Projektdokumentation

Implementierung einer Überwachungslösung in ein bestehendes Netzwerk

Fachinformatiker der Fachrichtung Systemintegration – Sommer 2008

Auszubildender:	**Julian Wilhelm** Neuwarfer Str. 6 26434 Wangerland * 07.08.1988 in Wilhelmshaven
Ausbildungsbetrieb:	Stadt Wilhelmshaven Städtische Datenverarbeitung Wilhelmshaven Bismarckstraße 162 26382 Wilhelmshaven
Projektbetreuer:	Thorsten Breer
Prüflingsnummer:	161 50156
Ausbildungsberuf:	Fachinformatiker der Fachrichtung Systemintegration
Durchführungszeitraum:	07.04.2008 bis 11.04.2008
Stand:	11.04.2008

2008

Implementierung einer Überwachungslösung in ein bestehendes Netzwerk

Julian Wilhelm

Städtische Datenverarbeitung Wilhelmshaven

11.04.2008

Julian Wilhelm

Dokumentationsinhalt

Tabellenverzeichnis

1 Einleitung

Das vorliegende Dokument ist die Dokumentation der Projektarbeit, die im Rahmen der Abschlussprüfung zum Fachinformatiker der Fachrichtung Systemintegration durchgeführt wurde.

Ich versichere, dass ich das Projekt selbständig und ohne fremde Hilfe anfertige und alle Stellen, die ich wörtlich oder annähernd wörtlich aus Veröffentlichungen entnehme, als solche kenntlich mache. Die Arbeit hat in dieser Form keiner anderen Prüfungsinstitution vorgelegen.

Das Projekt wurde im Zeitraum vom 07.04.2008 bis zum 11.04.2008 vollständig im Hause der Städtischen Datenverarbeitung Wilhelmshaven durchgeführt. Die Planung und die Umsetzung fanden innerhalb der Städtischen Datenverarbeitung Wilhelmshaven statt.

2 Problemstellung und Zielsetzung

Für die Städtische Datenverarbeitung Wilhelmshaven, die als Eigenbetrieb der Stadt Wilhelmshaven als Dienstleister im EDV- und Telekommunikationsbereich tätig ist, ist es notwendig, eine Überwachungslösung für die einzelnen Server und Standorte in das bestehende Netzwerk zu implementieren.

Da sich nicht alle Server im Serverraum befinden, sondern vereinzelt auch an den Außenstandorten, ist eine komplette Überwachung durch die vorhandenen Administratoren zurzeit fast unmöglich. Ein Ausfall von Diensten oder sogar der Ausfall eines kompletten Servers macht sich oft erst dann bemerkbar, wenn die Anwender das Problem bei der Störungshotline melden. Durch das nicht frühzeitige Erkennen eines Problems und den daraus resultierenden Folgen können erhebliche Kosten entstehen.

Ziel dieses Projektes ist es, mit Hilfe einer geeigneten Serveranwendung eine Möglichkeit zu finden, die Netzwerkinfrastruktur der Städtischen Datenverarbeitung Wilhelmshaven transparenter und übersichtlicher zu gestalten.

Die Überwachungssoftware soll dazu beitragen, das Probleme frühzeitig erkannt und die Ursachen der Probleme so schnell wie möglich beseitigt werden, um die Ausfallzeiten sowie die damit entstehenden Kosten für die Stadt Wilhelmshaven so gering wie möglich zu halten.

Es soll eine möglichst kostengünstige Serveranwendung in das Netzwerk implementiert werden, welche von den Administratoren zentral verwaltet werden kann.

3 IST Analyse

Das Netzwerk der Städtischen Datenverarbeitung Wilhelmshaven ist eine komplexe Netzwerkinfrastruktur, an das die einzelnen Standorte und Außenstellen der Stadtverwaltung Wilhelmshaven sowie ein Teil der umliegenden Schulen angeschlossen sind.

Neben einer kleineren Anzahl Linux – Server, welche für E - Mail und Anti – Spam Dienste eingesetzt werden, werden ausschließlich Microsoft Windows 2003 Server eingesetzt. Die Server bieten unter anderem Mail-, Drucker-, Datei- und Terminaldienste an.

Das bestehende Netzwerk umfasst insgesamt 42 Server, auf die von mehr als 600 Windows – Clients zugegriffen wird. Das Netzwerk der Städtischen Datenverarbeitung ist so groß, das eine komplette Überwachung durch die vorhandenen Administratoren fast unmöglich ist. So machen sich Ausfälle leider erst dann bemerkbar, wenn Anwender nicht mehr mit entsprechenden Anwendungen arbeiten können und das Problem bei der Störungshotline melden. Probleme lassen sich in solchen Fällen meist schwer lokalisieren, da Schritt für Schritt alle Netzwerk – Komponenten überprüft werden müssen.

Das die Anwender, die Städtische Datenverarbeitung auf Störungen im Netzwerk aufmerksam machen müssen, ist nicht weiter tragbar. Für die Fehlersuche und die anschließende Fehlerbehebung werden im Durchschnitt von den Administratoren 15 Minuten benötigt.

3.1 Städtische Datenverarbeitung Wilhelmshaven

Die Städtische Datenverarbeitung Wilhelmshaven stellt ihren Anwendern eine Windows Domäne zur Verfügung, die Funktionen wie servergespeicherte Profile und Laufwerke bereitstellt. Es gibt einen Primary – Domain – Controller mit insgesamt 2 Backup – Domain – Controllern, die in unterschiedlichen Serverräumen in Betrieb sind. Die Dateiserver, sowie die Mailumgebung werden ebenfalls über mehrere Windows – Server realisiert und sind großer Bestandteil des Netzwerkes. Zusätzlich stellt die Städtische Datenverarbeitung so genannte Terminalserver für die angeschlossenen Außenstandorte und Schulen zur Verfügung.

Die Außenstandorte und Schulen werden über Standleitung, Internet und Funkstrecken an das Netzwerk der Städtischen Datenverarbeitung angeschlossen und arbeiten größten Teils auf den vorhandenen Terminalservern.

3.2 Erfassung der vorhandenen Server und Dienste

Die Städtische Datenverarbeitung verwaltet einen Großteil von Servern, die sich zum Teil durch Einsatz von VMWare Enterprise, in einer virtuellen Umgebung befinden. Ein Teil dieser Server und Dienste, soll durch die Monitoring – Lösung überwacht werden.

Physikalische Server

Serveranzahl	Hardwareausstattung	Dienste
1	1 x 993 MHz, 1.0 GB RAM	Anti Viren – Server
1	1 x 1.3 GHz, 1.0 GB RAM	Programm- und File – Server
1	2 x 1.0 GHz, 2.0 GB RAM	Image – Server
1	1 x 1.8 GHz, 1.0 GB RAM	Fax – Server
1	2 x 1.0 GHz, 1.5 GB RAM	FTP – Server
1	2 x 1.4 GHz, 1.0 GB RAM	Blackberry – Server
1	2 x 1.5 GHz, 1.5 GB RAM	Gateway – Server
1	2 x 3.0 GHz, 2.0 GB RAM	Web Gis – Server
1	2 x 3.0 GHz, 4.0 GB RAM	Global Catalog Server
2	2 x 2.8 GHz, 4.0 GB RAM	Terminal- und ISA – Server
2	4 x 2.0 GHz, 2.0 GB RAM	Sicherungs- und Print – Server
3	2 x 3.0 GHz, 2.0 GB RAM	Unicenter, VMWare und USV Mopups
5	1 x 3.0 GHz, 1.0 GB RAM	NAS-, File- und Oracle - Server
2	2 x 3.0 GHz, 16.0 GB RAM	VMWare – Server
2	4 x 2.0 GHz, 32.0 GB RAM	VMWare – Server

Tabelle 1 Physikalische Server

Virtuelle Server

Serveranzahl	Hardwareausstattung	Dienste
1	1 x CPU, 2.0 GB RAM	File - Server
1	1 x CPU, 4.0 GB RAM	Microsoft SQL - Server
2	2 x CPU, 2.5 GB RAM	Lotus Domino - Server
3	2 x CPU, 2.0 GB RAM	Citrix & Application - Server
3	2 x CPU, 4.0 GB RAM	Oracle Datenbank - Server
7	1 x CPU, 1.0 GB RAM	Domain-, Print- und File- Server

Tabelle 2 Virtuelle Server

Die Gesamtzahl der von der Städtischen Datenverarbeitung verwalteten Server, liegt aktuell bei 42. Hierbei handelt es sich um 25 Physikalische- und 17 virtuelle Server.

4 SOLL Konzept

4.1 Ermittlung der Kundenwünsche

Im Vorfeld des Projektes wurde mir mitgeteilt, dass eine möglichst kostengünstige Monitoring – Lösung in das Netzwerk implementiert werden soll. Eine zentrale Administration der Software soll außerdem gewährleistet werden, damit die Administratoren bei auftretenden Problemen von überall überprüfen können, auf welchem Server das Problem besteht.

Zudem muss die Software die Erreichbarkeit der Server, einzelner Dienste und Prozesse, die CPU- und Speicherauslastung sowie die noch zur Verfügung stehenden Server – Ressourcen überwachen. Die Städtische Datenverarbeitung wünscht zudem ein Produkt, welches die Funktion der Netzwerk – Komponenten Überwachung ermöglicht.

Sobald ein Server ausfällt, verteilt sich die Last auf die noch zur Verfügung stehenden Server, was eine erhöhte CPU- und Speicherauslastung zur Folge hat.

Bei einem auftretenden Problem, wie zum Beispiel der nicht Erreichbarkeit eines Servers, sollen die zuständigen Administratoren per E-Mail über das Problem informiert werden.

4.2 Lösungsmöglichkeiten in der Kosten – Nutzen Analyse

Berücksichtigt man die oben genannten Punkte, ist die Implementierung einer OpenSource – Lösung erwünscht, um die Kosten für die Städtische Datenverarbeitung so gering wie möglich zu halten.

Des Weiteren wird eine leicht verwaltbare Softwarelösung gefordert, mit der es möglich ist, die Erreichbarkeit der Server, einzelner Dienste und Prozesse, CPU- und Speicherauslastung sowie andere Server – Ressourcen zu überwachen. Durch die Software sollen sowohl Windows- und Linux - Server sowie Netzwerkkomponenten überwacht werden.

Ich werde der Städtischen Datenverarbeitung die OpenSource Monitoring – Lösungen

- Zabbix von Alexei Vladishev und
- Nagios des Entwicklers Ethan Gakstad

als auch das kommerzielle Produkt

- WhatsUp Gold von IPSwitch

vorstellen.

Die OpenSource Produkte Zabbix und Nagios sind über die GNU lizenziert und gelten daher als freie Software. Beide Produkte laufen unter dem ebenfalls freien Betriebssystem Linux. Für OpenSource – Software gibt es weltweit Programmierer, welche aufbauend auf dem OpenSource – Gedanken die Software weiterentwickeln und Fehler beseitigen.

In der folgenden Gegenüberstellung werden die drei ausgewählten Monitoring – Lösungen verglichen. Die Programme werden in den Punkten Kosten, Konfiguration, Leistungsumfang, Modularität und Support gegenübergestellt. Als Faktor werden Punkte von eins bis drei vergeben, wobei drei einen besonders wichtigen Punkt darstellt. Leistungspunkte werden für die Produkte von eins bis zehn vergeben. Die Städtische Datenverarbeitung legt besonderen Wert auf niedrige Kosten und guten Support.

Kriterium	**Faktor**	**Zabbix**		**Nagios**		**WhatsUp Gold**	
		Leistungs-punkte	Erreichte Punkte	Leistungs-punkte	Erreichte Punkte	Leistungs-punkte	Erreichte Punkte
Kosten	3	10	30	10	30	2	6
Konfiguration	2	7	14	8	16	7	14
Leistungen	2	8	16	9	18	8	16
Modularität	2	8	16	9	18	6	12
Support	3	4	12	9	27	6	18
Summe			**88**		**109**		**66**

Tabelle 3 Produktvergleich

4.2.1 Produktkosten

In der unteren Tabelle werden die Kosten der drei Monitoring – Lösungen aufgezeigt. Für die Produkte Zabbix und Nagios entstehen durch die GNU Lizenzierung keine Kosten. Der Preis für das Produkt WhatsUp Gold wurde bei einem Vertragslieferanten der Städtischen Datenverarbeitung angefragt (Anlage A).

Kostenpunkt	**Zabbix**	**Nagios**	**WhatsUp Gold**
Serverlizenz	0,00 €	0,00 €	1.707,65 €
Clientlizenz	0,00 €	0,00 €	100 Clients inklusive
Summe	**0,00 €**	**0,00 €**	**1.707,65 €**

Tabelle 4 Produktkosten

4.3 Entscheidungsfindung

Nach der Vorstellung der drei Monitoring – Lösungen als Lösungsmöglichkeiten hat sich die Städtische Datenverarbeitung für das Produkt Nagios entschieden. Damit ist Nagios ein weiteres OpenSource Produkt welches bei der Städtischen Datenverarbeitung eingesetzt wird.

Die Monitoring – Software Nagios übernimmt die komplette Überwachung von Diensten und Ressourcen. Es überwacht die Erreichbarkeit der Server, einzelner Dienste und Prozesse, die CPU- und Speicherauslastung sowie die noch zur Verfügung stehenden Ressourcen auf den Servern. Die zu überwachenden Hosts müssen Nagios über Konfigurationsdateien bekannt gemacht werden. Hosts lassen sich zur besseren Übersicht in Gruppen zusammenfassen. In den Konfigurationsdateien kann ebenfalls festgelegt werden, was auf den jeweiligen Hosts überwacht werden soll. Die gesammelten Daten stellt Nagios übersichtlich in einem Webinterface dar. Bei einer Warnung oder einem kritischen Zustand werden die zuständigen Administratoren per E-Mail informiert.

Nagios wird die Überwachung 24 Stunden am Tag an sieben Tagen in der Woche durchführen. Die Werte des Interfaces werden regelmäßig aktualisiert und bei Über- oder Unterschreitung der Schwellenwerte eine Warnung- oder ein Ausfallstatus angezeigt.

Für die Konfiguration von Nagios 3.0 steht zurzeit kein Web – Frontend zur Verfügung.

Ein Teil der Administratoren der Städtischen Datenverarbeitung verfügen über Vorwissen im Bereich Linux – Administration.

4.3.1 Vor- und Nachteile von OpenSource Software

Vorteile

- Weltweite Entwicklung
- Entwickelt wird nach Bedarf, kein Einsatzzwang
- Software kostenlos
- Keine Einschränkungen
- Kein Lizenznachkauf

Nachteile

- Keine Sicherheit auf weitere Pflege der Software
- Keine Gewährleistung
- Bedarf und Nutzen können variieren

4.4 Erforderliche Hard- und Software festlegen

4.4.1 Hardware festlegen

Im Projektvorfeld konnte nicht geklärt werden, ob der Monitoring – Server in einer virtuellen Umgebung oder auf separater Hardware installiert werden soll. Aus diesem Grund wurde ein Angebotsvergleich zwischen den Lieferanten Reichelt, Mindfactory und BSH – EDV erstellt.

Nach einer gesetzten Frist von 24 Stunden wurden keine Angebote mit Server – Hardware zugesandt. Aus diesem Grund wurde der Angebotsvergleich mit Client – Hardware durchgeführt.

Das benötigte Zubehör wie Monitor, Tastatur und Maus sind bereits vorhanden.

Stückzahl	Bezeichnung	Produkt
1	CPU	Intel 3,2 GHz
1	Mainboard	Asus P5K
1	Arbeitsspeicher	1 GB DDR2 RAM Infineon / Corsair
1	Festplatte	Samsung SATA2 - 160 GB
1	DVD – Laufwerk	LiteOn DVD-ROM 16/48x
1	Netzteil	Coba 500 Watt ATX 2.2
1	Gehäuse	Miditower ATX

Tabelle 5 Serverhardware

Die Lieferanten wurden einem direkten Vergleich in den Punkten Kulanz, Erreichbarkeit, Lieferzeit, Skonto, Rabatt und Service unterzogen. Für die Leistungspunkte wurde der Bereich eins bis zehn festgelegt, wobei zehn einen wichtigen Punkt darstellt. Die maximal zu erreichenden Punkte liegen ebenfalls in Bereich eins bis zehn.

Der Lieferant BSH – EDV hat innerhalb einer gesetzten Frist kein Angebot eingereicht, daher wird er in der folgenden Auswertung nicht berücksichtigt.

Reichelt	Leistungspunkte	Erreichte Punkte	Ergebnis
Kulanz	7	8	56
Erreichbarkeit	9	9	81
Lieferzeit	8	6	46
Skonto	6	7	42
Rabatt	7	7	49
Service	9	9	81
Summe			**355**

Tabelle 6 Lieferantenvergleich

Mindfactory	Leistungspunkte	Erreichte Punkte	Ergebnis
Kulanz	7	7	49
Erreichbarkeit	9	9	81
Lieferzeit	8	7	56
Skonto	6	6	36
Rabatt	7	6	42
Service	9	8	72
Summe			**336**

Tabelle 7 Lieferantenvergleich

Beide Lieferanten bieten keine erweiterte Gewährleitung für PC – Systeme an. Aus diesem Grund wurde dieser Punkt im Vergleich nicht näher berücksichtigt. Aufgrund der aktuellen Gesetzeslage sind die Lieferanten jedoch zu 2 Jahren Gewährleistung verpflichtet.

Lieferant	Kulanz	Erreichbarkeit	Lieferzeit	Skonto	Rabatt	Service	Summe
Reichelt	56	81	46	42	49	81	**355**
Mindfactory	49	81	56	36	42	72	**336**

Tabelle 8 Übersicht Lieferantenvergleich

Der obigen Tabelle kann entnommen werden, dass der Lieferant Reichelt als bester unten den genannten Kriterien hervorgeht.

4.4.2 Software festlegen

Stückzahl	Bezeichnung	Produkt
1	Betriebssystem	Fedora Core 8
1	Webserver	Apache HTTP Webserver
1	PHP	PHP Version 5
1	Nagios	Nagios Version 3.0
1	NSClient++	Nagios Client

Tabelle 9 Benötigte Software

Für das Projekt werden ausschließlich freie Software – Produkte eingesetzt um die Kosten für die Städtische Datenverarbeitung so gering wie möglich zu halten.

4.5 Prozessschnittstellen festlegen

Durch die bereits durchgeführten Projektpunkte ergeben sich folgende Prozessschnittstellen.

Name	Typ	Firma	Beschreibung
Julian Wilhelm	Projektleiter	Städtische Datenverarbeitung	Projektdurchführer / Auszubildender der Städtischen Datenverarbeitung
Thorsten Breer	Projektbetreuer	Städtische Datenverarbeitung	Projektbetreuer / Ausbilder der Städtischen Datenverarbeitung
Diether Erdelyie	Rechnungswesen / Beschaffung	Städtische Datenverarbeitung	Zuständig für die Beschaffung benötigter Hard- und Software
Frank Gerwarth	Lieferant	Reichelt	Hardwarelieferant

4.6 Kostenplan erstellen

Der Stundenlohn errechnet sich durch die wöchentlichen- Kosten und Arbeitsstunden.

	Ressource	Anzahl	Einheit	Einzelkosten in Euro	Kosten in Euro
1	Personalkosten				
	Auszubildender	35	Stunde	6,45 €	225,75 €
2	Hardwarekosten				
	Nagiosserver	1	Stück	448,36 €	448,36 €
	• Intel 3,2 GHz				
	• Asus P5K				
	• 1 GB DDR2 RAM				
	• Samsung 160 GB				
	• DVD-ROM 16/48x				
	• Coba 500 Watt				
	• Miditower ATX				
3	Softwarekosten				
	Fedora Core 8	1	Stück	0,00 €	0,00 €
	Apache Webserver	1	Stück	0,00 €	0,00 €
	PHP Version 5	1	Stück	0,00 €	0,00 €
	Nagios 3.0	1	Stück	0,00 €	0,00 €
	NSClient++	42	Stück	0,00 €	0,00 €
4	**Gesamt**				**674,11 €**

Tabelle 10 Kostenplan

4.7 Zeitplanung

Nr.	Bezeichnung	Dauer in Stunden
1	IST – Analyse	1.0 Stunden
2	SOLL - Konzept	6.0 Stunden
3	Realisierungsphase	14.0 Stunden
4	Funktionstest	3.5 Stunden
5	Kalkulationsphase	1.0 Stunden
6	Erstellen der Projektdokumentationen	8.5 Stunden
7	Projektübergabe	1.0 Stunden
	Gesamtaufwand in Stunden	**35 Stunden**

Tabelle 11 Zeitplanung

5 Realisierungsphase

Da das Netzwerk der Städtischen Datenverarbeitung ständig zur Verfügung stehen muss, wird Nagios in einer ersten Testphase nicht im Produktiv- sondern in einem kleineren Testnetzwerk installiert.

5.1 Einrichten einer Testumgebung

Um die Kompatibilität von Nagios und insbesondere dem NSClient++ mit dem Netzwerk der Städtischen Datenverarbeitung zu gewährleisten, wurde eine Testumgebung mit identischen Betriebssystemen und Anwendungen eingerichtet. Das Testnetzwerk befindet sich in einem Klasse C Netz und besteht aus folgenden Servern und Komponenten.

IP - Adresse	Hostname	Betriebssystem	Funktion
XXX.XXX.XXX.XXX	nagiosserver	Fedora Core 8	Nagios Monitoring – Server
XXX.XXX.XXX.XXX	linuxclient	Fedora Core 5	Linux Testclient
XXX.XXX.XXX.XXX	server2k3	Windows Server 2003	Server 2003 Testclient
XXX.XXX.XXX.XXX	xpclient	Windows XP mit SP 2	Standard System – Testclient
XXX.XXX.XXX.XXX	lnw1	Windows Server 2003	Lotus Domino Cluster Mitglied 1
XXX.XXX.XXX.XXX	lnw2	Windows Server 2003	Lotus Domino Cluster Mitglied 2
XXX.XXX.XXX.XXX	zyxel	-	Zyxel Netzwerk – Switch
XXX.XXX.XXX.XXX	dns	Windows Server 2003	DNS Server

Tabelle 12 Testumgebung

5.2 Installation und Konfiguration des Monitoringservers

Für Nagios wird ein Server mit Linux Betriebssystem und ein C – Compiler benötigt. Zudem muss TCP/IP am Server konfiguriert werden, da die Überwachungsfunktion über das Netzwerk ausgeübt werden.

Die Städtische Datenverarbeitung setzt bisher die Linux – Distributionen RedHat und Fedora Core ein. Meine Auswahl fällt auf Fedora Core in der Version 8. Im Folgenden wird nur auf die wichtigsten Punkte der Installation eingegangen, eine detaillierte Installationsanleitung ist der Kunden- und Betriebsdokumentation zu entnehmen.

Fedora Core 8 wird mittels Standard – Installationsroutine installiert und eingerichtet. Zusätzlich zum Basissystem wird ein Webserver installiert.

Die im Server vorhandene Festplatte wird so eingerichtet, dass alle Daten auf einer Partition angelegt werden, auf eine separate Partitionierung wird in der Testphase verzichtet.

Um Komplikationen zu vermeiden, wird die interne Firewall und SELinux deaktiviert.

Da der Server über keine Verbindung mit dem Internet verfügt, wird ein lokaler Mirror eingerichtet. Internet – Mirror werden als Paketquellen deaktiviert.

Nagios benötigt für die Installation folgende Pakete, die mit Hilfe des Paketmanagers yum nachinstalliert werden müssen.

Paketname	Funktion
gcc	C - Compiler
glibc	C – Bibliothek
glibc-common	C – Bibliothek Binärdateien
gd	Programmbibliothek für dynamische Grafiken
gd-devel	Programmbibliothek für dynamische Grafiken

Tabelle 13 Benötigte Pakete

Für Nagios wird auf dem System eine eigene Gruppe eingerichtet. Mitglieder dieser Gruppe werden der Webserver- und ein zusätzlich einzurichtender Nagios – Benutzer.

Typ	Name
Gruppe	nagiosgroup
Benutzer	nagios

Tabelle 14 Zusätzliche Gruppen / Benutzer

5.3 Installation von Nagios

Um Nagios installieren zu können, wird die aktuelle Version heruntergeladen und mit einem USB-Stick auf den Server übertragen. Die Nagios – Dateien werden anschließend entpackt.

```
tar xzf nagios-3.0.tar.gz
tar xzf nagios-plugins-1.4.11.tag.gz
```

Nun werden die Nagios – Dateien im Verzeichnis nagios-3.0/ und die Nagios-Plugin Dateien im Verzeichnis nagios-plugins-1.4.11/ kompiliert.

```
./configure --with-command-group=nagiosgroup
make all && make install && make install-init && make install-config
&& make install-commandmode

./configure --with-nagios-user=nagios --with-nagios-
group=nagiosgroup
make && make install
```

Nachdem die Installation abgeschlossen wurde, kann mit der Konfiguration begonnen werden.

5.4 Konfiguration des Apache Webservers

Damit Nagios aufgerufen werden kann, muss der Apache Webserver konfiguriert werden. Hier zu müssen Verweise (Aliase) auf die HTML- und CGI – Dateien von Nagios eingerichtet werden.

Nachdem die Konfiguration vorgenommen und der Webserver neu gestartet wurde, kann Nagios über den Webbrowser über die Adresse http://localhost/nagios/ aufgerufen werden. Der Apache ist zurzeit so eingerichtet, dass jeder Zugriff auf Nagios hat. Im folgenden Punkt wird auf die Implementierung von Sicherheitsmechanismen eingegangen.

Eine Beispielkonfiguration des Webserver ist der Kunden- und Betriebsdokumentation zu entnehmen.

5.5 Implementierung von Sicherheitsmechanismen

Nicht jeder Benutzer soll Zugriff auf die Nagios – Software erhalten, zum Schutz wird diese mit einer Benutzer- und Passwort Authentifizierung gesichert. Mit dem Programm htpasswd wird eine Datei generiert, in der die Benutzernamen und Passwörter mit Zugangsberechtigung zu Nagios gespeichert werden. Das Passwort wird nicht im Klartext, sondern verschlüsselt in der Datei hinterlegt.

Es ist möglich, mehrere Benutzer in der Datei zu hinterlegen, in der Testphase gibt es nur einen zentralen Benutzeraccount.

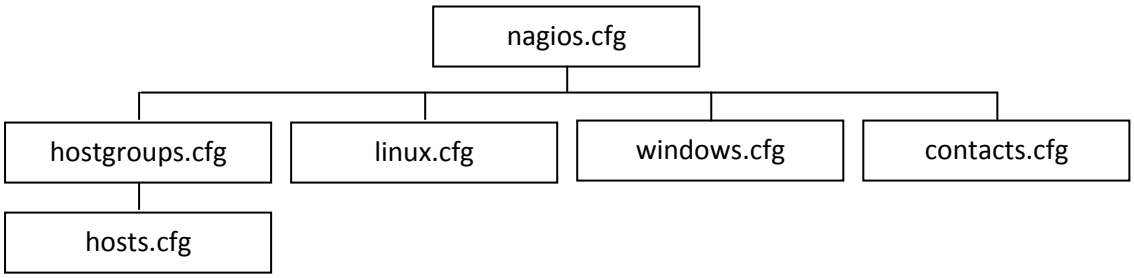
Folgender Befehl wird in der Konsole ausgeführt

```
htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin
```

Nach dem bestätigen des Befehls kommt eine Abfrage in der das Passwort für den Benutzer eingegeben werden muss. Dieses Passwort ist ausschließlich für den Zugriff auf die Nagios –Webseite. Der Webserver liest die Datei aus, sobald die Nagios –Webseite aufgerufen wird.

5.6 Konfiguration von Nagios

Für Nagios 3.0 steht kein Webinterface für die Konfiguration zur Verfügung, aus diesem Grund muss Nagios über die Konfigurationsdateien konfiguriert werden. In der Datei nagios.cfg findet die zentrale Konfiguration statt. Um dort nicht sämtliche Konfigurationsoptionen unterbringen zu müssen, werden diese in verschiedene Konfigurationsdateien aufgeteilt.



5.6.1 Erstellen und bearbeiten der Hostgruppen

Die Hostgruppen – Konfiguration wird benutzt, um einen oder mehrere Hosts in Gruppen zusammen zu fassen. Mit Hilfe von Hostgruppen lassen sich bestimmte Services leichter auf viele Clients im Netzwerk anwenden.

Eine Hostgruppe wird wie folgt Konfiguriert.

```
define hostgroup {

      hostgroup_name              Windowsstandard
      alias                       Standard für Windows Server

}
```

Option	Erklärung
hostgroup_name	Name der Hostgruppe
alias	Beschreibung der Hostgruppe

Tabelle 15 Hostgruppenoptionen

5.6.2 Erstellen und bearbeiten der Hosts

In der Hosts – Datei werden Nagios die einzelnen Hosts bekannt gemacht. Das Host – Objekt ist der zentrale Punkt, auf den alle Host- und Service – Checks basieren. Es definiert den zu überwachenden Rechner oder Netzwerkknoten.

Der Aufbau der Host – Konfiguration ist nahezu bei jedem Host gleich und könnte für jeden neuen Hosteintrag kopiert werden. Die relevanten Zeilen müssten nur angepasst werden.

Ein Host wird Nagios über folgende Deklaration bekannt gemacht.

```
define host {

      host_name                    server2k3
      alias                        Windows Server 2003
      address                      XXX.XXX.XXX.XXX
      hostgroups                   Windowsstandard
      check_command                check-host-alive
      check_interval               1
      max_check_attempts           3
      contact_groups               Windowsadministratoren
      notifications_enabled        1

}
```

Der Aufbau einer Host – Deklaration ist weitestgehend selbsterklärend, folgende Optionen müssen in der Host – Konfiguration enthalten sein.

Option	Erklärung
host_name	Name des Hosts
alias	Beschreibung für den Host
address	IP – Adresse des Hosts
hostgroups	Name der Hostgruppe
check_command	Name der Prüfverfahrens für den Host
check_interval	Intervall für die Hostprüfung (Minuten)
max_check_attempts	Anzahl der Service – Überprüfungen
contact_groups	Name der Kontaktgruppe
notifications_enabled	Benachrichtigung aktiviert

Tabelle 16 Hostoptionen

5.6.3 Erstellen und bearbeiten der Kontaktdateien

In der Kontaktdatei (contacts.cfg) wird festgelegt, wer im Fall eines Problems informiert werden soll. Es lassen sich verschiedene Kontakte in verschiedene Gruppen zusammenfassen. Eine Kontaktgruppe ist wie folgt aufgebaut.

```
define contactgroup {

      contactgroup_name          Windowsadministratoren
      alias                      Gruppe der Windowsadministratoren

}
```

In den Kontaktgruppen befinden sich die einzelnen Kontakte. Ein Kontakt ist wie folgt aufgebaut.

```
define contact {

      contact_name                          nagiosadmin
      alias                                 Nagiosadministrator
      email                                 root
      contactgroups                         Windowsadministratoren

      host_notifications_enabled            1
      host_notification_options             d,u,r,f
      host_notification_commands            notify-host-by-email
      host_notification_period              24x7

      service_notifications_enabled         1
      service_notification_options          w,u,r,f,s
      service_notification_commands         notify-service-by-email
      service_notification_period           24x7

}
```

Jeder Kontakt kann über verschiedene Probleme und Benachrichtigungswege informiert werden.

Option	Erklärung
contact_name	Name des Kontakts
alias	Beschreibung des Kontakts
email	E-Mailadresse des Kontakts
contactgroup	Kontaktgruppe
host_notifications_enabled	Benachrichtigung über Hostprobleme
host_notification_options	Benachrichtigungsoptionen
host_notification_commands	Benachrichtigungstyp
host_notification_period	Zeitraum in dem benachrichtigt werden soll
service_notifications_enabled	Benachrichtigung über Serviceprobleme
service_notification_options	Benachrichtigungsoptionen
service_notification_commands	Benachrichtigungstyp
service_notification_period	Zeitraum in dem benachrichtigt werden soll

Tabelle 17 Kontaktoptionen

Host - Benachrichtigungsoptionen	Erklärung
d	Benachrichtigung bei DOWN Zustand
u	Benachrichtigung bei UNREACHABLE Zustand
r	Benachrichtigung bei UP Zustand
f	Benachrichtigung bei Flattern des Hosts
s	Benachrichtigung bei geplanten Hostausfall
n	Keine Benachrichtigung versenden

Tabelle 18 Host Benachrichtigungsoptionen

Service - Benachrichtigungsoptionen	Erklärung
w	Benachrichtigung bei WARNING Zustand
u	Benachrichtigung bei UNKNOWN Zustand
f	Benachrichtigung bei Flattern des Services
s	Benachrichtigung bei geplanten Serviceausfall
n	Keine Benachrichtigung versenden

Tabelle 19 Service Benachrichtigungsoptionen

5.6.4 Erstellen und bearbeiten der Services

Die Checks die der Nagios – Server überprüfen soll, werden standardmäßig in der Service – Konfigurationsdatei (services.cfg) definiert. Zur besseren Übersicht, wurde diese Datei separiert. Checks für Windows – Systeme befinden sich nun in der windowsservices.cfg und Checks für Linux – Systeme in der linuxservices.cfg.

Nun folgt die Beschreibung eines Services zur Überwachung der CPU – Auslastung eines Windows 2003 Servers.

```
define service {

      hostgroup_name                  Windowsstandard
      service_description             CPU
      check_command                   check_nt!CPULOAD!-l 5,80,90
      max_check_attempts              3
      check_interval                  1
      retry_interval                  1
      check_period                    24x7
      notification_period             24x7

}
```

Option	Erklärung
hostgroup_name	Name der Hostgruppe die Überwacht werden soll
service_description	Beschreibung des Services
check_command	Überprüfungsbefehl
max_check_attempts	Anzahl der Service – Überprüfungen
check_interval	Überprüfungsintervall
retry_interval	Überprüfungsintervall bei Fehlern
check_period	Zeitraum in dem überprüft werden soll
notification_period	Zeitraum in dem benachrichtigt werden soll

Tabelle 20 Serviceoptionen

5.7 Konfigurieren des NSClient++

Nach der erfolgreichen Installation und Konfiguration von Nagios auf dem Nagios – Server wurden die Clients zur Überwachung durch den Nagios – Server vorbereitet. Der NSClient++ wurde vorkonfiguriert und anschließend auf die Client übertragen und installiert. Der NSClient++ lässt sich durch eine INI – Datei konfigurieren, in die der Nagios – Server als erlaubter – Überwachungsserver (Eintrag: allowed_host) eingetragen wird.

Es müssen keine weiteren Einstellungen vorgenommen werden, da der Nagios – Server sämtliche Checks ausführt.

6 Funktionstest

Um Nagios zu testen, werden Dienste beendet, Server heruntergefahren und Ressourcen beansprucht. Sobald einer dieser Fälle eintritt, müsste Nagios eine E-Mail Benachrichtigung an einen Administrator versenden.

Folgende Situationen wurden simuliert.

Funktionstest	Benachrichtigung erhalten / nicht erhalten
Nicht – Erreichbarkeit eines Hosts	Benachrichtigung durch E-Mail erhalten
CPU – Auslastung höher als 80%	Benachrichtigung durch E-Mail erhalten
RAM – Auslastung höher als 80%	Benachrichtigung durch E-Mail erhalten
Ping – Latenzzeit höher als 50 Millisekunden	Benachrichtigung durch E-Mail erhalten
Nicht – Erreichbarkeit eines Dienstes	Benachrichtigung durch E-Mail erhalten
Festplattenspeicher weniger als 20%	Benachrichtigung durch E-Mail erhalten

Tabelle 21 Funktionstest

Die im Rahmen der Testphase durchgeführten Funktionstests sind alle erfolgreich verlaufen. Da innerhalb der Testphase keine Fehler mit Nagios oder dem NSClient++ aufgetreten sind, kann dieser ohne Bedenken auf den Servern im Produktivnetz installiert werden.

Das Webinterface wurde ebenfalls auf Funktionalität geprüft. Drei Administratoren der Städtischen Datenverarbeitung haben sich gleichzeitig auf dem Nagios – Webinterface angemeldet.

7 Nachkalkulation

7.1 Projektkosten

Im der folgenden Kostengegenüberstellung sind die Projektkosten, nach Projektabschluss, aufgelistet.

	Ressource	Anzahl	Einheit	Einzelkosten in Euro	Kosten in Euro
1	Personalkosten				
	Auszubildender	35	Stunde	6,45 €	225,75 €
2	Hardwarekosten				
	Nagiosserver	1	Stück	448,36 €	448,36 €
3	Softwarekosten				
	Linux und Software	1	Stück	0,00 €	0,00 €
4	Raumkosten				
	Testumgebung	5	m²	3,75 €	18,75 €
	Arbeitsplatz	15	m²	3,75 €	56,25 €
5	Sonstige Kosten				
	Strom	25,3	KW/h	0,05 €	1,27 €
	Gesamt				**750,38 €**

Tabelle 22 Projektkosten

7.2 Projektplan

Wie aus dem unten aufgestellten Projektplan zu entnehmen ist, unterscheiden sich die SOLL- und IST – Zeiten in einigen Projektpunkten.

	Projektpunkt	**SOLL Zeit**	**IST Zeit**
1	Ist Analyse		
	• Erfassung vorhandener Server und Dienste	1.0	1.0
2	Soll Konzept		
	• Klärung der detaillierten Kundenwünsche	0.5	0.5
	• Klärung der Anforderungen	0.5	1.0
	• Kosten- / Nutzen-Analyse der Überwachungsanwendungen	1.0	1.5
	• Entscheidungsfindung	1.0	0.5
	• Festlegung der erforderlichen Hardware und Software	1.0	0.5
	• Prozessschnittstellen festlegen	0.5	1.0
	• Gesamtkostenplan erstellen	1.0	1.5
	• Bestellung und Beauftragung	0.5	1.0
3	Realisierungsphase		
	• Installation und Konfiguration des Überwachungsservers	3.0	2.5
	• Konfiguration des Webservers	1.0	1.0
	• Implementierung von Sicherheitsmechanismen	0.5	0.5
	• Erstellen und konfigurieren der Host – Dateien	3.0	2.5
	• Erstellen und konfigurieren der Service – Dateien	3.0	2.5
	• Anpassen des Webinterfaces	1.5	1.0
	• Installation der Client - Überwachungssoftware	2.0	2.5
4	Funktionstest		
	• Testen der Server und Clients	3.5	2.5
5	Kalkulationsphase		
	• Nachkalkulation	1.0	1.0
6	Erstellung der Dokumentationen		
	• Erstellen von Schulungsunterlagen	1.0	1.5
	• Erstellen der Kundendokumentation	1.5	2.0
	• Erstellen der Projektdokumentation	6.0	6.0
7	Projektabschluss		
	• Projektübergabe	1.0	1.0
	Projektzeit in Stunden	**35**	**35**

Tabelle 23 Projektplan

8 Reflektion

Der Projektbetreuer der Städtischen Datenverarbeitung ist mit der Monitoring – Software und der aus dem Projekt entstandenen Erkenntnisse zufrieden. Meiner Empfehlung nach sollte Nagios später mit dem SMS – Gateway der Städtischen Datenverarbeitung verbunden werden, um so noch schneller auf Fehler und Probleme aufmerksam zu machen.

9 Literaturverzeichnis

9.1 Allgemeine Informationen

- Enzyklopädie Wikipedia (http://de.wikipedia.org/wiki/)

9.2 Produktinformationen

- Nagios (http://www.nagios.org)
- Zabbix (http://www.zabbix.com)
- WhatsUp Gold (www.ipswitch.com/international/german/whatsupgold.html)

10 Anhang

10.1 Kunden- und Betriebsdokumentation

Installation Fedora Core 8

Die Installation von Fedora Core 8 wird mit einer DVD durchgeführt. Der PC muss zur Installation von CD / DVD booten.

Das Tastaturlayout und die Sprache werden auf Deutsch gesetzt.

Die Festplatte wird wie vorgegeben so partitioniert, das alles auf einer Partition abgelegt wird.

Als Bootloader wird Grub gewählt.

Die Netzwerkeinstellungen werden wie folgt manuell vorgeben

Rechnername: nagiosserver
IP – Adresse: XXX.XXX.XXX.XXX
Netzmaske: XXX.XXX.XXX.XXX
IP v6 wird deaktiviert und Netzwerkgeräte starten automatisch

Es wird nur der Benutzer root erstellt und ein Passwort vergeben

Benutzer: root
Passwort: ********

Neben dem Standardsystem, wird ein Webserver mit installiert.

Nach der Installation wird ein Systemupgrade durchgeführt. Das Upgrade wird über die Konsole wie folgt durchgeführt und alle Punkte mit YES bestätigt

yum upgrade

Es werden zusätzlich einige Softwarepakete installiert

yum install gcc
yum install glibc glibc-common
yum install gd gd-devel

Installation von Nagios

Für Nagios wird eine neue Gruppe und ein neuer Benutzer erstellt und mit Passwort versehen

useradd Nagios &&passwd Nagios

groupadd Nagiosgroup

In die Gruppe Nagiosgroup wird der neue Benutzer und der Webserver – Benutzer aufgenommen

usermod –G nagiosgroup Nagios
usermod –G nagiosgroup apache

Nagios wird mittels CD auf den Nagiosserver übertragen und in das Verzeichnis /root kopiert.
Anschließend werden beide Pakete entpackt

tar xzf nagios-3.0.tar.gz
tar xzf nagios-plugins-1.4.11.tar.gz

Nach dem entpacken führen wir folgende Befehle im Nagios – Verzeichnis aus

./configure --with-command-group=nagiosgroup
make all && make install && make install-init && make install-config && make install-commandmode && make install-webconf

wurden alle Befehle erfolgreich ausgeführt, wurde Nagios erfolgreich kompiliert. Als nächstes wird in der Nagios Kontaktdatei der Administrator eingetragen

vi /usr/local/nagios/etc/objects/contacts.cfg

In dieser Datei wird im Punkt nagiosadmin die E-Mailadresse geändert.

Im zweiten Verzeichnis werden die Nagios Plugins kompiliert

./configure --with-nagios-user=nagios --with-nagios-group=nagiosgroup
make && make install

Um Zugriff auf das Webinterface zu erlangen, muss der Nagiosadmin in die HTACCESS Datei von Nagios eintragen werden

htpasswd -c /usr/local/nagios/etc/htpasswd.users nagiosadmin

Der Nagios- und HTTPD Dienst werden als Autostart – Dienste eingetragen und aktiviert

chkconfig --add Nagios && chkconfig nagios on

chkconfig --add httpd && chkconfig httpd on

Um alle Funktionen zu testen, wird der Nagiosserver neu gestartet.

Nachdem der Nagiosserver hochgefahren ist, kann der erste Zugriff auf das Webinterface über die Adresse http://localhost/nagios/ getestet werden.

Konfiguration von Nagios

In Verzeichnis /usr/local/nagios/etc/ liegt die nagios.cfg in der die Zentrale Konfiguration von Nagios stattfindet. Um dort nicht alle Konfigurationspunkte aufführen zu müssen, wurde diese in einzelne Dateien aufgeteilt. Der nagios.cfg werden diese wie folgt bekannt gemacht

cfg_file=/usr/local/nagios/etc/objects/commands.cfg
cfg_file=/usr/local/nagios/etc/objects/contacts.cfg
cfg_file=/usr/local/nagios/etc/objects/timeperiods.cfg
cfg_file=/usr/local/nagios/etc/objects/templates.cfg
cfg_file=/usr/local/nagios/etc/objects/hostgroups.cfg
cfg_file=/usr/local/nagios/etc/objects/windowsservices.cfg
cfg_file=/usr/local/nagios/etc/objects/linuxservices.cfg
cfg_file=/usr/local/nagios/etc/objects/hosts.cfg

Die kompletten Konfigurationsdateien befinden sich auf einem USB – Stick, welcher der Städtischen Datenverarbeitung am Ende des Projektes übergeben wurde.

Nachdem Nagios und der Apache erfolgreich konfiguriert wurde, wird Nagios darauf geprüft ob es im späteren Betrieb fehlerfrei läuft

/usr/local/nagios/bin/nagios –v /usr/local/nagios/etc/nagios.cfg

Die in der Testumgebung aufgeführten Server werden auf Erreichbarkeit, CPU – Auslastung, freie Speicherkapazitäten und Ping – Reaktionszeit mit Nagios überwacht.

Konfiguration des NSClient++

Der NSClient++ befindet sich ebenfalls auf dem USB – Stick, welcher der Städtischen Datenverarbeitung ausgehändigt wurde. Zur Installation muss die erstellte Batchdatei (install.bat) ausgeführt werden. Nach der Installation kann mit der Überwachung begonnen werden.