13 - Grafická prezentace získaných hodnot ***zdroj:vlastní***



Obrázek 14 - Podrobnosti o stavu příkazu ***zdroj:vlastní***

Tento stručný návod obsahuje popis nastavení všech základních důležitých částí pro spuštění monitorování serveru. Podrobnější postupy a pokročilá nastavení jsou popsána v dokumentaci dostupné i přes menu Help – Documentation[5].

2.7 Tvorba skriptů pro monitoring vlastních parametrů

Jednou z výhod využití dohledového systému s Nagios Core je podpora uživatelských pluginů. Pro jejich tvorbu je podporováno mnoho programovacích a skriptovacích jazyků.

[5] http://docs.opsview.com/doku.php?id=opsview-core

Pro správné rozpoznání stavu monitorované vlastnosti musí uživatelský skript vracet jednu z předurčených návratových hodnot:

- 0 = OK,
- 1 = WARNING,
- 2 = CRITICAL,
- 3 = UNKNOWN.

Žádné další náležitosti systém nevyžaduje, a tak mají autoři pluginů při tvorbě velmi volnou ruku. Pro firmu Arit s.r.o. jsem vytvořil v rámci implementace správy dohledového systémů ve spolupráci s dalšími zaměstnanci několik pluginů. Několik pluginů bylo implementováno i ze serveru Exchange Nagios. V přílohách E, F a G naleznete celé zdrojové kódy uživatelských pluginů. Všechny pluginy jsou jednoduché skripty napsány pro Bash. Pro zjištění informací využívají většinou nástrojů operačního systému a jeho výstupy následně přebírají, analyzují a na základě výsledků odesílají hlášení pro dohledový systém formou návratové hodny a textového výstupu.

3 Vyhodnocení

Po zavedení dohledového systému a jeho začlenění do spravované sítě se dostavili první výsledky již po měsíci provozu, kdy data nasbíraná systémem byli již dostatečně informativní pro plánování servisních návštěv zákazníků. V současné době data získaná dohledovým systémem slouží vedení firmy k sestavení servisních plánů, které se mění přibližně jednou za měsíc podle aktuálního stavu a počtu servisních zákazníků a jsou aplikovány ve 2 týdenních cyklech. Aktuální monitorovaná data jsou zobrazena také na velké obrazovce v prostoru kanceláří servisních techniků a zvyšují tak přehled zaměstnanců.

Snížení počtu mimořádných výjezdů servisních techniků se projevilo až v delším časovém období., především proto, že někteří servisní technici po zavedení systému málo využívali systému varování. Proto firma přistoupila k zakoupení moderních chytrých telefonů s datovými tarify, což umožnilo technikům příjem emailů 24 hodin denně a možnost přístupu na monitorovací systém bez nutnosti vazby na počítač. Ve druhém čtvrtletí využívání dohledového systému se dostavili i očekávané výsledky, které demonstruje následující tabulka (Tabulka 3) V tabulce (Tabulka 3) vidíme vývoj mimořádných výjezdu během fungování dohledového systému u několika vybraných zákazníků. První sloupec uvádí sledovaného zákazníka. Ve druhém je uveden průměrný počet mimořádných servisních výjezdů za měsíc před zavedením dohledového systému. Ve třetím sloupci je uveden průměrný počet mimořádných výjezdu za měsíc v období prvních třech měsíců fungování dohledového systému a ve čtvrtém sloupci hodnota uvádí průměrný počet mimořádných servisních výjezdů za měsíc ve druhém čtvrtletí fungování dohledového systému.

Tabulka 3 - Vývoj mimořádných servisních zásahu za měsíc v průběhu využívání dohledového systémů

Název zákazníka	Před zavedením dohledového systému	První čtvrtletí po zavedení dohledového systému	Druhé čtvrtletí po zavedení dohledového systému
Milan Škoda - Fotocentrum	10	9	5
Poll s.r.o.	6	6	2
CITYPLAN spol. s r.o.	8	6	3
CK Sport-S	5	4	2

Závěr

Teoretická část práce čtenářům podrobně představuje současné možnosti pro management a monitoring počítačových sítí a to především pomocí open source produktů. Vysvětleny jsou i principy a postupy sloužící k monitorování počítačových sítí. Praktická část se věnuje nasazení vybraného monitorovacího systému v reálném firemním prostředí firmy Arit s.r.o.

Cílem nasazení dohledového systému ve firmě Arit s.r.o. byla především informovanost vedení a servisních techniků o stavu spravovaných serverů. Získávání těchto informací bylo dříve velmi kostrbaté a nedostačující pro kvalitní plánování servisní zásahů a údržby serverů. S rostoucím počtem zákazníků a spravovaných serverů byla tato situace dlouhodobě neudržitelná. Dalším cílem bylo dosažení nižšího počtu mimořádných servisních zásahů a snížení downtime spravovaných serverů. K snížení těchto hodnot měl dohledový systém dopomoci aktuálními informacemi a systémem varování servisních techniků.

Nasazení dohledového systému ve firmě Arit s.r.o., dle dosažených výsledků shrnutých v kapitole 3 – Vyhodnocení, splnilo všechna očekávání a dohledový systém se stal součástí každodenního chodu firmy. Aktivně slouží především servisním technikům a pomáhá jim analyzovat technické problémy, předvídat události a včasně varovat před možnými problémy. Vedení firmy získalo nástroj pro plánování servisních návštěv a údržby serverů. Velkým přínosem byla i instalace velké obrazovky do prostoru kanceláří, která po celý den zobrazuje aktuální monitorovaná data, a tak zajišťuje informovanost všech přítomných zaměstnanců. K dosažení všech cílů dopomohl i nákup moderních chytrých telefonů umožňujících příjem emailů a přístup pohodlný přístup na internet. Implementace dohledového systému tedy byla pro firmu Arit s.r.o. přínosná.

Na začátku této bakalářské práce byla pouze moje osobní zvědavost, jak se monitorují počítačové sítě, když nemáte pouze jeden domácí server a několik notebooků. Vypracováním jsem tedy nabyl nejen teoretické znalosti o monitoringu a managementu počítačových sítí, ale vyzkoušel jsem si i praktické nasazení monitoringu v reálné síti. Vzhledem k velmi malým znalostem problematiky serverů a dohledových systémů jsem v průběhu vypracování prošel několikrát peklem, které mi ve výsledku rozšířilo obzory a zkušenosti. Za dobu 10 měsíců jsem vyzkoušel prakticky všechny možnosti implementovaného systému v reálném prostředí. V závěru implementace jsem byl i začleněn do využití výstupů dohledového serveru při plánování servisních návštěv.

V průběhu psaní této práce změnil projekt OpsView své obchodní strategie a ukončil vývoj verze OpsView Community Edition, která byla v rámci práce využita. OpsView nyní nabízí produkt OpsView Core, který již není nadstavbou Nagios Core, ale samostatným řešením. Tyto informace jsem zpětně zmínil i u popisu produktu OpsView. Na základě těchto změn se vedení firmy Arit s.r.o. rozhodlo migrovat na jiný monitorovací systém a oslovilo mě s nabídkou spolupráce na migraci a nasazení nového systému.

Literatura

1. **Bouška, Petr.** Počítačové sítě - computer networks. *www.samuraj-cz.com*. [Online] 30. září 2007. [Citace: 18. březen 2012.] http://www.samuraj-cz.com/clanek/pocitacove-site-computer-networks/.

2. —. Zařízení v síti pod kontrolou. *http://www.samuraj-cz.com*. [Online] 21. září 2009. http://www.samuraj-cz.com/clanek/zarizeni-v-siti-pod-kontrolou.

3. **Pick, Michael.** Dohledové systémy zabezpečující firemní sítě. *Svět sítí*. [Online] 11. srpen 2008. [Citace: 5. únor 2012.] http://www.svetsiti.cz/clanek.asp?cid=Dohledove-systemy-zabezpecujici-firemni-site-1182008.

4. **Bouška, Petr.** Začínáme s monitoringem sítě. *http://www.samuraj-cz.com*. [Online] 1. září 2009. [Citace: 5. únor 2012.] http://www.samuraj-cz.com/clanek/zaciname-s-monitoringem-site/.

5. **Odvárka, Petr.** Význam monitoringu sítí. *www.svetsiti.cz*. [Online] 6. prosinec 2010. [Citace: 5. únor 2012.] http://www.svetsiti.cz/clanek.asp?cid=Vyznam-monitoringu-siti-6122010.

6. **MonitorTools.com.** Network Monitor Software and Windows Development Tools. *Network Monitor Software and Windows Development Tools*. [Online] 2012. http://www.monitortools.com/.

7. **Cottrell, Les.** Network Monitoring Tools. *Network Monitoring Tools*. [Online] 2012. http://www.slac.stanford.edu/xorg/nmtf/nmtf-tools.html.

8. **ping?, what is.** What is Ping? Ping is the ultimate tool in computer networking. *What is Ping? Ping is the ultimate tool in computer networking*. [Online] 2012. http://www.whatisping.net/.

9. **Štěpán, Jakub.** Sledování sítě. *Sledování sítě*. [Online] 2011. http://www.fi.muni.cz/~kas/p090/referaty/2011-jaro/ut/snmp.html.

10. **Harrington, David, Wijnen, Bert a Presuhn, Randy.** RFC 3411 - An Architecture for Describing Simple Network Management Protocol (SNMP) Management Frameworks. *http://tools.ietf.org*. [Online] prosinec 2002. http://tools.ietf.org/html/rfc3411.

11. **Rockwood, Ben.** Three Flavors of SNMP. *http://www.cuddletech.com*. [Online] 23. listopad 2004. [Citace: 7. únor 2012.] http://www.cuddletech.com/articles/snmp/node4.html.

12. **McCloghrie, Keith a Blumenthal, Uri.** RFC 3826 - The Advanced Encryption Standard (AES) Cipher Algorithm in the SNMP User-based Security Model. *http://tools.ietf.org*. [Online] červen 2004. http://tools.ietf.org/html/rfc3826.

13. **Bouška, Petr.** SNMP - Simple Network Management Protocol. *http://www.samuraj-cz.com*. [Online] 20. prosinec 2006. [Citace: 8. únor 2012.] http://www.samuraj-cz.com/clanek/snmp-simple-network-management-protocol/.

14. **Rockwood, Ben.** The Cuddletech Guide to SNMP Programming. *http://www.cuddletech.com*. [Online] 17. listopad 2004. [Citace: 5. únor 2012.] http://www.cuddletech.com/articles/snmp/.

15. *Multiagent System Implementation for Network*. **M., Néstor D. Duque, a další.** Berlin : Springer-Verlag Berlin Heidelberg, 2009.

16. **Klaška, Luboš.** SNMP objekty a MIB. *http://www.svetsiti.cz*. [Online] 13. červen 2000. [Citace: 17. únor 2012.] http://www.svetsiti.cz/clanek.asp?cid=SNMP-objekty-a-MIB-1362000.

17. **Postel, J. a Reynolds, J.** RFC 1700 - Assigned Numbers. *http://tools.ietf.org*. [Online] říjen 1994. http://tools.ietf.org/html/rfc1700.

18. TCP/IP - navázání a ukončení spojení. *http://www.samuraj-cz.com*. [Online] 13. září 2007. http://www.samuraj-cz.com/clanek/tcpip-navazani-a-ukonceni-spojeni.

19. **Haller, Martin.** Skenování portů: techniky. *lupa.cz*. [Online] 31. říjen 2006. http://www.lupa.cz/clanky/skenovani-portu-techniky.

20. **Häring, David.** Ochrana před scanováním portů: Portsentry. *Linuxové noviny*. [Online] 5. listopad 2000. http://www.linux.cz/noviny/2000-11/clanek10.html.

21. **Čečák, Ondřej.** Linux v příkazech - diagnostika sítě. *linuxsoft.cz*. [Online] 19. srpen 2004. http://www.linuxsoft.cz/article.php?id_article=278.

22. **Monitoring, Icinga: Open Source.** Home - Icinga: Open Source Monitoring. *Home - Icinga: Open Source Monitoring*. [Online] 2012. http://www.icinga.org.

23. —. Architecture - Icinga: Open Source Monitoring. *Icinga: Open Source Monitoring*. [Online] 2012. https://www.icinga.org/nagios/architecture/.

24. **Team, Icinga Development.** Icinga Version 1.7 Documentation. *Icinga Version 1.7 Documentation*. [Online] 2009-2012. [Citace: 20. květen 2012.] http://docs.icinga.org/latest/en/.

25. **Nagios Enterprises, LLC.** Nagios - The Industry Standard in IT Infrastructure Monitoring. *Nagios - The Industry Standard in IT Infrastructure Monitoring*. [Online] 2012. http://nagios.org/.

26. **Membrey, Peter, Verhoeven, Tim a Angenendt, Ralph.** Monitoring Your Network Using Nagios. *The Definitive Guide to CentOS*. místo neznámé : Apress, 2009.

27. **Ltd., Opsview.** Opsview | IT Monitoring for Networks, Applications, Virtual Servers & the Cloud. *Opsview | IT Monitoring for Networks, Applications, Virtual Servers & the Cloud.* [Online] 2012. http://www.opsview.org.

28. Opsview Documentation. *Opsview Documentation.* [Online] 2012. http://docs.opsview.com/doku.php.

29. Zenoss Community - Open Source Network Monitoring and Systems Management. *Zenoss Community.* [Online] 2005-2011. http://community.zenoss.org/index.jspa.

30. Space: Wiki - Open Source Network Monitoring and Systems Management. *Zenoss Community.* [Online] 2005-2011. http://community.zenoss.org/community/documentation/wiki.

31. **SIA, Zabbix.** Homepage of Zabbix :: An Enterprise-Class Open Source Distributed Monitoring Solution. *Zabbix SIA.* [Online] 2001-2012.

32. —. Documentation. *Homepage of Zabbix :: An Enterprise-Class Open Source Distributed Monitoring Solution.* [Online] 2010. http://www.zabbix.com/documentation.php.

33. opsview3.1 - Opsview Documentation. *Opsview Documentation.* [Online] 2009. http://docs.opsview.org/doku.php?id=opsview3.1.

34. psview3.1:centos-installation - Opsview Documentation. *OpsView Documentation.* [Online] 2009. http://docs.opsview.org/doku.php?id=opsview3.1:centos-installation.

Příloha A – komentovaný konfigurační soubor agenta na systému Linux

```
#############################################################################
####
# Sample NRPE Config File
# Written by: Ethan Galstad (nagios@nagios.org)
#
# Last Modified: 12-11-2006
#
# NOTES:
# This is a sample configuration file for the NRPE daemon.  It needs to
be
# located on the remote host that is running the NRPE daemon, not the
host
# from which the check_nrpe client is being executed.
#############################################################################
####

# PID FILE
# The name of the file in which the NRPE daemon should write it's process
ID
# number.  The file is only written if the NRPE daemon is started by the
root
# user and is running in standalone mode.

pid_file=/var/run/nrpe.pid

# PORT NUMBER
# Port number we should wait for connections on.
# NOTE: This must be a non-priviledged port (i.e. > 1024).
# NOTE: This option is ignored if NRPE is running under either inetd or
xinetd

server_port=5666

# SERVER ADDRESS
# Address that nrpe should bind to in case there are more than one
interface
# and you do not want nrpe to bind on all interfaces.
# NOTE: This option is ignored if NRPE is running under either inetd or
xinetd

#server_address=192.168.1.1

# NRPE USER
# This determines the effective user that the NRPE daemon should run as.
# You can either supply a username or a UID.
#
# NOTE: This option is ignored if NRPE is running under either inetd or
xinetd

nrpe_user=nrpe

# NRPE GROUP
# This determines the effective group that the NRPE daemon should run as.
# You can either supply a group name or a GID.
#
```

```
# NOTE: This option is ignored if NRPE is running under either inetd or
xinetd

nrpe_group=nrpe

# ALLOWED HOST ADDRESSES
# This is an optional comma-delimited list of IP address or hostnames
# that are allowed to talk to the NRPE daemon.
#
# Note: The daemon only does rudimentary checking of the client's IP
# address.  I would highly recommend adding entries in your
/etc/hosts.allow
# file to allow only the specified host to connect to the port
# you are running this daemon on.
#
# NOTE: This option is ignored if NRPE is running under either inetd or
xinetd

# localhost povolit jen na testovani
#allowed_hosts=217.11.226.6,127.0.0.1
allowed_hosts=217.11.226.6


# COMMAND ARGUMENT PROCESSING
# This option determines whether or not the NRPE daemon will allow
clients
# to specify arguments to commands that are executed.  This option only
works
# if the daemon was configured with the --enable-command-args configure
script
# option.
#
# *** ENABLING THIS OPTION IS A SECURITY RISK! ***
# Read the SECURITY file for information on some of the security
implications
# of enabling this variable.
#
# Values: 0=do not allow arguments, 1=allow command arguments

dont_blame_nrpe=1

# COMMAND PREFIX
# This option allows you to prefix all commands with a user-defined
string.
# A space is automatically added between the specified prefix string and
the
# command line from the command definition.
#
# *** THIS EXAMPLE MAY POSE A POTENTIAL SECURITY RISK, SO USE WITH
CAUTION! ***
# Usage scenario:
# Execute restricted commmands using sudo.  For this to work, you need to
add
# the nagios user to your /etc/sudoers.  An example entry for alllowing
# execution of the plugins from might be:
#
# nagios          ALL=(ALL) NOPASSWD: /usr/lib/nagios/plugins/
#
# This lets the nagios user run all commands in that directory (and only
them)
```

```
# without asking for a password.  If you do this, make sure you don't
give
# random users write access to that directory or its contents!

#command_prefix=/usr/bin/sudo


# DEBUGGING OPTION
# This option determines whether or not debugging messages are logged to
the
# syslog facility.
# Values: 0=debugging off, 1=debugging on

debug=1


# COMMAND TIMEOUT
# This specifies the maximum number of seconds that the NRPE daemon will
# allow plugins to finish executing before killing them off.

command_timeout=60



# CONNECTION TIMEOUT
# This specifies the maximum number of seconds that the NRPE daemon will
# wait for a connection to be established before exiting. This is
sometimes
# seen where a network problem stops the SSL being established even
though
# all network sessions are connected. This causes the nrpe daemons to
# accumulate, eating system resources. Do not set this too low.

connection_timeout=300


# WEEK RANDOM SEED OPTION
# This directive allows you to use SSL even if your system does not have
# a /dev/random or /dev/urandom (on purpose or because the necessary
patches
# were not applied). The random number generator will be seeded from a
file
# which is either a file pointed to by the environment valiable $RANDFILE
# or $HOME/.rnd. If neither exists, the pseudo random number generator
will
# be initialized and a warning will be issued.
# Values: 0=only seed from /dev/[u]random, 1=also seed from weak
randomness

#allow_weak_random_seed=1

# INCLUDE CONFIG FILE
# This directive allows you to include definitions from an external
config file.

include=/etc/nagios/command-arit.cfg

# INCLUDE CONFIG DIRECTORY
# This directive allows you to include definitions from config files
(with a
```

```
# .cfg extension) in one or more directories (with recursion).

#include_dir=<somedirectory>
#include_dir=<someotherdirectory>
#include_dir=/etc/nrpe.d


# COMMAND DEFINITIONS
# Command definitions that this daemon will run.  Definitions
# are in the following format:
#
# command[<command_name>]=<command_line>
```

Příloha B – komentovaný konfigurační soubor agenta na systému Windows

```
##############################################################################
####
# Sample NRPE Config File
#
# NOTES:
# This is a sample configuration file for the NRPE_NT service.  It needs
to be
# located on the remote host that is running the NRPE_NT service, not the
host
# from which the check_nrpe client is being executed.
##############################################################################
####

# PORT NUMBER
# Port number we should wait for connections on.

server_port=5666

# SERVER ADDRESS
# Address that nrpe should bind to in case there are more than one
interface
# and you do not want nrpe to bind on all interfaces.

#server_address=192.168.1.1

# ALLOWED HOST ADDRESSES
# This is a comma-delimited list of IP address of hosts that are allowed
# to talk to the NRPE daemon.
#
# NOTE: The daemon only does rudimentary checking of the client's IP
#       address.

allowed_hosts=217.11.226.6

# COMMAND ARGUMENT PROCESSING
# This option determines whether or not the NRPE_NT service will allow
clients
# to specify arguments to commands that are executed
# *** ENABLING THIS OPTION IS A SECURITY RISK! ***
#
# Values: 0=do not allow arguments, 1=allow command arguments

dont_blame_nrpe=1

# DEBUGGING OPTION
# This option determines whether or not debugging messages are logged to
the
# eventlog.
# Values: 0=debugging off, 1=debugging on

debug=1

# COMMAND TIMEOUT
# This specifies the maximum number of seconds that the NRPE_NT service
will
# allow plugins to finish executing before killing them off.
```

```
command_timeout=30

# INCLUDE CONFIG FILE
# This directive allows you to include definitions from an external
config file.

#include=<somefile.cfg>
include=c:\Program Files\nrpe-nt\command-arit.cfg

# INCLUDE CONFIG DIRECTORY
# This directive allows you to include definitions from config files
(with a
# .cfg extension) in one or more directories (with recursion).
#NOTE: This option is currently ignored with NRPE_NT!
#include_dir=<somedirectory>
#include_dir=<someotherdirectory>

# LOGLEVEL / NRPE_NT only
# severity of events logged to nrpe_nt.log if debug = 1
# Useful Values:
# 1: Log Critical
# 4: Log Errors (Default)
# 6: Log Informational
# 7: Log Debug
loglevel=4

# USE_WIN_METACHARS / NRPE_NT only
# use NASTY_METACHARS_WIN, allow \ and " to allow easier passing of
# pathnames as parameter
# Values: 0=use default NASTY_METACHARS definition, 1=use relaxed
NASTY_METACHARS_WIN definition
use_win_metachars=1

# COMMAND DEFINITIONS
# Command definitions that this daemon will run.  Definitions
# are in the following format:
#
# command[<command_name>]=<command_line>
#
# When the daemon receives a request to return the results of
<command_name>
# it will execute the command specified by the <command_line> argument.
#
# Unlike Nagios, the command line cannot contain macros - it must be
# typed exactly as it should be executed.
#
# Note: Any plugins that are used in the command lines must reside
# on the machine that this daemon is running on!  The examples below
# assume that you have plugins installed in a D:\NRPE_NT
# directory.  Also note that you will have to modify the definitions
below
# to match the argument format the plugins expect.  Remember, these are
# examples only!

# The following examples use no command arguments...

#command[check_hallo]=D:\NRPE_NT\hallo.exe
#command[check_cmd]=D:\NRPE_NT\test.cmd
#command[check_perl]=D:\bin\perl.exe D:\NRPE_NT\test.pl
```

```
# The following examples allow user-supplied arguments and can
# only be used if NRPE_NT  was compiled with support for
# command arguments *AND* the dont_blame_nrpe directive in this
# config file is set to '1'...

#command[check_arg]=D:\NRPE_NT\testarg.cmd $ARG1$
#command[check_arg]=D:\NRPE_NT\testarg.exe -H $ARG1$ -p $ARG2$
```

Příloha C – soubor command-arit.cfg na systému Linux

```
################################################################################
# Standard ARIT checks                                                         #
################################################################################

command[check_raid]=/usr/lib64/nagios/plugins/check_linux_raid
command[check_hd_root]=/usr/lib64/nagios/plugins/check_disk -w 20% -c 10%
-p /dev/mapper/VolGroup-lv_root

command[check_hd_home]=/usr/lib64/nagios/plugins/check_disk -w 20% -c 10%
-p /dev/md2

command[check_hd_backup]=/usr/lib64/nagios/plugins/check_disk -w 20% -c
10% -p /dev/sdb1

command[check_vmware_swap]=/usr/lib64/nagios/plugins/check_disk -w 10% -c
5% -p /tmp/vmware

command[check_swap]=/usr/lib64/nagios/plugins/check_swap -w 75% -c 50%
command[check_mysql]=/usr/lib64/nagios/plugins/check_mysql -H localhost -
p <heslo>

command[check_pgsql]=/usr/lib64/nagios/plugins/check_pgsql -H localhost -
p <heslo>

command[check_reboot]=/usr/lib64/nagios/plugins/check_restart.sh
/var/log/restart.log

command[check_cputemp]=/usr/lib64/nagios/plugins/check_cpu_temp
command[check_mailq]=/usr/lib64/nagios/plugins/check_mailq -w 200 -c 500
command[check_securitu]=/usr/lib64/nagios/plugins/check_security_update
command[check_smart_temp]=sudo /usr/lib64/nagios/plugins/check_smart_temp
-d /dev/sda /dev/sdb -w 50 -c 60

command[check_smart_reallocated_sector]=sudo
/usr/lib64/nagios/plugins/check_smart_reallocated_sector -d /dev/sda
/dev/sdb -w 1 -c 10

command[check_smtp]=/usr/lib64/nagios/plugins/check_smtp -H localhost
command[check_restart]=/usr/lib64/nagios/plugins/check_restart.sh
/var/log/restart.log

#command[check_strecha]=/usr/lib64/nagios/plugins/check_ping -H
10.192.207.185 -w 10,10% -c 20,20%

command[check_restart]=/usr/lib64/nagios/plugins/check_restart.sh
/var/log/restart.log

command[check_load]=/usr/lib64/nagios/plugins/check_load -r -w 8,8,8 -c
12,12,12

command[check_users]=/usr/lib64/nagios/plugins/check_users -w 5 -c 10
command[check_tps]=/usr/lib64/nagios/plugins/check_tps.sh -w 200 -c 300
command[check_procs]=/usr/lib64/nagios/plugins/check_procs -w 200 -c 300
```

Příloha D – soubor command-arit.cfg na systému Windows

```
#HDD
command[nt_check_disk_c]=C:\Program Files\nrpe-
nt\bin\diskspace_nrpe_nt.exe b: 93 98

command[nt_check_disk_c]=C:\Program Files\nrpe-
nt\bin\diskspace_nrpe_nt.exe c: 80 90

command[nt_check_disk_d]=C:\Program Files\nrpe-
nt\bin\diskspace_nrpe_nt.exe d: 95 99

command[nt_check_disk_e]=C:\Program Files\nrpe-
nt\bin\diskspace_nrpe_nt.exe e: 80 90

command[nt_check_disk_f]=C:\Program Files\nrpe-
nt\bin\diskspace_nrpe_nt.exe g: 80 90

command[nt_check_disk_i]=C:\Program Files\nrpe-
nt\bin\diskspace_nrpe_nt.exe i: 80 90
command[nt_check_disk_f]=C:\Program Files\nrpe-
nt\bin\diskspace_nrpe_nt.exe o: 80 90

#Performace
command[nt_cpuload]=C:\Program Files\nrpe-nt\bin\cpuload_nrpe_nt.exe 50
80
command[nt_memload]=C:\Program Files\nrpe-nt\bin\memload_nrpe_nt.exe 80
90
command[nt_ipconfig]=C:\Program Files\nrpe-nt\bin\check_ipconfig.exe
192.168.10.254
command[nt_pagefile]=C:\Program Files\nrpe-nt\bin\check_pagefile.exe --
used-crit-percent=90% --used-warn-percent=75%
command[nt_shadowcopy]=C:\Program Files\nrpe-nt\bin\check_shadowcopy.exe
c:

#Program / Service
command[nt_service]=C:\Program Files\nrpe-nt\bin\service_nrpe_nt.exe
"Backup Exec Server,FileZilla Server FTP server,SQL Server
(MSSQLSERVER),Server DHCP,Server DNS,Služba AD DS (Active Directory
Domain Services)"
command[nt_service_nodserver]=C:\Program Files\nrpe-
nt\bin\service_nrpe_nt.exe "ESET Remote Administrator Server"
command[nt_service_sql]=C:\Program Files\nrpe-nt\bin\service_nrpe_nt.exe
"SQL Server (AUTODESKVAULT)"
command[nt_service_wsus]=C:\Program Files\nrpe-nt\bin\service_nrpe_nt.exe
"Služba Update Service"
command[nt_processes]=C:\Program Files\nrpe-nt\bin\check_processes.exe /
jmeno procesu k hlidani /
command[nt_service_stoped]=C:\Program Files\nrpe-
nt\bin\check_services_stopped.exe /jmeno zastaveneho servisu/


#System State
command[nt_eventlog]=C:\Program Files\nrpe-nt\bin\eventlog_nrpe_nt.exe -m
7200 -s "Service Control Manager"
command[nt_uptime]=C:\Program Files\nrpe-nt\bin\check_uptime.exe --
critical=43200 --warning=120600
command[nt_userquota]=C:\Program Files\nrpe-nt\bin\check_userquota.exe
/pisnko disku kde chceme hlidat qvoty/
```

```
command[nt_netshare]=C:\Program Files\nrpe-nt\bin\check_netshares.exe
/naze sdileni ktere chceme hlidat/
command[nt_printers]=C:\Program Files\nrpe-nt\bin\check_printers.exe / --
help parametry hlida pripojene tiskarny/
command[nt_taskscheduler]=C:\Program Files\nrpe-
nt\bin\check_taskscheduler.exe -t /jmeno tasku/ -w /kolik sec od
posledniho spusteni/ -c /jako w/
command[nt_uname]=C:\Program Files\nrpe-nt\bin\check_uname.exe
command[check_backupexec]=c:\windows\system32\cscript.exe //NoLogo //T:10
"C:\Program Files\nrpe-nt\bin\check_backupexec.wsf"
command[check_update]=c:\windows\system32\cscript.exe //NoLogo //T:10
"C:\Program Files\nrpe-nt\bin\check_updates.wsf"

#Network
command[nt_tcp]=C:\Program Files\nrpe-nt\bin\check_tcp.exe -H /hostmane/
-p /port/ -w /timeoout/ -c /tomeout/
command[check_ping_qnap]=C:\Program Files\nrpe-nt\bin\check_ping.exe -H
192.168.10.1

#test
command[check_adaptecraid]=c:\windows\system32\cscript.exe //NoLogo
//T:10 "C:\Program Files\nrpe-nt\bin\adaptec\check_adaptec.wsf"

#RAID
command[check_swraid]=c:\windows\system32\cscript.exe //NoLogo //T:10
"C:\Program Files\nrpe-nt\bin\check_windows_raids.wsf"
command[check_restart]=C:\Program Files\nrpe-nt\bin\check_restart.cmd
```

Příloha E – skript pro zjištění aktuálního počtu spojení

```
#!/bin/bash

OK=0
WARNING=1
CRITICAL=2
UNKNOWN=3
VALUE_WARNING=$1
VALUE_CRITICAL=$2

# ziskani hodnoty
err="`sudo netstat  -anp | grep tcp | wc  -l`"

#porovnání s warn a crit a navraceni stavu
if [ $err -lt $VALUE_WARNING ]
then
  echo "CONN OK - Number of connections: $err on $HOSTNAME OK;|
TCP_conn="$err";"$VALUE_WARNING";"$VALUE_CRITICAL""
  exit $OK
else
  if [ $err -lt $VALUE_CRITICAL ]
  then
    echo "CONN WARNING - Number of connections: $err on $HOSTNAME OK;|
TCP_conn=  "$err";"$VALUE_WARNING";"$VALUE_CRITICAL""
    exit $WARNING
  else
    echo "CONN CRITICAL - Number of connections: $err on $HOSTNAME OK;|
TCP_conn="$err";"$VALUE_WARNING";"$VALUE_CRITICAL""
    exit $CRITICAL
  fi
  exit $OK
fi
```

Příloha F – skript pro ověření zaplacených faktur

```
#!/bin/bash
# 0 = zaplceno, 1 = do 30ti dnu po stplatnost, 2=více jak tricet dnu po
#splatnosti
VYPISTEXT=""
OUT=0

#dotazem  do API FlexiBee zjistime stav faktur
VALUE=`curl -u user:password -L -f
"http://192.168.99.249:5434/c/arit_nabidky/uzivatelsky-
dotaz/22/call.xml?id=${1}&detail=full" | grep result | cut -b 13`


#pokud nedostaneme odpoved něco se pokazilo
if [ -z $VALUE ];
then
    VYPISTEXT="UNKNOW: Nepodarilo se ziskat data!"
    OUT=3
else
    if [ "$VALUE" -eq 0 ];
    then
      VYPISTEXT="OK: Vsechny pohledavky radne zaplaceny !"
    else
        if [ "$VALUE" -eq 1 ];
      then
          VYPISTEXT="WARNING: Neuhrazena faktura do 30ti dnu po
splatnosti! "
          OUT=1
      else
          if [ "$VALUE" -eq 2 ];
          then
            VYPISTEXT="CRITICAL: !!STOPSTAV!! Faktura po splatnosti vice
jak 30 dni !!STOPSTAV!!"
            OUT=2
          fi
      fi
    fi
fi

echo $VYPISTEXT
exit $OUT
```

Příloha G – skript pro kontrolu teploty procesoru

```
#!/bin/bash
# author pleskot
# vytvoreno dle vzoru check_smart_temp by aleshus
# posledni uprava 16.12.11

OK="LM_SENSORS_CHECK "
ERR=""
ERRMSG="SMART_CHECK ERROR: "
WARN=-300
CRIT=-300
WARNFLAG=0
CRITFLAG=0
TEMP=0
OUT=0

##-- rozprarsovani parametru --##
while [ $# -gt 0 ]; do

while ( echo $2 | egrep '^-[dwch]$' >/dev/null 2>/dev/null ); do
      shift
done

case "$1" in
      "-w" )
            shift
            WARN="$1"
            if !(echo $WARN | egrep '^-?[0-9]*$' >/dev/null 2>/dev/null);
            then
                  ERR="Spatny format hodnoty hodnoty pro warning"
            fi
            shift
            ;;
      "-c" )
                shift
                CRIT="$1"
                if !(echo $CRIT | egrep '^-?[0-9]*$' >/dev/null
2>/dev/null);
                then
                     ERR="Spatny format kriticke hodnoty"
            fi
                shift
                ;;

      "-h" | * )
            echo "===========check_CPU_temp scritp==============="
            echo
            echo " -w warning hodnota"
            echo " -c kriticka hodnota"
            echo
            echo "================================================"
            exit 0
            ;;
esac

done

if [ "$WARN" -gt "$CRIT" ];
then
```

```
        ERR="Warning hodnota nesmi byt vyssi nez kriticka";
fi

if [ "$CRIT" -eq -300 ];
then
      ERR="Chybi kriticka hodnota";
fi

if [ "$WARN" -eq -300 ];
then
      ERR="Chybi warning hodnota";
fi

if [ "$ERR" ];
then
      echo "$ERRMSG $ERR";
      exit -1
else
            TEMP="`sensors | grep CPU1 | cut -d' ' -f3 | cut -d'+' -f2 |
cut -d'°' -f1`"
            if [ -z $TEMP ];
            then
                  echo "$ERRMSG Nelze ziskat potrebne informace"
                  exit -1
            else
                if [ $TEMP -gt $CRIT ]; then CRITFLAG=1;
                elif [ $TEMP -gt $WARN ]; then WARNFLAG=1
                fi
            fi
            OKVYPIS="$OKVYPIS Teplota CPU: $TEMP °C; "
            OKVYPISTEXT="$OKVYPISTEXT CPU_Temp=$TEMP;$WARN;$CRIT;0;0;"

      if [ $CRITFLAG -gt 0 ];
        then
                OK="$OK CRITICAL: "
                OUT=2
        elif [ $WARNFLAG -gt 0 ];
        then
                OK="$OK WARNING: "
                OUT=1
        else OK="$OK OK: "
      fi
      OK="$OK $OKVYPIS|$OKVYPISTEXT"
      echo $OK
      exit $OUT
fi
```