Holm Schwantner

Monitoring heterogener Systeme am Beispiel des IT-Systems im “Das TIETZ”

DIPLOMARBEIT

HOCHSCHULE MITTWEIDA (FH)

UNIVERSITY OF APPLIED SCIENCES

Fachbereich Elektro- und Informationstechnik

Mittweida, Mai 2014

Holm Schwantner

Monitoring heterogener Systeme am Beispiel des IT-Systems im “Das TIETZ”

eingereicht als

DIPLOMARBEIT

an der

HOCHSCHULE MITTWEIDA (FH)

UNIVERSITY OF APPLIED SCIENCES

Fachbereich Elektro- und Informationstechnik

Mittweida, Mai 2014

Erstprüfer: Prof. Dr.-Ing. Thomas Beierlein
Zweitprüfer: Dipl.-Kffr. Sibylle Löwe

Vorgelegte Arbeit wurde verteidigt am:

Bibliographische Beschreibung:

Holm Schwantner:
Monitoring heterogener Systeme am Beispiel des IT-Systems im “Das TIETZ”
2014. - 71 S. Mittweida,
Hochschule Mittweida (FH) - University of Applied Sciences,
Fachbereich Elektro- und Informationstechnik, Diplomarbeit, 2014

Referat:

Ein störungsfreier Betrieb der Informationstechnik in Unternehmen und Einrichtungen ist heute Voraussetzung für deren Betriebsfähigkeit. Monitoring ist eine bewährte Vorgehensweise in der Systemadministration, um das zu bewerkstelligen.

Diese Arbeit beschäftigt sich mit der Auswahl und Untersuchung eines Monitoringsystems für die Überwachung der heterogenen IT-Infrastruktur eines städtischen Eigenbetriebes. Mit Methoden der Anforderungsanalyse werden zunächst alle Komponenten identifiziert und klassifiziert, die in das Monitoring übernommen werden sollen. Mit Icinga wird eine für das Monitoring der vorgefundenen Infrastruktur sinnvolle Open-Source-Lösung ausgewählt und anschließend eine mögliche Umsetzung in ein funktionierendes Monitoringsystem unter der Anwendung von Standards gezeigt.

Inhaltsverzeichnis

Abbildungsverzeichnis

Abkürzungsverzeichnis

BYOD	Bring your own Device
DHCP	Dynamic Host Configuration Protocol
DNS	Domain Name Service
FTP	File Transport Protocol
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
IT	Informationstechnik
ITIL	IT Infrastructure Library
LAMP	Linux Apache MySQL PHP
LAN	Local Area Network
LDAP	Lightweight Directory Access Protocol
MIB	Management Information Base
NAS	Network Attached Storage
NRPE	Nagios Remote Plugin Executor
NSCA	Nagios Service Check Acceptor
NTP	Network Time Protocol
PC	Personalcomputer
PHP	PHP: Hypertext Preprocessor
SMI	Structure of Management Information
SMTP	Simple Mail Transfer Protocol
SNMP	Simple Network Management Protocol
SSH	Secure Shell
SSL	Secure Socket Layer
USV	unterbrechungsfreie Stromversorgung
WAN	Wide Area Network
WLAN	Wireless LAN

1 Einleitung

1.1 Motivation

“DasTIETZ”, das sind die drei Einrichtungen Stadtbibliothek, Volkshochschule und Museum für Naturkunde zusammengefasst als kommunaler Eigenbetrieb. Gemeinsam mit der Neuen Sächsischen Galerie sind sie unter dem Dach des einstigen TIETZ-Kaufhauses in Chemnitz untergebracht.

Seit der Gründung des Eigenbetriebs “Das TIETZ” vor 10 Jahren ist Informationstechnik, kurz IT, in alle Bereiche der Einrichtung, besonders in den Geschäftsbetrieb, vorgedrungen und zu einem unternehmenskritischen Faktor geworden. Ein störungsfreier IT-Betrieb ist inzwischen Voraussetzung für die Betriebsfähigkeit des Eigenbetriebes.

Diese Entwicklung wird sich fortsetzen. Aktuelle Trends wie Virtualisierung, Cloud Computing oder BYOD (“Bring your own Device”) sind, obwohl bisher noch nicht durch konkrete Projekte eingeführt, bereits im Eigenbetrieb angekommen und erfordern Beachtung.

Während die Anforderungen an den IT-Betrieb ständig zunehmen, ist die Anzahl der Beschäftigten in der IT-Betreuung bisher gleich geblieben. Das wird sich aufgrund der Situation im öffentlichen Bereich auch in nächster Zeit nicht ändern. Dazu kommt die ständig zunehmende Vielfalt und Heterogenität der IT-Landschaft, die im “Das TIETZ” durch zwei Systemadministratoren betreut wird.

Ein Ansatz zur Lösung ist der Einsatz automatisierter Tools zur Administration und Überwachung. Nach (Des12) kann ein Systemadministrator durch manuelle Administration elf UNIX-Server oder 30 Windows-Server betreuen. Durch Automatisierung läßt sich dieser Anteil wesentlich erhöhen. Bei (Pes12) wird die Automatisierung des IT-Betriebs als wesentliche Voraussetzung für kommende Herausforderungen beschrieben: “Man kann gerade durch Automatisierung einen schnellen Mehrwert erhalten ...” (S. 8). Im Eigenbetrieb kommt zur Betreuung der Server noch die von Netzwerk, Arbeitsplatztechnik und Fachverfahren hinzu, außerdem sind Helpdesk und Provisioning abzudecken.

Eine Einführung von werkzeuggestützten Verfahren für Überwachung und Betrieb des IT-Systems dient also zunächstx der Erleichterung der eigenen Arbeit.

1.2 Idee

Bereits vor einigen Jahren bestand in der Systemadministration im "Das TIETZ" die Idee, ein Monitoringsystem zu etablieren. Dazu wurden bereits einige Vorüberlegungen getroffen (Sch10) und für Testzwecke eine Appliance mit einem vorkonfigurierten Monitoringsystem eingerichtet. Bei der Beschäftigung mit dem System stellte sich jedoch schnell heraus, dass die Schwierigkeiten an anderer Stelle liegen.

Während die Installation relativ schnell erledigt ist, da viele Linux-Distributionen bereits fertige Pakete enthalten und gute Anleitungen für deren Installation existieren, stellt die Anpassung des Systems an die spezifischen Erfordernisse des jeweiligen IT-Systems und die Überführung in den produktiven Betrieb die eigentliche Herausforderung dar.

Spezielle Randbedingung ist eine relativ kleine, dafür sehr heterogene Umgebung und damit eine Vielfalt an zu überwachenden Objekten, die über das bloße Überwachen und Kontrollieren von Netzwerkgeräten und Servern hinausgeht.

Bereits Kurt Tucholski beschrieb 1930 diesen Konflikt in seinem Gedicht "Danach"[1]:

> Es wird nach einem Happy-end
> im Film jewöhnlich abjeblendt.
> Man sieht bloß noch in ihre Lippen
> den Helden seinen Schnurrbart stippen –
> da hat sie nu den Schentelmen ...
> Na, un denn –?
> ...

Mit der Entscheidung für ein Monitoringssystem ist das Problem also noch nicht gelöst. Genauso wichtig ist es, herauszufinden, was überwacht wird und wie. Nach der Installation beginnt somit die eigentliche Arbeit. Damit die Vorteile aus dem Einsatz automatisierten Monitorings nicht wieder aufgehoben werden, ist es notwendig, diese Schritte werkzeuggestützt durchzuführen oder wenn möglich, zu automatisieren.

1.3 Ziel

Aufgabe ist die Inbetriebnahme eines Open-Source-Monitoringsystems im IT-Bereich des Eigenbetriebs "Das TIETZ", die Konzeption geeigneter Parameter und eine Einrichtung vorrangig im Bereich Server und Netze.

Hauptziel bei der Einführung des Monitoringsystems ist, das Prinzip Agieren statt Reagie-

[1] Theobald Tiger in: Die Weltbühne, 01.04.1930, 26/1, Nr. 14, S. 517, wieder in: Lerne Lachen.

ren zu etablieren (Sch10). Damit soll vor allem folgendes erreicht werden:

- ein stabiler, planbarer Systembetrieb,
- die Vorbeugung von Störungen,
- eine Vermeidung unvorhergesehener Kostenspitzen und
- eine Erhöhung der Systemsicherheit.

Das Ziel dieser Arbeit ist es, ein exemplarisches Monitoringsystem zu untersuchen, um die Tauglichkeit solcher Systeme für den spezifischen Einsatz in der IT-Landschaft des TIETZ besser beurteilen zu können.

Dabei sind folgende Fragen zu beantworten:

- Wie kann die IT-Landschaft im TIETZ mithilfe eines Open-Source-Monitoringsystems überwacht werden?
- Welche Anforderungen bestehen an die Funktionsfähigkeit des IT-Systems?
- Welche Geräte und Dienste sind am besten geeignet, den Systemzustand der IT darzustellen?
- Welche Parameter sind für die Überwachung der ausgewählten Geräte und Dienste geeignet?

Folgende Ergebnisse sollen erreicht werden:

- Analyse des IT-Systems auf Monitorbarkeit
- Auswahl von Hilfsmitteln und Werkzeugen
- Kategorisierung, Klassifizierung, Einordnung der einzelnen Elemente des IT-Systems
- Aufstellen eines Workflows zur Einrichtung von Monitoringsystem und Hosts
- Automatisierung der Konfigurationserstellung

1.4 Überblick

Im Kapitel 2 werden notwendige Grundlagen und Begriffe zu Monitoring und Netzwerk beschrieben.

Im Kapitel 3 wird das IT-System im TIETZ qualitativ und quantitativ dargestellt und eine entsprechende Einordnung vorgenommen. Es werden Anforderungen an das Monitoringsystem aufgestellt und festgelegt, was überwacht werden soll. Die zu diesem Zweck durchgeführte Anforderungsanalyse und die daraus entstandene Klassifizierung der gefundenen Objekte werden näher vorgestellt.

Im Kapitel 4 wird die Wahl des Monitoringssystems begründet und dieses beschrieben. Die sich aus der genaueren Untersuchung des Monitoringsystems abgeleiteten Richtlinien, Abläufe und Vorlagen für eine automatisierte Erstellung der Konfiguration werden vorgestellt. Anschließend werden bisherige Erfahrungen erörtert und die Umsetzung der Anforderungen ausgewertet.

Im Kapitel 5 werden die Ergebnisse zusammengefasst und ein Ausblick auf weiterführende Arbeiten gegeben.

2 Grundlagen

In diesem Kapitel werden einige Grundlagen und Begriffe erläutert, auf die sich diese Arbeit stützt und die für ein besseres Verständnis der kommenden Ausführungen nützlich sind. Dazu wird zuerst kurz auf Aufbau und Funktion der hier betrachteten Rechnernetze eingegangen und danach der Zusammenhang mit Monitoring hergestellt. Schließlich werden einige Begriffe erläutert, die das Monitoring von IT-Infrastrukturen betreffen.

2.1 Rechnernetze

Als in der Stadtbibliothek 1992 der erste Rechner angeschafft wurde, um Ausleihe und Katalogisierung, also die Kernbereiche bibliothekarischen Handelns, mit Computern oder wie wir heute sagen würden, IT[1] zu unterstützen, gab es bereits eine Form von Vernetzung. Sie diente dazu, die einzelnen Bildschirme an den Arbeitsplätzen über Zweidrahtleitungen mit dem zentralen Rechner zu verbinden und hatte noch nichts mit den heute verwendeten Netzen zu tun.

Inzwischen stehen an den Arbeitsplätzen in der Stadtbibliothek und den anderen Einrichtungen des TIETZ keine bloßen Bildschirme mehr, sondern Rechnersysteme, von denen jedes einzelne so leistungsfähig ist wie der damalige zentrale Rechner. Geblieben ist die Notwendigkeit, zwischen den einzelnen Arbeitsstationen Nachrichten auszutauschen. Für diesen Zweck werden heute lokale Netze verwendet. Lokale Netze oder LAN[2] sind die Grundlage der IT-Infrastruktur an einem bestimmten Standort wie zum Beispiel im TIETZ und bilden im Rahmen einer Systemarchitektur immer ein Kommunikations-Subsystem mit hohem Integrationsgrad in die einzelnen Systeme (Kau03). Über Weitverbundnetze oder WAN[3] werden lokale Netze miteinander verbunden und ermöglichen so Verbindungen zwischen Rechnern in verschiedenen lokalen Netzen.

Das lässt für den einzelnen Nutzer "das Netz" als großes Ganzes erscheinen, was unter anderem daran liegt, dass die Rechner in den verschiedenen Netzen in gleicher Weise miteinander kommunizieren. Die Kommunikation basiert auf Technologien, die zuerst im Internet

[1] IT: Informationstechnik (Oberbegriff für Informations- und Datenverarbeitung)
[2] LAN: Local Area Network
[3] WAN: Wide Area Network

verwendet wurden. Das Internet[4] ist also in Wirklichkeit kein einzelnes Netz, sondern besteht aus miteinander verbundenen lokalen Netzen, die auf der TCP[5]/IP[6]-Protokollfamilie beruhen und miteinander durch Backbones verbunden sind. Für diese Verbindungen werden zum Teil auch andere Technologien und Protokolle verwendet, auf die hier aber nicht weiter eingegangen wird (Win09).

2.1.1 Begriffe

Allgemein wird unter einem Kommunikationsnetz ein Transportsystem für transformierte Nachrichten verstanden. Die an der Kommunikation teilnehmenden Kommunikationspartner bestehen aus End- und Netzeinrichtungen, es wird auch von Stationen und (Netz-)Knoten gesprochen (Win09). Endeinrichtungen können der Rechner am Arbeitsplatz oder der Server im Rechnerraum oder Internet sein. Netzeinrichtungen sind alle Geräte, die für die Funktion des Netzwerkes benötigt werden, wie zum Beispiel Router oder Switche.

Für die Verbindung zwischen End- und Netzeinrichtungen unter- und miteinander gibt es verschiedene Verbindungsstrukturen. Diese werden als Netzwerktopologie bezeichnet, wichtige Topologien sind Ring, Stern, Baum oder Bus. In größeren Netzen wie zum Beispiel im TIETZ werden aus geographischen und organistorischen Gründen Mischstrukturen daraus eingesetzt. Dabei kann es Unterschiede in der physikalischen und logischen Struktur geben, ein LAN kann physikalisch ein Stern und logisch ein Ring sein.

Aus verschiedenen Gründen, zum Beispiel wenn es Datenschutz oder IT-Sicherheit erfordern, können lokale Netze als geschlossene Einheiten angelegt sein. Es besteht dann keine Verbindung zu anderen Netzen. Falls doch bestimmte Verbindungen nach außen benötigt werden, werden an den Verbindungsstellen zu anderen Netzen spezielle Router, sogenannte Firewalls eingesetzt, die den Datenverkehr filtern und nur bestimmte Verbindungen zulassen. Insbesondere für die Anbindung des eigenen lokalen Netzes an die Außenwelt (“das Internet”) werden zusätzliche Zwischennetze als Puffer mit Firewallroutern an den Übergangsstellen eingerichtet, um das eigene Netz besser zu schützen. Das verkompliziert die Netzwerkstruktur zusätzlich und hat Auswirkungen auf die Erreichbarkeit der angeschlossenen End- und Netzeinrichtungen.

Als Server werden im allgemeinen Rechnersysteme verstanden, die als wesentliche Aufgabe Dienste für andere zur Verfügung stellen. Die Rechner sind meist für diesen Verwendungszweck entsprechend ausgestattet und es werden spezielle Serverbetriebssysteme und Software verwendet. Host ist eine ältere Bezeichnung für Server und stammt noch aus der Zeit der Großrechentechnik. Der Begriff Server hat in der Informatik noch eine weitere Bedeu-

[4] von Interconnected Networks
[5] TCP: Transmission Control Protocol
[6] IP: Internet Protocol

tung, dort wird eine Software so bezeichnet, die einen Dienst bereitstellt. Im folgenden wird der Begriff aber nur in der ersten Bedeutung verwendet.

Ein Dienst oder Service bezeichnet die Bereitstellung einer bestimmten Funktionalität durch eine Einrichtung über eine genau definierte Schnittstelle. Es werden Teledienste wie Fernsprechen, WWW[7] oder E-Mail, Übertragungsdienste wie leitungs- oder paketvermittelter Dienst und Zusatzdienste wie Makeln unterschieden (Win09).

Um die Kommunikation zwischen den Einrichtungen zum Beispiel bei der Nutzung von Diensten zu ermöglichen, müssen bestimmte Vereinbarungen bezüglich Syntax, Semantik und Synchronisation des Nachrichtenaustausches getroffen werden. Diese Vereinbarungen werden als Protokoll zusammengefasst. Meist ist für eine Kommunikation das Zusammenspiel mehrerer Protokolle erforderlich. Die einzelnen Protokolle erfüllen dabei Aufgaben auf verschiedenen Ebenen, die als Schichtenmodell dargestellt werden können. Ein grundlegendes Modell ist das OSI[8]/ISO[9]-Schichtenmodell, von dem sich das im Internet benutzte TCP/IP-Referenzmodell ableiten lässt.

2.1.2 Schichtenmodell

Das OSI/ISO-Referenzmodell ist die abstrakte Definition eines Kommunikationssystems, reale Implementierungen auf der Basis eines Modells bilden ein offenes, also miteinander kompatibles System. Das Modell beruht auf einer Schichtenhierarchie, jede Schicht realisiert bestimmte Dienste, die sie der darüber liegenden Schicht zur Verfügung stellt und selbst die Dienste der darunter liegenden Schicht nutzt. Die Dienste jeder Schicht werden durch Protokolle realisiert. Der Schichtenaufbau wird auch als Stack bezeichnet. Im OSI/ISO-Modell werden sieben Schichten unterschieden (Win09):

Schicht 7 – Anwendungsschicht stellt Dienste für Anwendungen zur Anforderung, Bereitstellung oder Änderung von Daten oder für Operationen auf entfernten Systemen zur Verfügung. Verwendete Protokolle sind unter anderem SMTP[10], POP[11] oder IMAP[12] für die Anwendung E-Mail oder HTTP[13] und HTTPS[14] für die Anwendung WWW[15]. Wichtige Dienste auf dieser Ebene sind außerdem DHCP[16] und DNS[17].

[7] WWW: World Wide Web
[8] OSI: Open Systems Interconnection Model
[9] ISO: International Organization for Standardization
[10] SMTP: Simple Mail Transfer Protocol
[11] POP: Post Office Protocol
[12] IMAP: Internet Message Access Protocol
[13] HTTP: Hyper Text Transfer Protocol
[14] HTTPS: Hyper Text Transfer Protocol Secure
[15] WWW: World Wide Web
[16] DHCP: Dynamic Host Configuration Protocol
[17] DNS: Domain Name Service

Schicht 6 – Präsentationsschicht bietet Dienste zum Datenaustausch an.

Schicht 5 – Sitzungsschicht bietet Dienste zur Regelung der Kommunikation und verwendet dafür Sitzungsprotokolle.

Schicht 4 – Transportschicht stellt Dienste für den Ende-zu-Ende-Datentransfer bereit. Wichtige Transportprotokolle sind TCP (nummerierter, sicherer Datentransport) und UDP[18] (nichtquittierter Austausch).

Schicht 3 – Netzwerkschicht stellt Mittel zur Herstellung netzweiter Verbindungen bereit, um Verbindungsabschnitte aus der Schicht 2 logisch zusammenzuschalten und so einen Datentransport zwischen Endeinrichtungen der Schicht 3 zu ermöglichen. Dafür werden Adressierungsmechanismen verwendet, die Wegewahl und Routing ermöglichen. Protokolle in dieser Schicht sind IP, ICMP[19] und ARP[20].

Schicht 2 – Datenübertragungsschicht stellt Mittel zur Adressierung im lokalen Bereich zur Verfügung, um Mehrfachzugriff oder mehrere logische Verbindungen über eine physikalische Verbindung zu realisieren. Protokolle in dieser Schicht sind MAC[21] und LLC[22].

Schicht 1 – Bitübertragungsschicht bietet Dienste zur Bitübertragung über ein physikalisches Medium an, zum Beispiel über Drahtleitungen, Lichtwellenleiter oder Funk.

Die heutigen Kommunikationsstacks werden meist nicht vollständig abgebildet, aber nach diesen Regeln strukturiert und entwickelt. Das TCP/IP-Referenzmodell fasst verschiedene Schichten zusammen und enthält nur vier Schichten:

Schicht 4 – Anwendungsschicht fasst Sitzungs-, Präsentations- und Anwendungsschicht zusammen, umfasst alle Protokolle für Anwendungen

Schicht 3 – Transportschicht ist die Transportschicht, verwendet die Protokolle TCP und UDP

Schicht 2 – Internetschicht entspricht der Vermittlungsschicht, Kern ist das IP-Protokoll

Schicht 1 – Netzzugangsschicht vereint Sicherungs- und Bitübertragungsschicht, nutzt Protokolle zur Datenübertragung wie Ethernet oder 802.11 (WLAN[23])

[18] UDP User Datagram Protocol
[19] ICMP: Internet Control Message Protocol
[20] ARP: Adress Resolution Protocol
[21] MAC: Media Access Control
[22] LLC: Logical Link Control
[23] WLAN: Wireless LAN

2.2 Monitoring

Das Management von Netzwerken hat bereits eine lange Geschichte. Netzwerke waren früher nicht so allgegenwärtig wie heute, es gab sie vor allem in großen Unternehmen und Einrichtungen. Damit bestand die Notwendigkeit, eine große Anzahl von Netzeinrichtungen zu steuern und zu überwachen. Das war relativ einfach, da in solchen Netzen zwar viele Informationen zu verarbeiten, diese jedoch einfach strukturiert waren. Die angeschlossenen Endeinrichtungen waren meist Hosts, die eine gute Unterstützung für das Management boten. Die wesentlichen Standards für das Netz-Management sind SNMP[24] und RMON[25] (Kau03). Die Aufgaben des Netz-Managements werden in einem von der ISO entwickelten Modell wie folgt beschrieben:

- Fehlermanagement: Prophylaxe, Erkennung und Behebung von Fehlern
- Konfigurationsverwaltung: Planung, Erweiterung und Änderung sowie Pflege der Konfiguration
- Abrechnungsmanagement: Erfassung der Netzbenutzung für die Abrechnung
- Leistungsmanagement: Messung und Verbesserung des Leistungsverhaltens
- Sicherheitsmanagement: Authentifizierung und Autorisierung

Mit dem Einzug des Personalcomputers und der TCP/IP-basierten lokalen Netze in die Büros und Wohnzimmer verschieben sich die Verhältnisse. An jedem Arbeitsplatz steht ein netzwerkfähiges Gerät und während die Zahl der vernetzten Geräte steigt, sind im Verhältnis zu den Netzeinrichtungen immer mehr Endeinrichtungen zu verwalten. Damit ergibt sich einerseits weiterhin die Notwendigkeit, diese Netze zu verwalten, andererseits bietet sich auch die Möglichkeit dazu. Aus Netz-Management entwickelt sich Monitoring.

Deutlich wird das beispielsweise an der Nutzung der Begriffe Host und Service. Diese werden zwar nach wie vor benutzt, bekommen aber einen erweiterten Kontext. Als Host werden beim Monitoring alle Einrichtungen bezeichnet, die an das Netzwerk angeschlossen sind. Unter Service wird nicht nur die Bereitstellung einer Funktionalität im Netz verstanden, sondern es werden auch nicht netzwerkspezifische Leistungen oder Kennwerte, wie zum Beispiel Temperatur oder das Auftreten von Ereignissen einbezogen.

Die bisher für das Netz-Management eingesetzten Protokolle werden vor allem bei den Netzeinrichtungen im professionellen Bereich weiterhin verwendet. Darüber hinaus sind aber mit der allgemeinen Vernetzung für das Management netzfähiger Geräte neue Standards entstanden. Diese sind meist herstellerspezifisch und verdrängen teilweise die bisher verwendeten Standards. So hat Microsoft angekündigt, SNMP in der kommenden Version

[24] SNMP: Simple NetworkManagement Protocol
[25] RMON: Remote Monitoring

seines Serverbetriebssystems nicht mehr einzusetzen[26]. Geräte für den Massenmarkt enthalten im Gegensatz dazu oft gar keine Möglichkeit für das Netz-Management mehr.

2.2.1 SNMP

SNMP kommt aus dem TCP/IP-Umfeld und ist für das Management von Netzeinrichtungen nach wie vor ein verbreiteter Standard. Das Konzept von SNMP basiert auf einem SNMP-Manager als Verwalter und SNMP-Agenten auf den zu überwachenden Geräten. Die Kommunikation erfolgt mittels SNMP-Nachrichten über UDP und IP. Diese läuft recht einfach ab: der Manager führt Objekt-Management-Operationen auf den SNMP-Managed-Objects aus. Das sind Parametersammlungen aus der Management Information Base (MIB). Die MIB ist eine Art virtuelle Datenbasis für Managed-Objects und listet alle Informationen, die von einem Agenten abgefragt werden können sowie die Pointer zum Auffinden. Form und Speicherung der Informationen sind herstellerspezifisch und nicht vorgeschrieben, Syntax und Semantik der Managed-Objects sind jedoch zu vereinbaren. Neben allgemein verbindlichen Objekten können auch nicht standardisierte Objekte eingefügt werden. SNMP ist also ein Oberbegriff für:

- das Protokoll SNMP
- die Structure of Management Information (SMI), in der Regeln für die Struktur und die Notation der Management Information Base (MIB) festgelegt sind
- die MIB, in der die zu verwaltende Information beschrieben ist

Für die Kommunikation zwischen Manager und Agent gibt es folgende Protokollelemente:

- GET-REQUEST, GETNEXT-REQUEST
- SET-REQUEST
- GET-RESPONSE
- TRAP

Request-Kommandos werden vom SNMP-Manager gesendet, Response vom Agenten. Trap ist eine Möglichkeit für den Agenten, sich beim Manager bemerkbar zu machen (Kau03).

Die SNMP-Protokollelemente lassen sich unter Linux mit Kommandos aus dem Paket Net-SNMP verwenden. Ein linuxbasierter Rechner kann damit als SNMP-Manager eingesetzt werden (MS05).

Problematisch ist, dass die Standard-MIB für viele Hersteller oft wenige für die Verwaltung ihrer Geräte interessante Elemente enthält und so jeder Hersteller seine eigenen Definitionen

[26]URL: `http://technet.microsoft.com/en-us/library/hh831568.aspx`

erstellt hat. Das macht eine einheitliche Verwaltung recht schwierig.

RMON ist eine Erweiterung der MIB und bietet eine gegenüber SNMP erweiterte Möglichkeit, Daten abzufragen.

2.2.2 IPMI

IPMI[27] ist eine Spezifikation von standardisierten Schnittstellen zur Verwaltung von Servern, die von mehreren Herstellern entwickelt wurde[28]. Die Schnittstellen bieten einerseits die Möglichkeit, die Hardware zu überwachen und Fehlerzustände zu protokollieren als auch den Server neu zu starten. Die Schnittstellen funktionieren auch bei ausgeschaltetem Server, solange eine Standbyversorgung vorhanden ist. IPMI kann über das lokale Netz benutzt werden und Fehler über SNMP senden. Moderne Server haben dafür einen speziellen Anschluss.

2.2.3 WMI

WMI[29] ist eine Schnittstelle für die Administration und Fernwartung von windowsbasierten Rechnern[30]. Es handelt sich dabei um eine herstellerspezifische Implementierung von Standardfunktionen des Common Information Models. Über WMI kann sowohl lesend als auch schreibend auf viele Windowseinstellungen zugegriffen werden. Der Zugriff erfolgt lokal oder über das lokale Netz. Die Verwendung von WMI für das Management von windowsbasierten Systemen anstelle von SNMP wird von Microsoft dringend angeraten.

[27] IPMI: Intelligent Platform Management Interface
[28] URL: `http://www.intel.de/content/www/de/de/servers/ipmi/ipmi-home.html`
[29] WMI Windows Management Instrumentation
[30] URL: `http://msdn.microsoft.com/de-de/library/dn151197.aspx`

3 Anforderungen

In diesem Kapitel werden die Anforderungen an ein Monitoringsystem aus verschiedener Sicht betrachtet. Zuerst wird ein Überblick über die Infrastruktur im “Das TIETZ” gegeben und dann einige allgemeine Forderungen aufgestellt. Mit der Darstellung der durchgeführten Anforderungsanalyse werden die Anforderungen schließlich konkreter untersetzt.

3.1 Darstellung IT im “Das TIETZ”

3.1.1 Überblick

Die IT-Infrastruktur ist 2004 durch die Zusammenlegung der drei Einrichtungen Museum für Naturkunde, Stadtbibliothek Chemnitz und Volkshochschule Chemnitz im Eigenbetrieb “DasTIETZ” entstanden. Aufgrund der Übernahme der gesamten Informationstechnik aus der Stadtverwaltung ist sie auch heute noch grundsätzlich an deren IT-Landschaft orientiert.

Mit der Erstausstattung bei der Gründung des TIETZ wurde die Grundlage für eine gemeinsame Arbeitsweise geschaffen. Inzwischen sind neue Technik und Verfahren dazu gekommen, alte Technik (Personalcomputer, Server) wurde abgelöst. Die sowohl horizontal als auch vertikal ursprünglich sehr heterogene Struktur konnte in den letzten Jahren etwas verringert werden. Allerdings wird es durch die Ausschreibungspflicht im öffentlichen Sektor und die kurze Produktlebenszyklen der Hersteller auf Geräteebene immer eine gewisse Bandbreite an verschiedenen Geräten geben. Aus vertikaler Sicht hat sich die Struktur nicht verändert, im TIETZ werden die wichtigsten Dienste selbst erbracht, so dass ein kompletter Rechenzentrumsbetrieb vorliegt.

3.1.2 hardwaretechnische Infrastruktur

Die gegenwärtige IT-Infrastruktur im TIETZ wird auf der Abbildung 3.1 vereinfacht dargestellt. Die einzelnen Bestandteile der gegenwärtigen IT-Infrastruktur werden im folgenden grob beschrieben und systematisiert.

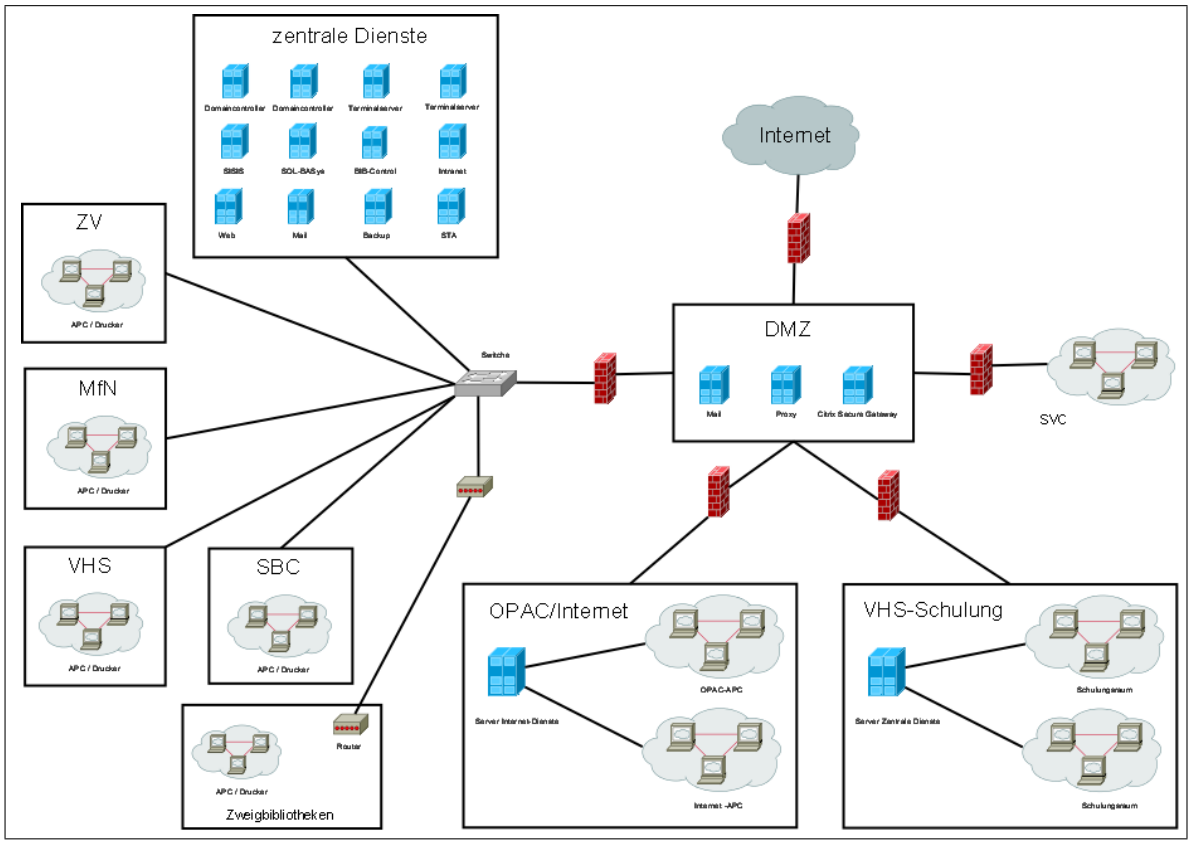

Bild 3.1: IT-Struktur im TIETZ

Server

Für jeden Anwendungsfall werden dedizierte Server eingesetzt. Durch den Einsatz von Virtualisierung ist dafür nicht mehr jedesmal ein eigener physischer Server notwendig. Damit konnte in den letzten Jahren eine Entkopplung von Diensten bei gleichzeitiger Reduzierung der Anzahl der physischen Servern erreicht werden, dafür stieg die Zahl der (virtuellen) Servern.

Die Server laufen je nach Anwendung unter Windows Server oder SuSE Linux, auf den Virtualisierungshosts kommt VMware ESXi zum Einsatz. Insgesamt werden jetzt noch 16 physische Server betrieben, davon sind 10 Virtualisierungshosts. Darauf laufen insgesamt 31 Server, entweder direkt oder virtualisiert, davon 19 mit Windows und 12 mit Linux.

Die Server sind auf Prozessor-, Speicher- und Plattenauslastung sowie auf Fehlerzustände zu überwachen.

Personalcomputer

Zur Zeit werden im TIETZ insgesamt etwa 250 Arbeitsplatz-PCs[1] eingesetzt, sowohl im Mitarbeiter- als auch im Benutzerbereich. Den Benutzerbereich gibt es vorrangig in der Stadtbibliothek und in der Volkshochschule. Es wird sowohl herkömmliche PC-Technik als auch ThinClient-Technik (“IGEL”) eingesetzt. Durch den direkten Kontakt mit dem Benutzer werden Fehler recht schnell erkannt und gemeldet.

Netzwerk

Es gibt im Haus verschiedene logisch und physisch voneinander getrennte Netze, u. a. für Mitarbeiter, für Leser der Stadtbibliothek oder Schüler in der Volkshochschule. Alle Netze nutzen die strukturierte Verkabelung im Haus und werden über HP-ProCurve-Switche verwaltet und verteilt.

Neben dem drahtgebundenen Netz gibt es WLAN mit im Haus verteilten 25 Access-Points, die eine Abdeckung im ganzen Haus ermöglichen. WLAN wird vor allem in der Volkshochschule zur Vernetzung der mobilen Klassenzimmer, sogenannten Net Education Center, genutzt. Leser der Stadtbibliothek können WLAN mit privaten Geräten nutzen. Für Veranstaltungen wird Internetzugang in der Regel über WLAN bereit gestellt.

Das lokale Netz ist redundant über zwei Standleitungen zu Kernnetzknoten des Deutschen Forschungsnetzes an das Internet angebunden. Die Anbindung für das WLAN wird davon getrennt über einen Anschluss mit DSL-Technik realisiert.

Die gesamte Netzinfrastruktur bildet das Rückgrat des IT-Systems und ist auf Verfügbarkeit, Auslastung und Fehlerzustände zu überwachen.

Drucker

Für den Druck kommt im TIETZ inzwischen fast ausschließlich netzfähige Technik zum Einsatz. Das hat einerseits zu einer Reduzierung der Anzahl an Druckern geführt. Andererseits ist damit eine Nutzung von verschiedenen Stellen aus möglich und die Drucker können zentral verwaltet werden. Durch den verstärkten Einsatz von Kopierern an zentralen Stellen wurde die Druckeranzahl weiter reduziert. Direkt einem Arbeitsplatz zugeordnete Drucker gibt es keine mehr, auch Drucker in Einzelbüros können von überall genutzt werden.

Insgesamt sind etwa 80 Drucker im Einsatz, die meisten davon sind Netzdrucker. Hinzu kommen etwa 15 als Netzdrucker eingerichtete Kopierer. Im zentralen Netz sind mehr als 100 Druckerwarteschlangen eingerichtet, im Benutzernetz der Stadtbibliothek zehn.

[1] PC: Personalcomputer

Überwacht werden müssen Toner- und Trommelverbrauch sowie Fehlerzustände. Für die Drucktechnik gibt es ein Angebot der Firma Triumph-Adler, eine Software zur Druckerüberwachung einzusetzen.

Telefonie

Als Telefonanlage ist eine Hicom 4000 im Einsatz. Die Telefonanlage ist im Verbund der Stadt Chemnitz angeschlossen, sie wird durch die Stadtverwaltung technisch betreut. Vor zwei Jahren wurde im Kernbereich der Stadtverwaltung auf IP-Telefonie umgestellt, das TIETZ ist davon bisher ausgenommen.

3.1.3 softwaretechnische Infrastruktur

Serverdienste

Infrastrukturdienste sind für die Betriebsfähigkeit ganz wesentlich, deshalb muss Aufrechterhaltung und Ausfallsicherheit eine erhöhte Aufmerksamkeit geschenkt werden. Sie werden sowohl im Mitarbeiter- wie im Benutzerbereich benötigt und jeweils durch dedizierte Server erbracht. Sie müssen einzeln auf Verfügbarkeit, Funktionsfähigkeit und Fehler überwacht werden.

Netzwerkdienste wie DNS, DHCP, NTP[2], SMTP, HTTP, FTP[3] und andere werden zentral bereitgestellt und von den dedizierten Servern im Mitarbeiter- und Benutzerbereich einzeln bereitgestellt.

Webserverdienste werden in drei Ebenen bereitgestellt: es gibt eigene Serverkapazität für interne und externe Anwendungen, außerdem gemietete Kapazität bei einem Provider.

Authentisierungsdienste, Verzeichnisdienste werden im Mitarbeiterbereich über Windows Active Directory und OpenLDAP[4] bereitgestellt und umfassen unter anderem Anmeldung, Rechte- und Einstellungsverwaltung. Die Dienste beruhen derzeit auf Windows 2008 Server.

[2]NTP: Network Time Protocol
[3]FTP: File Transfer Protocol
[4]LDAP: Lightweight Directory Access Protocol

Dateiablage (File-Server) ist als zentrale Ablage über Netzlaufwerke realisiert. Der Speicherplatz wird über ein Netzwerkspeichergerät, ein sogenanntes NAS[5] bereit gestellt, zur Zeit sind drei Geräte im Einsatz. Der Speicherbedarf hat sich seit 2004 vervielfacht, diese Tendenz wird sich fortsetzen. Projekt- oder nutzerbezogene Anforderungen zu Speicherplatz gibt es derzeit noch nicht. Hier ist eine spezielle Überwachung der Kontingente notwendig.

Druckdienste (Print-Server) werden über zentrale Printserver für die jeweils angeschlossenen Drucker erbracht und über ein Druckerverzeichnis angesprochen.

Terminalserverdienste bilden die Basis für die ThinClient-Arbeitsplätze. Sie werden von 8 Terminalservern unter Windows 2003 Server und Citrix erbracht. Die ThinClients werden über eine zentrale Instanz verwaltet.

Verfügbarkeit, Auslastung und verschiedene Fehlerzustände müssen überwacht werden.

System und Managementdienste

Datensicherung erfolgt je nach Datenquelle auf verschiedene Weise.

Die Windowsserver werden über ein BackUp-System mit Bandbibliothek gesichert. Das BackUp wird mit ArcServe verwaltet und folgt einem 5-Tage-Konzept. Längerfristig werden Daten nur auf Anfrage archiviert.

Die Linux-Server werden über einen Sicherungsserver mithilfe von Skripten auf ein Netzlaufwerk gesichert.

Die SISIS-SunRise-Datenbank wird mithilfe von Skripten auf das im SISIS-Server eingebaute Bandlaufwerk gesichert.

Die verschiedenen Sicherungen müssen auf Fehler überwacht werden.

IT-Fachverfahren

SISIS-SunRise das integrierte Bibliothekssystem SISIS-SunRise ist ein 2-schichtiges Verfahren und läuft als Client-Server-System. Es verwendet das Datenbanksystem Sybase und bietet verschiedene Dienste an.

Die Dienste müssen jeweils auf Funktionsfähigkeit überwacht werden.

[5] NAS: Network Attached Storage

KuferSQL das System KuferSQL für den Volkshochschulbereich ist ein 2-schichtiges Verfahren und läuft als Client-Server-System. Als Datenbanksystem wird MS-SQL-Server 2008 eingesetzt.

Systemdienst und Datenbank müssen überwacht werden.

Bib-Control das Bibliotheks-Controllingsystem Bib-Control ist ein 2-schichtiges Verfahren mit einer integrierten Datenbank.

Es sollte überwacht werden.

HKR/SESSION die Haushaltssoftware HKR und das Ratsanfragesystem SESSION werden in der Stadtverwaltung gehostet und betreut. Der Zugang erfolgt über ein Access Gateway, das über Internet erreicht wird.

Hier kann nur die Erreichbarkeit überwacht werden.

Nutzeraufkommen

Mitarbeiter Im "Das TIETZ" sind über 100 ständige Mitarbeiter zu betreuen, dazu kommen weitere zeitweise Mitarbeiter wie Praktikanten oder geringfügig Beschäftigte. Insgesamt sind etwa 150 Accounts eingerichtet.

Benutzer Die Benutzer der Einrichtungen im TIETZ, also Besucher, Leser oder Schüler nutzen ebenfalls IT-Technik der Einrichtungen. Dabei fallen zwar auch in kleinerem Maß Betreuungsaufgaben an, der wesentliche Aufwand wird aber durch die Betreuung der IT-Technik wie oben beschrieben verursacht.

Das trifft im gleichen Maß auf die für Benutzer eingerichteten Netzsegmente wie das OPAC-Netz in der Stadtbibliothek oder das Schulungsnetz in der Volkshochschule zu. Die Netzsegmente werden durch die gleiche Technik bereitgestellt wie das Mitarbeiternetz.

3.2 Forderungen an ein Monitoringsystem

Die Analyse der IT-Infrastruktur im TIETZ zeigt wie erwartet ein komplexes und heterogenes Bild. Es müssen sehr verschiedene Dienste und Rechner einbezogen werden, die sehr unterschiedliche Anforderungen an die Überwachung stellen. Um die eingangs formulierten Ziele zu erreichen, sollte das einzuführende Monitoringsystem verschiedene Anforderungen erfüllen sowie bestimmte Eigenschaften aufweisen. Diese werden im folgenden mit

kurzen Erläuterungen entwickelt und dargestellt.

3.2.1 allgemeine Forderungen

geringer Aufwand Der Zeitaufwand für die Überwachung muss sich deutlich verringern, deshalb sollten so viele Aspekte wie möglich automatisiert ablaufen.

geringe Komplexität Einrichtung und Administration des Monitoringsystems müssen ebenfalls zur Verringerung des Aufwandes beitragen und dürfen diesen nicht wieder zunichte machen. Das System darf keine Spezialkenntnisse erfordern, sondern muss im normalen Alltagsgeschäft durch den Systemadministrator nicht nur bedient, sondern auch eingerichtet und angepasst werden können.

umfassendes Abbild Es muss die gesamte Infrastruktur abbildbar sein, möglichst auf einen Blick und mit der Möglichkeit zur Detaillierung. Durch das heterogene IT-System ist dazu Plattform- und Herstellerunabhängigkeit erforderlich.

auf Code basierend Eine auf Code basierende Konfiguration ermöglicht eine Integration mit anderen Tools und damit eine weitere Automatisierbarkeit verschiedener an das Monitoring anschließender oder vorausgehender Prozesse. Damit wird ein durchgängig automatisierter IT-Betrieb möglich, was wiederum zur Verringerung des Aufwandes beiträgt.

automatisierte Konfigurationserstellung Für das System sollten Werkzeuge für eine automatisierte Konfigurationserstellung vorhanden sein.

preiswert Das IT-System im TIETZ ist im Bereich “kleinere Umgebung” anzusiedeln, die möglichen Kosten sollten dazu im Verhältnis stehen. Da das Projekt ausschließlich im Rahmen dieser Arbeit bearbeitet wird, steht außerdem kein investives Budget zur Verfügung.

Open Source Die vorliegende Arbeit trägt durchaus einen anteiligen Forschungsaspekt. Deshalb soll ein Einblick in den Quelltext möglich sein und nicht durch restriktive Lizenzbedingungen eingeschränkt oder unmöglich werden. Ein Austausch der Ergebnisse wird ebenfalls vereinfacht. Zudem ist eine einfache Skalierbarkeit gegeben, falls mehrere Systeme eingesetzt werden sollen.

nachhaltig Für das Monitoringsystem muss eine aktive Weiterentwicklung gegeben sein, um zukünftige Anforderungen abbilden sowie Fehler und Sicherheitslücken einfach beheben zu können. Im Falle der Einstellung des System muss es durch Nachfolgesysteme einfach ablösbar sein.

3.2.2 funktionale Forderungen

ganzheitliche Sicht Das System muss alle Komponenten des IT-System berücksichtigen: Server, Hosts, Netzwerkgeräte, Datenbanken, Netzwerkdienste und Anwendungen. Außerdem sollen auch andere Aspekte des IT-Betriebes wie Logs, Ereignisse oder Umgebungsparameter erfassbar sein. Letztendlich soll ein Endnutzer-Monitoring ermöglicht werden, dass nicht nur feststellt, ob ein Server innerhalb vorgegebener Parameter funktioniert, sondern auch, ob die Anwender einwandfrei arbeiten können.

Virtualisierung Virtualisierung erhöht die Komplexität beim Betrieb erheblich. Das Monitoringsystem muss diese erhöhte Komplexität und die entstehenden gegenseitigen Abhängigkeiten abbilden und verarbeiten können.

24x7-Monitoring Die Überwachung muss kontinuierlich erfolgen. Das IT-System ermöglicht es den Einrichtungen mittlerweile, auch außerhalb der Öffnungszeiten bestimmte Leistungen zu erbringen. Außerdem ist im Fehlerfall eine rechtzeitige Alarmierung oder die Erkennung von sich ankündigenden Problemen über Trends möglich.

Schwellwerte Um nicht nur Ausfälle, sondern auch Leistungsminderungen rechtzeitig erkennen zu können, muss der Abgleich mit benutzerdefinierten Schwellwerten möglich sein. Ebenfalls muss sich der Grund für Leistungsminderungen feststellen lassen.

Benachrichtigungsoptionen Für die Fehler- und Zustandsbenachrichtigung sollen verschiedene Möglichkeiten gegeben sein.

Trendberichte Das System muss die Aufzeichnung von Statistiken ermöglichen. Für eine frühzeitige Erkennung von Problemen soll daraus die Erstellung von Berichten möglich sein. Andererseits sollen so auch nicht ausgelastete Ressourcen erkennbar werden.

3.3 Infrastrukturanalyse

Für das Einrichten des Monitoringssystems müssen alle zu überwachenden Objekte und ihre zu überwachenden Eigenschaften detailliert bekannt sein. Außerdem muss ihr Platz in der betrachteten Infrastruktur, insbesondere ihre Erreichbarkeit im Netzwerk bekannt sein. Da ein Monitoringsystem seine Kontrollen über das Netzwerk ausführt, können nicht über das Netzwerk erreichbare Infrastrukturobjekte auch nicht trivial überwacht werden. Als Beispiel sei hier der Serverraum und seine Merkmale Unversehrtheit und Raumtemperatur genannt. Es wird im Folgenden aber noch gezeigt, dass es Möglichkeiten gibt, auch solche Objekte und ihre Merkmale zu überwachen.

Eine mögliche Quelle für Objektlisten sind Inventarverzeichnisse. Diese können in verschiedener Form schriftlich oder als Datei vorliegen und müssen analysiert und in eine auswertbare Form gebracht werden. Da IT-Technik in der Regel über Investitionen beschafft wird, liegen bereits aus der Beschaffung entsprechende Listen vor, die untersucht werden können. Außerdem ist ein täglicher Betrieb ohne Übersicht, wo welche Technik im Einsatz ist, nur schlecht möglich. Diese Übersichten können ebenfalls untersucht und ausgewertet werden.

Inventarlisten können aber auch mithilfe von Werkzeugen erstellt werden, die das Netzwerk absuchen und dabei Hosts und Services erfassen. Das hat den Vorteil, dass für die gefundenen Hosts auch gleich die Erreichbarkeit gegeben ist. Hosts, die nicht erreichbar sind, werden dagegen nicht erfasst.

Eine weitere mögliche Quelle sind vorhandene Kontrollroutinen. Dabei können sowohl zu überwachende Technik als auch Dienste ausgemacht werden. Kontrollroutinen können tägliche Routinen wie Kontrollgänge oder Kontrolle durch Augenschein sein, aber auch Kontrollen, die mithilfe von Checklisten am Computer oder automatisiert durch Skripte vorgenommen werden. Diese Routinen gilt es aufzuspüren, zu erfassen und auszuwerten.

Bei der Auswertung kann es verschiedene Gründe geben, warum eine Überwachung von gefundenen Objekten und Merkmalen nicht möglich oder sinnvoll ist. Das können Fälle sein, bei denen eine Erreichbarkeit nicht oder nur mit hohem Aufwand hergestellt werden kann. Ein besonderer Fall in diesem Zusammenhang ist Technik, die nur zeitweise erreichbar ist, weil sie außerhalb ihrer Nutzungszeit ausgeschaltet wird. Da die Einschaltzeit nicht genau bestimmbar ist, würde das regelmäßig zu Fehlalarmierungen führen. Deshalb muss es eine Aufgabe der Erhebung sein, eine Auswahl aus den gefundenen Objekten zu treffen.

3.3.1 Dokumentenanalyse

Für die tägliche Arbeit bei der Systembetreuung, aber auch als Inventarübersicht oder als Basis für die Ersatzbeschaffung, wird neben der Inventarisierung durch die zentrale Verwal-

tung im städtischen Inventarsystem KVV[6] eine eigene Übersicht, die sogenannte *dv-tietz*, gepflegt. Die darin erfassten Merkmale und die erfasste Technik unterscheiden sich von denen in der KVV-Inventarisierung erfassten, weil mit den Daten unterschiedliche Ziele verfolgt werden. Während die Erfassung im KVV vor allem der Vermögensverwaltung dient und eher wirtschaftliche Aspekte berücksichtigt, enthält die *dv-tietz* vor allem technische Details, die für die tägliche Arbeit benötigt werden. Im KVV werden außerdem nur inventarisierungspflichtige Gegenstände erfasst, dagegen erfasst die *dv-tietz* jegliche relevante Technik. Deshalb werden hier diese Daten als Grundlage für die Inventarisierung verwendet. Die Daten liegen als dBase-Datenbank vor und können als Report ausgegeben werden.

Für die Vergabe der IP-Adressen wurde beim Einzug ins TIETZ ein IP-Konzept erstellt. Entsprechend der dort festgelegten Vorgaben werden die IP-Adressen für alle Geräte, die an das Netzwerk angeschlossen werden sollen, vergeben. Das IP-Konzept und die bereits vergebenen Adressen für zentrale IT-Technik werden im Wiki der IT-Systembetreuung dokumentiert, das so ebenfalls als Datenquelle genutzt werden kann.

Im Active Directory der Windows-Domäne im TIETZ werden für alle angeschlossenen Geräte IP-Adressen und Hostnamen gepflegt. Das IP-Konzept sieht vor, dass angeschlossene Geräte ihre IP-Adresse per DHCP bekommen, dazu wird bei der Inbetriebnahme per Hand jeweils ein entsprechender Eintrag gepflegt. Im Active Directory wird der vom Gerät gemeldete Hostname automatisch im DNS eingetragen, so dass das Gerät unter seinem Hostnamen angesprochen werden kann. Aufgrund der automatischen Vergabe sollten alle angeschlossenen Geräte erfasst werden. Das funktioniert nicht in jedem Fall, weil es Geräte gibt, die die entsprechend notwendigen Netzwerkprotokolle nicht oder nicht richtig beherrschen. Weiterhin kommt es zu Fehlern durch Inkompatibilitäten zwischen dem Active Directory unter Windows-Server und Geräten, die unter einer anderen Version oder gar nicht unter Windows laufen.

Bei der Auswertung der oben genannten Quellen fiel auf, dass sich die ermittelten Daten voneinander unterscheiden. Deshalb wurde eine Synopse in Form von Tabellen angefertigt, um die Unterschiede aufklären zu können. Die Unterschiede konnten auf unterschiedliche Datenstände bei der Erfassung, Fehler bei der Eingabe und die gerade erwähnten Probleme technischer Art zurück geführt werden. Durch die Verbindung der verschiedenen Quellen und deren Vergleich ließ sich aber eine belastbare Datenbasis herstellen.

3.3.2 Netzwerkscan

Für die Inventarisierung bietet sich der Einsatz von Netzwerk-Tools an, weil die Überwachung der Infrastruktur über das Netzwerk erfolgt (vgl. (Lim06, S. 163 ff.)). Inventarisie-

[6] KVV: Kommunale Vermögensverwaltung (Software zur kommunalen Vermögensverwaltung)

rung ist als Basis für verschiedene Aufgaben der IT-Systembetreuung, zum Beispiel für Lizenzverwaltung oder Softwareverteilung, eine immer wiederkehrende Aufgabe im Tagesgeschäft. Es gibt deshalb bereits eine Vielzahl von möglichen Tools, die diesen Prozess unterstützen. Für komplette Inventarisierungs-Management-Systeme sind Einarbeitung und Aufwand ähnlich hoch wie beim betrachteten Monitoringsystem. Im Rahmen dieser Arbeit können deshalb nur einfache, schnell zu erlernende Tools zum Einsatz kommen. Die Wahl fiel auf das Netzwerkanalysetool NMAP[7], weil es bereits bei früheren Aufgaben eingesetzt worden war und Erfahrungen damit vorliegen. Zum Vergleich wurde außerdem das Netzwerktool The Dude[8] herangezogen. Während NMAP auf einem Linuxrechner installiert ist, läuft The Dude unter Windows.

Zur Inventarisierung genügt bei NMAP ein einfacher Ping-Test (Lyo09), der über längere Zeiträume mehrfach durchgeführt wurde, um auch nicht ständig eingeschaltete Objekte erfassen zu können. Außerdem wurden verschiedene IP-Adressbereiche verwendet, um die Last auf dem Netzwerk klein zu halten. Folgender Aufruf soll als Beispiel dienen:

```
nmap -sn -PE -PA21,23,80,3389 172.16.0.0/24
```

Dabei entstehen Listen, die ähnlich wie folgende aufgebaut sind:

Listing 3.1: NMAP Ergebnis

```
Starting Nmap 5.61TEST2 ( http://nmap.org )
Nmap scan report for zentral.stadtbibliothek-chemnitz.de (172.16.0.1)
Host is up (0.0051s latency).
MAC Address: 00:0F:20:E4:87:00 (Hewlett-Packard Company)
Nmap scan report for 172.16.0.2
Host is up (0.0049s latency).
MAC Address: 00:10:7B:12:7C:9E (Cisco Systems)
Nmap scan report for 172.16.0.10
Host is up (0.0020s latency).
MAC Address: 00:21:A0:A8:EC:91 (Cisco Systems)
Nmap scan report for sbc-21.stadtbibliothek-chemnitz.de (172.16.0.21)
Host is up (0.00023s latency).
MAC Address: 00:19:99:01:8A:FB (Fujitsu Technology Solutions)
 ...
```

Man erkennt, das aus der Liste die IP-Adresse, der Hostname, sofern ermittelbar, die MAC-Adresse und ein möglicher Hersteller ablesbar sind. Aus der Herstellerbezeichnung lassen sich Rückschlüsse auf die Art des Gerätes ziehen. Die Scanergebnisse können außerdem in

[7]URL: http://nmap.org/
[8]URL: http://www.mikrotik.com/dude/

einem NMAP-eigenen XML-Format abgespeichert werden, aus dem sich wiederum Tabellen ähnlich denen aus der Dokumentenanalyse generieren lassen.

Einige der Scans wurden mit The Dude wiederholt, da sich die Ergebnisse nicht wesentlich unterschieden, wurde im folgenden nur noch NMAP verwendet.

Aufgrund der verschiedenen Erfassungszeiten ergaben sich lediglich minimale Unterschiede, deren Ursachen leicht aufgeklärt werden konnten. Die Ergebnisse der Netzwerkscans wurden nun ebenfalls in die oben beschriebene Synopse eingefügt und miteinander verglichen. Die dabei auftretenden Unterschiede konnten in der Regel darauf zurückgeführt werden, dass Geräte nicht ans Netzwerk angeschlossen waren oder sich in einem Netzwerksegment befanden, das für die Scans nicht erreichbar war. Daraus lassen sich bereits erste Schlußfolgerungen für die grundsätzliche Erreichbarkeit der einzelnen IT-Geräte ziehen. Entweder muss diese über verschiedene Wege hergestellt werden oder die Geräte und die darauf laufenden Dienste lassen sich nicht mit dem Monitoringssystem überwachen.

Durch den Vergleich der Ergebnisse aus der Analyse der Inventarverzeichnisse und der Netzwerkscans und die nachfolgende Aufklärung der Unterschiede ließ sich wie gewünscht die Datenbasis für die Einrichtung des Monitoringsystems weiter vervollständigen.

3.3.3 Ablauferhebung

Für die Überwachung der Infrastruktur ist es nicht nur wichtig zu wissen, welche Geräte überwacht werden sollen, sondern auch welche Dienste von diesen Geräten erbracht werden. Nur diese sind es schließlich, die letztlich für den Nutzer interessant sind. Diese Daten lassen sich nicht so formal einfach erheben wie das bei den vorher beschriebenen Methoden der Fall war. Deshalb sollte außerdem eine Ablauferhebung durchgeführt werden, um möglichst viele Dienste als Überwachungsobjekte identifizieren zu können.

In die Erhebung wurden vor allem solche Abläufe einbezogen, die beschreiben, wie bisher Überwachung betrieben wurde. Die betrachteten Abläufe haben dabei formalen Charakter und sind meist aus bisher gemachten Erfahrungen und aufgetretenen Problemen hervorgegangen.

“Turnschuh-Monitoring” Aus der täglichen Arbeit heraus sind Kontrollroutinen entstanden, die durchaus formalen Charakter tragen, da sie in der Regel täglich durchgeführt werden und immer ähnlich ablaufen. Zusammengefasst kann man diese Kontrollroutinen als “Turnschuh-Monitoring” beschreiben, da sie den Systemadministrator als Kontrollinstanz erfordern und auf eine Kontrolle durch Augenschein vor Ort hinauslaufen. Die wichtigste ist der Kontrollgang durch den Serverraum.

Dazu wird der täglich notwendige Wechsel des Sicherungsbands für das Bibliothekssystem genutzt, um die Rackschränke und alle enthaltenen Geräte auf Auffälligkeiten zu kontrollieren. Ausfälle zeigen sich dabei ebenso wie Defekte in der Regel durch Veränderungen der Bereitschafts- und Warnanzeigen, diese können mit einem Blick erfasst werden. Beispielsweise wurden die meisten Festplattenausfälle auf diese Weise zuerst festgestellt. Außerdem ist eine einfache Kontrolle der Internetanbindung sowie der Funktion von kabelgebundenem und drahtlosem Netzwerk möglich. Kontrolliert werden kann ebenfalls die Funktion des Klimaschranks zur Abwärmeabführung, die Temperatur im Serverraum und der Zugang zum Serverraum.

Diese Kontrolle ist dabei durchaus effektiv: innerhalb sehr kurzer Zeit können eine Vielzahl von Zuständen auf Normalfunktion oder Abweichungen erfasst werden. Ein Nachteil ist, dass auftretende Fehler erst wieder nach 24 h, über das Wochenende sogar erst nach drei Tagen bemerkt werden. Außerdem bleibt die Kontrolle an der Oberfläche und erfasst nur Fehler und Warnungen, die durch Anzeigen an den Geräten signalisiert werden. Eine Erkennung von Trends, die auf sich anbahnende Probleme hinweisen, kann sie dagegen nicht leisten.

Trotzdem lassen sich aus diesem Ablauf eine ganze Reihe von Überwachungsobjekten ableiten, die Eingang in die Konfiguration des Monitoriungssystems finden sollen. Dazu wurde der Ablauf dokumentiert und auf einzelne Geräte, Dienste und deren Eigenschaften aufgeschlüsselt.

Checklisten Eine weitere wesentliche Routine zur Überwachung des IT-Betriebs ist die Kontrolle am Bildschirm mithilfe von Logs, entweder direkt oder in aufbereiteter Form, anhand von Systemnachrichten als Mails oder mit einfachen Shellkommandos zur Ausgabe von Serverzuständen. Diese Routine ist als Checkliste erfasst und wird täglich abgearbeitet.

Logs Unter einem Log wird hier die automatische aufgezeichnete Protokollierung von Ereignissen auf einem Computersystem verstanden. Die Ereignisse können dabei sowohl Hardware als auch Betriebssystem oder Anwendungssoftware betreffen. Die Logs werden entweder direkt auf dem jeweiligen Gerät oder aufbereitet über Tools betrachtet. Das ist vor allem dadurch bedingt, dass die Protokolle verschiedene Herkunft haben und meist eigene Formate aufweisen. Das Ereignislogformat unter Windows unterscheidet sich beispielsweise deutlich vom Syslogformat unter Linux oder von einem Router. Jede Anwendungssoftware bringt weitere Formate mit. Als Werkzeuge zur Aggregation werden Tools wie AWStats[9], LogAnalyzer[10] oder MyEventViewer[11] eingesetzt. Die Durchsicht von Logs

[9] URL: `http://www.awstats.org/`
[10] URL: `http://loganalyzer.adiscon.com/`
[11] URL: `http://www.nirsoft.net/utils/my_event_viewer.html`

erfordert ein aktives Vorgehen, es erfolgt keine Signalisierung bestimmter Ereignisse. Innerhalb der Kontrollroutine werden nur ausgewählte Logs regelmäßig durchgesehen, andere Logs werden nur bei Bedarf verwendet.

Mails Bestimmte Systemereignisse werden nicht nur in Logdateien aufgezeichnet, sondern auch als Mail verschickt. Dafür gibt es eine entsprechend eingerichtete Mailadresse, über die die Mails abgerufen werden. Diese Form der Benachrichtigung wird entweder für besondere Ereignisse eingesetzt oder wenn nur diese Möglichkeit besteht. Ein Beispiel für den ersten Fall sind die Benachrichtigung des Backupsystems über den Verlauf der täglichen Datensicherung und die statistische Auswertung vom Mailsystem. Ebenso werden die Ausgaben von zeitgesteuerten Skripten per Mail ausgegeben. Nachrichten der Netzwerkdrucker zu Ereignissen wie `Toner alle` oder `Trommel ersetzen` sind ein Beispiel für den zweiten Fall.

Die Benachrichtigung per Mail erfordert kein aktives Vorgehen. Allerdings muss auch das Ausbleiben von Mails beachtet werden, da es oftmals ein Hinweis auf einen schwerwiegenden Fehler ist. So verschickt das Backupsystem erst nach Beenden der Datensicherung eine Mail, bei Verzögerung oder Abbruch fehlt diese.

Shellbefehle Ein recht guter und schneller Überblick über den Serverzustand lässt sich durch Einsatz einfacher Shellbefehle bekommen. Unter Linux ist das per Entwurf Stand der Technik, unter Windows gewinnt es mit dem Einzug der Power-Shell zunehmend an Bedeutung. So bekommt man mit dem Befehl `top`[12] einen Überblick über die laufenden Prozesse und die Speicherverwendung, mit `mailq`[13] werden ausstehende Mails angezeigt, `df`[14] beschreibt die Auslastung der Festplatten und `uptime`[15] liefert eine Aussage über die Serverlast. Davon wird routinemäßig Gebrauch gemacht, es erfordert ebenfalls ein aktives Vorgehen.

Aus der verwendeten Checkliste lassen sich die Überwachungsobjekte direkt ableiten, sie stellt somit ein wichtiges Dokument für die Erhebung dar.

Skripte Die eben beschriebenen Shellbefehle eignen sich hervorragend für die Automatisierung von Überwachungsaufgaben mithilfe von Skripten. Im Laufe der Zeit sind so für wichtige oder häufig wiederkehrende Aufgaben Skripte entstanden, die zeitgesteuert aufgerufen werden und je nach Problemlage Meldungen verschicken oder Aktivitäten auslösen. Damit werden zum Beispiel Server oder Dienste geprüft, ob sie noch laufen oder es wird

[12]URL: `http://unixhelp.ed.ac.uk/CGI/man-cgi?top+1`
[13]URL: `http://www.postfix.org/sendmail.1.html`
[14]URL: `http://unixhelp.ed.ac.uk/CGI/man-cgi?df`
[15]URL: `http://man7.org/linux/man-pages/man1/uptime.1.html`

getestet, ob notwendige Netzverbindungen in die Außenstellen noch bestehen. Ein Skript prüft regelmäßig zu Beginn der Öffnungszeit die Benutzer-PCs der Stadtbibliothek, ob das Einschalten über Wake-on-LAN erfolgreich war und ob in der jeweiligen Logdatei Fehler protokolliert sind. Das Ergebnis wird dann per Mail verschickt und kontrolliert.

Die verwendeten Skripte wurden erfasst und die darin enthaltenen Objekte identifiziert. Bei kritischer Betrachtung fällt auf, dass die entstandenen Skripte nicht systematisch entstanden sind und so nur Teile der gesamten Infrastruktur abbilden. Notwendig ist deshalb eine Ergänzung im Rahmen einer ganzheitlichen Betrachtung.

3.3.4 Dokumentation

Für die Dokumentation der Analyseergebnisse wurden die bisher bei der Erhebung entstandenen Dokumente gesammelt und zusammengestellt. Dieser Schritt vereinfacht sich dadurch, dass alle bisherigen Ergebnisse als Computerdateien vorliegen oder darin überführt werden konnten. Die weitere Auswertung, Prüfung und Verwaltung erfolgt nun werkzeuggestützt mit einem Wiki. Verwendet wird PmWiki[16], die Auswahl der Wikisoftware erfolgte bereits bei der Vorbereitung im Praktikum (Sch10).

Die Anwendung eines Wikis für die Anforderungsverwaltung bietet zwar nicht den Funktionsumfang, den spezielle Softwaresysteme für diesen Zweck haben. Es hat aber den Vorteil, dass es sehr einfach zu bedienen ist und sich leicht an den Anwendungszweck anpassen lässt. Folgende Qualitätseigenschaften werden an Anforderungen gestellt (Kra14):

- Korrektheit
- Machbarkeit
- Notwendigkeit
- Priorisierung
- Eindeutigkeit
- Testbarkeit

Diese Eigenschaften lassen sich recht gut durch ein Wiki unterstützen. Dazu werden verschiedene Funktionen der Wikisoftware genutzt. Durch die Versionierung der Wikiseiten ist eine Rückverfolgbarkeit gegeben. Über Wikilinks lassen sich einfach Verweise pflegen, zum Beispiel von Anforderung auf Umsetzung und Test und umgekehrt. Generierbare Listen, sogenannte Pfade, gewährleisten eine einfache Form der Testbarkeit und Eindeutigkeit. Änderungen werden über eine eingefügte Kategorie *Änderung* verwaltet, die in einer eigenen Wikiseite dargestellt wird. Über Basisfunktionen wie `Rückverweise`, `fehlende` oder

[16]URL: `http://www.pmwiki.org/`

`verwaiste Seiten` oder `letzte Änderungen` lässt sich die Anforderungsverwaltung weiter vereinfachen. Durch konsequente Nutzung dieser Funktionen lässt sich auf formale Art und Weise ein Modell aufstellen, an dem fehlende Informationen erkannt und ergänzt werden können.

Zentrales Element ist ein Formular, das der Beschreibung der Überwachungsobjekte fungiert. Es enthält alle notwendigen Angaben und Elemente, die zur Einbindung in den Kontext der Anforderungsverwaltung benötigt werden. Das Formular kann einfach kopiert und in eine neue Wikiseite eingefügt werden.

Listing 3.2: Wiki Formular

```
(:title Vorlage:)
(:keywords Monitoring, Icinga:)
(:description text:)

%rframe red%Seite wird gerade bearbeitet - [[~holm]] 01.03.2014%%

[[Monitoring/HomePage | Home]] | %trail% <<|[[Monitoring.NetzKnoten |
   +]]|>>
----

!! Hostname
||
||Name: || ||
||Alias: || ||

!! Attribute
||
||IP-Adresse: ||... ||
||MAC-Adresse: ||::::: ||
||Standort: ||. ||

!! Dienste
*

!! Klasse
* [[!Kategorie]]

----
[[ Monitoring.{$Name}?action=attr | Eigenschaften ]]
```

Alle anderen Seiten sind Freitextseiten, die je nach Erfordernis angelegt und formatiert werden. Dabei wird auf bestimmte, immer wiederkehrende Elemente geachtet, wie Link zur `Startseite`, `Bearbeitungsstatus` oder bei längeren Seiten `Inhaltsverzeichnis`. Alle Elemente unterliegen dabei einer ständigen, iterativen Anpassung.

3.3.5 Klassifizierung

Die bisher beschriebene Erhebung hat eine Vielzahl von möglichen Überwachungsobjekten ergeben. Für die weitere Bearbeitung ist es sinnvoll, diese Objekte zu strukturieren. Dadurch können ähnliche Objekte für eine einfachere Handhabung und Übersicht zusammengefasst und Prioritäten vergeben werden. Außerdem können Hierarchien und Abhängigkeiten erkannt und berücksichtigt werden. Entsprechende Kriterien lassen sich aus der Erhebung ableiten.

Eine offensichtliche Einteilung ist die nach der Art der Netzknoten. Geräte gleichen Typs wie beispielsweise Switche werden als eine Kategorie zusammengefasst. Weitere Kategorien können aus dem verwendeten Betriebssystem gebildet werden, bei der Ablaufanalyse hat sich gezeigt, dass in dieser Art gleiche Hosts auch gleiche Behandlung bei der Handhabung bieten. Das ist für die Einrichtung des Monitoringssystems interessant. Die unten stehende Grafik zeigt die Aufteilung der eingerichteten virtuellen Maschinen auf den vorhandenen Virtualisierungshosts, aus der sich beispielhaft entsprechende Kategorien ableiten lassen.

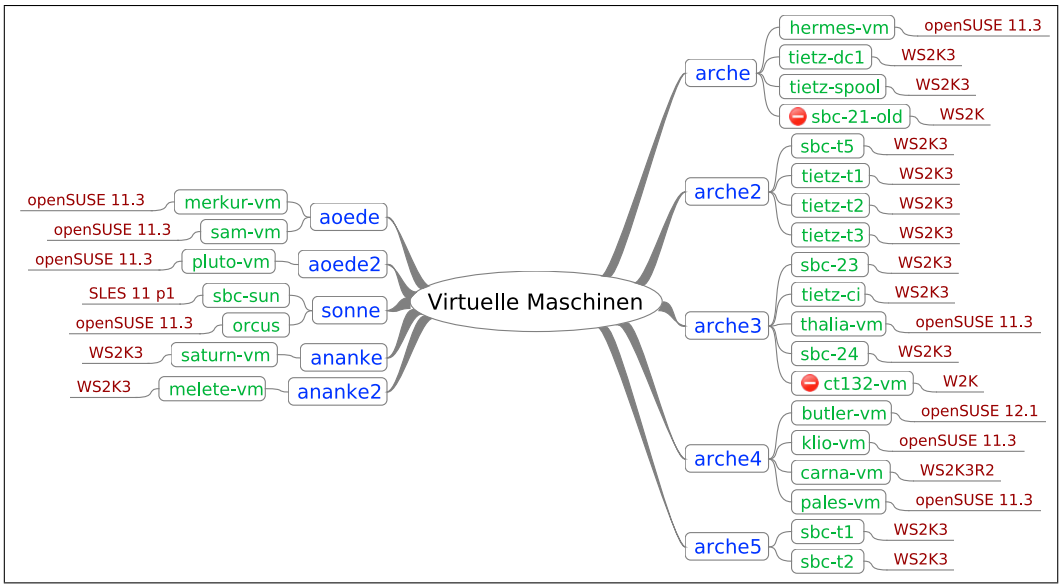

Bild 3.2: Virtuelle Maschinen

Grundsätzlich handelt es sich bei der betrachteten Infrastruktur um eine Umgebung kleinerer Größe. Die zentrale Server- und Netzwerktechnik ist räumlich und zahlenmäßig allerdings so umfangreich, dass die im Serverraum zusammengefasste Technik durchaus als klei-

nes Rechenzentrum aufgefasst werden kann und entsprechende Verfahren und Best Practice in kleinem Maßstab angewandt werden können (vgl. (Zie05)). Fehler und Ausfälle in diesem Bereich haben Auswirkungen auf die Betriebsfähigkeit der gesamten Einrichtung.

Der Schwerpunkt der Überwachung soll deshalb in diesem Bereich liegen.

Dagegen ist die Arbeitsplatztechnik schwieriger zu handhaben, da sie in der Regel nicht ständig eingeschaltet ist und schlecht zwischen ausgefallenem und ausgeschaltetem Zustand unterschieden werden kann. Der Ausfall eines PCs betrifft nur den jeweiligen Arbeitsplatz, von dem es in der Regel eine sofortige Rückkopplung im Problemfall gibt, ganz ohne Monitoringsystem.

Eine Ausnahme bildet die Technik im Benutzerbereich der Stadtbibliothek, von dort gibt es oft keine sofortige Rückmeldung und Fehler werden nicht so schnell erkannt. Die bereits eigerichteten Überwachungsmechanismen zeigen aber einen Bedarf an. Aufgrund der komplexeren Betrachtung wird die Übernahme in das Monitoringsystem mit nachrangiger Priorität eingeordnet.

Drucktechnik ist dagegen keine arbeitsplatzbezogene Technik mehr und steht meist nicht am Arbeitsplatz, sondern getrennt. Da die Drucktechnik einen Betreuungsaufwand über bloße Fehlerbeseitigung hinaus verursacht, soll sie ebenfalls in das Monitoring einbezogen werden. Sich abzeichnende Probleme wie Tonermangel oder Papierstau können so rechtzeitig bemerkt und vorausschauend behoben werden.

Ausgehend von diesen Überlegungen wurden unterschiedliche Klassen gebildet und die gefundenen Objekte entsprechend zugeordnet. Die gebildeten Klassen und ihre Beziehungen untereinander sind im Diagramm 3.3 dargestellt. Objekte können dabei verschiedenen Klassen gleichzeitig zugeteilt sein. So kann ein Server sowohl in der Klasse *Windowsserver* als auch in der Klasse *Virtuelle Maschine* enthalten sein.

Die Zuordnung der Objekte zu den einzelnen Klassen wird im Wiki dokumentiert. Dazu wird auf der Objektseite eine entsprechende Kategorie eingetragen. Auf der Kategorieseite werden dann alle Objekte dieser Kategorie automatisch angezeigt.

Mit der Auswahl und Klassifizierung der Überwachungsobjekte und ihrer Dokumentation im Wiki sind die Anforderungen für die Einrichtung des Monitoringsystems hinreichend genau bekannt. Diese müssen nun in das Monitoringsystem übertragen werden.

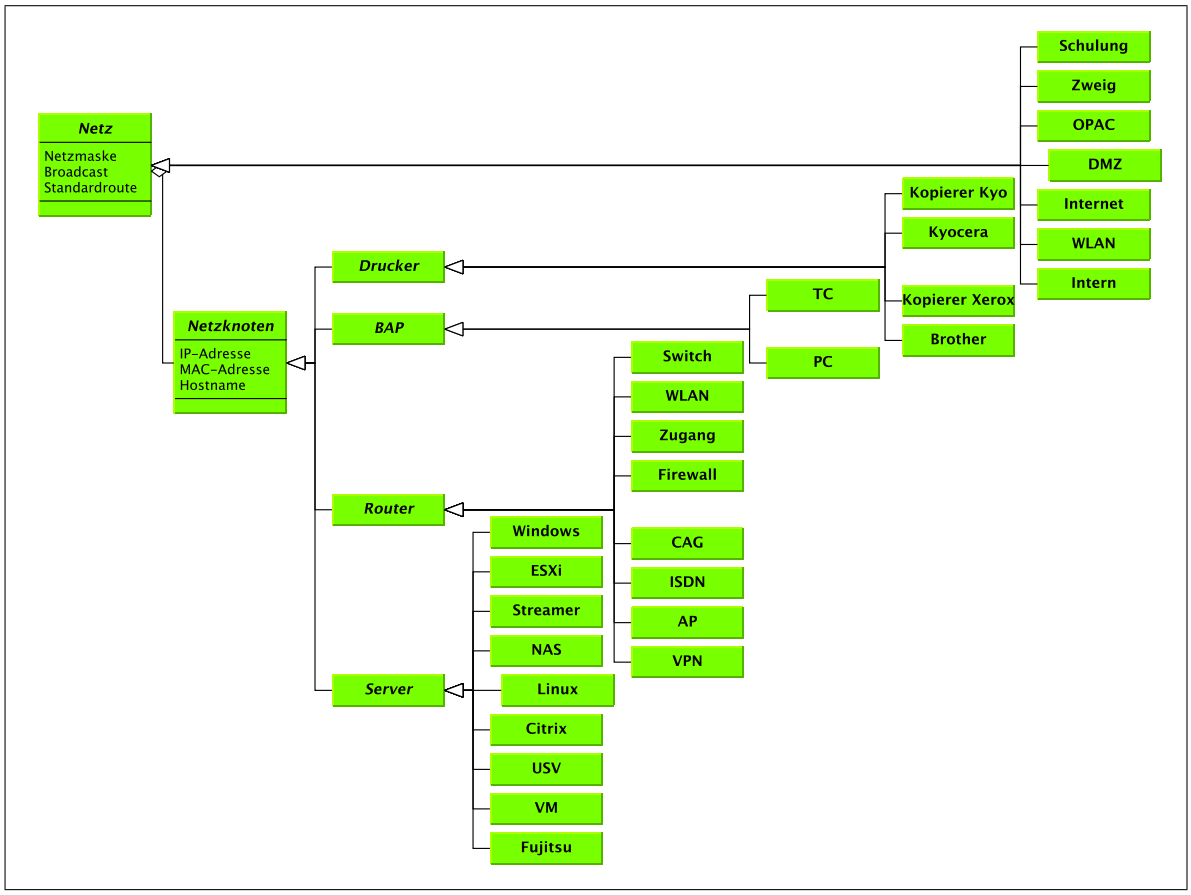

Bild 3.3: Klassen

4 Realisierung

In diesem Kapitel wird die praktische Umsetzung der Anforderungen beschrieben. Dazu werden die Auswahl des Monitoringsystems etwas genauer betrachtet, dann dessen Möglichkeiten beschrieben und schließlich eine mögliche konkrete Umsetzung konzipiert.

4.1 Auswahl Monitoringsystem

Zur Auswahl des zu verwendenden Monitoringsystems wurden bereits im Hauptpraktikum einige Überlegungen getroffen (Sch10). Diese sollen hier, auch aufgrund des vergangenen Zeitraums und der damit verbundenen Entwicklung, die die Systeme genommen haben, aufgegriffen und vertieft werden. In der Zwischenzeit haben sich nicht nur die Systeme selbst weiterentwickelt, sondern auch deren Anzahl ist gewachsen. Wurden von Jurzik noch 13 verschiedene Überwachungstools für kleine und mittlere Umgebungen miteinander verglichen (Jur07), verzeichnet die englische Wikipedia aktuell 60 verschiedene Systeme für das Netzwerkmonitoring[1], die deutsche Wikipedia 22 Systeme[2] und die Webseite der HW group, einem Anbieter IP-basierter Sensoren, insgesamt 25 Tools von Drittanbietern[3] [4].

Im "Das TIETZ" werden bereits verschiedene Überwachungslösungen verwendet, deshalb bietet es sich an, zunächst diese Lösungen zu betrachten und anhand einer (unvollständigen) Übersicht weitere Schlüsse zu ziehen. Bei der Auswahl wird dabei keinesfalls ein streng validierendes Verfahren angewandt, weil das mit einem sehr hohen Aufwand verbunden ist und den Rahmen der Arbeit deutlich übersteigen würde (Mig12).

Stattdessen wird anhand von Recherchen im Internet und Informationen verschiedener Anbieter versucht, anhand einfacher ingenieurtechnischer Überlegungen eine Wahl zu treffen. Dabei wird durchaus in Kauf genommen, nicht die optimalste Software ausgewählt zu haben. Für die vorliegende Arbeit reicht es aus, wenn alle notwendigen Anforderungen erfüllt werden können. Der Schwerpunkt liegt in der Untersuchung der IT-Umgebung und der Beurteilung an einem exemplarischen Monitoringsystem.

[1] URL: `http://en.wikipedia.org/wiki/Comparison_of_network_monitoring_systems`
[2] URL: `http://de.wikipedia.org/wiki/Netzwerk-Monitoring`
[3] URL: `http://de.wikipedia.org/wiki/Netzwerk-Monitoring`
[4] jeweils abgerufen am 27.4.2014

4.1.1 vorhandene Systeme

Im TIETZ werden aus verschiedenen Gründen bereits mehrere Tools zur Überwachung eingesetzt. In der Regel ist das Software, die die Hersteller den Geräten zur Verwaltung beifügen. Die Software ist in der Regel gut geeignet, um eine spezielle Geräteklasse zu beobachten und zu managen. Interessant ist hier vor allem, welche Parameter die Hersteller für ihre Geräte ausgewählt haben und wie sie diese darstellen.

HP ProCurveManager Für die Verwaltung der eingesetzten Switch-Infrastruktur wird der HP ProCurveManager der Firma Hewlett Packard eingesetzt. Im Einsatz ist die Free-Version, diese kann bis zu 25 verschiedene Geräte verwalten. Da mehr Geräte im Einsatz sind, muss hier bereits mit einem Trick gearbeitet werden. Gestackte Geräte, das sind mehrere, zusammengeschaltete Geräte an einem Standort, können über den ersten Switch, den sogenannten Commander, gemeinsam verwaltet werden. Da es weniger als 25 Standorte gibt, lassen sich alle Geräte verwalten.

Der HP ProCurveManager bietet folgende Funktionen:

- Übersicht Netzwerkstatus:
  Netzwerkstatus, der Informationen über Netzwerkgeräte, Endknoten, Ereignisse und Verkehrsaufkommen auf einem Bildschirm zeigt, von hier aus können weitere Details abgerufen werden
- Warnungen und Fehlersuche:
  eine Ereignisübersicht zeigt Warnmeldungen und stuft sie nach Schweregrad ein, so dass Engpässe und Probleme im Netz verfolgt werden können, es werden Informationen bis hin zu den spezifischen Ports angezeigt
- automatische Geräteerkennung:
  erkennt alle durch HP ProCurve verwaltbaren Netzwerkgeräte, es lassen sich IP-Subnetze und VLANs definieren, um die Erkennung zu beschleunigen
- Netzwerk-Topologie und Zuordnung:
  erstellt automatisch eine Karte der gefundenen Netzwerkgeräte, die Karten sind farblich gekennzeichnet, um den Gerätestatus anzuzeigen, hat mehrere Ansichten (physische, Subnet- oder VLAN-Ansicht)
- Geräteverwaltung:
  viele geräteorientierten Aufgaben können direkt von der Software durchgeführt werden, außerdem lässt sich zur Verwaltung aus der Software direkt auf die Web- und Befehlszeilenschnittstellen zugreifen

Die Software ist mittlerweile nicht mehr erhältlich. Stattdessen wird das sogenannte Intel-

ligent Management Center IMC vermarktet. Für die Software muss eine kostenpflichtige Lizenz erworben werden.

RingMaster Zur Verwaltung des WLAN-Netzes wird die RingMaster-Software der Firma Juniper Networks, ehemals Trapeze Networks, eingesetzt. Sie dient der Konfiguration, Überwachung und Optimierung der WLAN-Access-Points und ihres Zusammenspiels über ein Steuergerät, dem sogenannten WLAN-Controller.

Der RingMaster bietet folgende Funktionen:

- Konfiguration:
  Konfigurationsassistenten für die automatische Konfiguration der WLAN-Controller und Access-Points
- Monitoring:
  grafisches Dashboard zur Überwachung in Echtzeit von WLAN-Statusdaten, Datenverkehrsmustern, Client-Konnektivität, Access-Point- und WLAN-Controller-Status sowie Warnungen
- Alarme:
  Client-Kontrollliste für detaillierte Fehlerbehebung von korrelierten Client-Sitzungsdaten über einen längeren Zeitraum hinweg, sicherheitsrelevante Warnungen wie unbefugte Access-Points, DoS- und Probe-Angriffe sowie Ad-hoc-Netzwerke
- Berichte:
  anpassbare sowie Standardberichte, zum Beispiel Inventarübersicht, Clientsitzungen, Rogueübersicht, Switchkonfiguration und Installation von Geräten

Da die Firma Trapeze Networks übernommen wurde, wird die Software inzwischen von der Firma Juniper Networks angeboten. Sie wird als eine mit höchster Zuverlässigkeit, Performance und Sicherheit vermarktet. Die Software muss kostenpflichtig lizensiert werden, die Lizenz ist von der Anzahl Access-Points abhängig, die verwaltet werden sollen.

Citrix Management Console Die Citrix Management Console dient der Konfiguration und Überwachung von Citrix XenApp. Sie ist integraler Bestandteil der Citrix-Software und beinhaltet verschiedene Funktionen, unter anderem:

- Published Applications:
  Definition von veröffentlichten Anwendungen, das sind Anwendungen, die sich wie lokal installierte verhalten, jedoch auf dem Server laufen
- Installation Manager:
  Installation von Anwendungen auf einer großen Anzahl von Servern, die Anwendung

muss nur einmal installiert werden

- Resource Manager:
  überwacht verschiedene, definierbare Hard- und Software-Parameter und kann beim Überschreiten von einstellbaren Schwellwerten Benachrichtigungen auslösen
- Network Manager:
  stellt Schnittstelle über SNMP zur Einbindung in andere Monitoringsysteme bereit

Die Software kann sowohl eigenständig zur Überwachung eingesetzt werden als auch als Quelle für andere Systeme dienen. Besonders interessant sind hier die definierten Parameter, die als Vorlage dienen können. Für Citrix muss eine kostenpflichtige Lizenz erworben werden, die Console ist ein Bestandteil von Citrix.

Fujitsu ServerView Die Fujitsu ServerView Suite ist eine Software, die der Verwaltung von Industriestandard-Servern, speziell der Firma Fujitsu, dient.

Mit der Software werden folgende Ziele verfolgt:

- sichere und automatisierte Installation von Servern
- durchgehendes Status-Monitoring zur Gewährleistung eines möglichst unterbrechungsfreien und energieeffizienten Serverbetriebs
- Funktionen für die Vereinfachung und Automatisierung eines flexiblen IT-Betriebs
- Funktionen zur Vermeidung von Ausfallzeiten und damit Sicherung der Servicequalität
- Integration der PRIMERGY-Server in unternehmensweite Managementlösungen

Interessant ist vor allem der Teilbereich Servermanagement, der die Überwachung der Hardwarekomponenten wie Prozessoren, Arbeitsspeicher, Festplatten sowie ihrer Leistung, die Überwachung und Steuerung des Energieverbrauchs, die Analyse von Performance- und Auslastungsdaten sowie die Aktualisierung der Serverkonfiguration beinhaltet. Außer einer eigenen Darstellung kann die Software die gesammelten Daten über eine Schnittstelle an andere Monitoringlösungen weitergeben. Die Software ist Bestandteil der Serverausstattung und mit dem Kauf lizensiert.

4.1.2 kommerzielle Systeme

Im kommerziellen Bereich gibt es nur wenige Lösungen, die für einen Einsatz geeignet erscheinen. Die meisten wenden sich an Unternehmen und Einrichtungen mit einer großen System- und Netzwerk-Infrastruktur. Am ehesten kommen die Produkte *PRTG Network Mo-*

nitor[5] der Firma Paessler und *WhatsUp Gold*[6] der Firma Ipswitch für eine Betrachtung in Frage.

PRTG Network Monitor Der PRTG Network Monitor kann sowohl Windows- als auch Linux-Server auf Verfügbarkeit und Netzwerkverkehr überwachen. Die Überwachung erfolgt über WMI oder SNMP sowie verschiedene Routerprotokolle. Für jedes überwachte Gerät können verschiedene Eigenschaften wie Prozessorlast, Netzwerkverkehr oder Speicherbelegung betrachtet werden. Die Daten werden in einer Weboberfläche dargestellt, bei Warnungen und Fehlern werden Benachrichtigungen generiert. Die Software muss kostenpflichtig lizensiert werden und läuft unter Windows.

WhatsUp Gold WhatsUp Gold bietet Möglichkeiten für eine Netzwerk-, Host- und Serviceüberwachung. Die zu überwachenden Geräte können durch einen Netzwerkscan ermittelt werden oder werden manuell eingepflegt. Die Überwachung erfolgt über verschiedene Netzwerkprotokolle. Die Software kann sowohl den Status von Diensten als auch Daten einzelner Geräte wie Speicher- und Prozessorauslastung ermitteln und auswerten. Die Ergebnisse werden über eine betriebssystemspezifische oder eine Webschnittstelle dargestellt. Falls definierte Schwellwerte überschritten werden, erfolgt eine Benachrichtigung. Die Software muss kostenpflichtig lizensiert werden und läuft unter Windows.

4.1.3 Open-Source-Systeme

OpenNMS OpenNMS[7] ist ein Netzwerkmanagementsystem, das für die Anwendung in großen Unternehmen entwickelt wurde. Es hat den Anspruch, bis zu einer beliebigen Anzahl von Netzwerkknoten zu skalieren und dabei alle Aspekte des professionellen Netzwerkmanagements abzudecken.

Die Anwendung umfasst drei wesentliche Funktionen:

- Servicepolling: die Netzwerkknoten und die darauf laufenden Dienste werden über zyklische Abfragen bestimmt und überwacht
- Datenerfassung: alle Netzwerkinformationen werden erfasst, gespeichert und über Berichte abgerufen, für die Schwellwerte erstellt werden können
- Ereignismanagement: es können sowohl interne als auch externe Ereignisse empfangen und verwaltet werden, die die Benachrichtigungen und Eskalationen speisen

[5] URL: `http://www.de.paessler.com/prtg`
[6] URL: `http://www.whatsupgold.com/de/`
[7] URL: `http://www.opennms.org/`

OpenNMS verwendet für den Abruf der Informationen bevorzugt SNMP, hier liegt eine große Stärke des Systems. Es können aber auch andere Techniken eingesetzt werden bis hin zur Verwendung von Nagios-Plugins. Die Software ist hauptsächlich in Java programmiert und erfordert einige spezifische Voraussetzungen auf der Zielplattform. Es kann unter verschiedenen Betriebssystemen eingesetzt werden, der Haupteinsatz erfolgt unter Linux.

Zabbix Zabbix[8] ist ein Netzwerkmonitoringsystem und zur Überwachung großer IT-Infrastrukturen konzipiert. Für die Überwachung wird ein spezieller Zabbix-Agent eingesetzt, den es für verschiedene Betriebssysteme gibt. Darüber hinaus werden verschiedene Protokolle wie SNMP, SSH[9], WMI oder IPMI unterstützt, um ohne Agent Informationen abzurufen. Die Verwaltung erfolgt mittels Browser über eine Webschnittstelle. Ein besonderes Merkmal ist die komfortable Darstellung der Netzwerkknoten auf Karten unterschiedlichster Art. Ein Alarmsystem gibt die Möglichkeit, Benachrichtigungen bei wichtigen Ereignissen über verschiedene Wege zu verschicken. Die Software kann unter verschiedenen Unix-ähnlichen Betriebssystemen eingesetzt werden und unterstützt verschiedene Datenbanksysteme zur Speicherung der Daten.

Zenoss Zenoss[10] ist ein Netzwerkmonitoringsystem, das auf dem Zope Application Server, einer in Python geschriebenen Plattform für die Realisierung von Webanwendungen, basiert. Es ermöglicht die Übersicht der gesamten IT-Infrastruktur angefangen von Netzwerkgeräten bis hin zu Applikationen. Es bietet automatische Erkennung und Inventarisierung von Netzwerkknoten in eine Datenbank, Überwachung der Verfügbarkeit, graphische Darstellung von Leistungskurven sowie verschiedene Alarmierungen. Bedienung und Verwaltung erfolgen über den Browser und eine Webschnittstelle. Die Software kann unter verschiedenen Linux-Betriebssystemen eingesetzt werden und nutzt dabei eine Reihe anderer Open-Source-Anwendungen.

4.1.4 Nagios und Varianten

Das Monitoringsystem mit den derzeit meisten Installationen und Anwendern weltweit stellt wohl Nagios dar. In der Selbstdarstellung wird es allgemein als Industriestandard für Monitoringssysteme bezeichnet (Mie11). Gleichzeitig ist es ein System, aus dem heraus eine Vielfalt von neuen Projekten und Produkten entstanden ist. Das sind sowohl Lösungen, die Nagios mit anderen Open-Source-Komponenten integrieren als auch aus dem frei verfügbaren Code neu entstandene oder nur die Architektur übernehmende Projekte.

[8] URL: `http://www.zabbix.com/`
[9] SSH: Secure Shell
[10] URL: `http://www.zenoss.org/`

Nagios selbst ist aus dem Vorläufer NetSaint hervorgegangen und im Kern Open-Source-Software. Es besteht aus dem sogenannten Core, der selbst keine Überwachung übernimmt, sondern diese organisiert und verwaltet. Dazu wird die zu überwachende Infrastruktur in Form einer objektorientierten Konfiguration abgebildet und gegenseitig in Beziehung gesetzt. Die eigentliche Überwachung übernehmen verschiedene Module oder Plugins. Dadurch ist das System sehr flexibel und lässt sich einfach an spezielle Überwachungsaufgaben anpassen. Das mittlerweile erreichte hohe Niveau der Standardisierung ermöglicht einen sehr universellen Einsatz.

Die Konfiguration erfolgt mit einfachen Textdateien, die wiederum durch Zusatztools auch grafisch bearbeitet werden können. Ebenso gibt es für die Bedienung und Verwaltung des Systems verschiedene Oberflächen, in der Regel wird die Webschnittstelle über den Browser benutzt. Bei entsprechend konfigurierten Ereignissen erfolgt eine Benachrichtigung über verschiedene Wege. Darüber hinaus können als Reaktion auch Aktionen gestartet werden.

Für die Überwachung können verschiedene Methoden eingesetzt werden. Das reicht von der Nutzung verschiedener Netzwerkprotokolle über die Ausführung von Plugins auf den Hosts über Fernzugriff bis zum Einsatz von speziellen Agenten. Auch hier ist inzwischen neben den offiziell enthaltenen Plugins eine Vielzahl von zusätzlichen Tools entstanden. Die Software läuft unter verschiedenen Unix-ähnlichen Betriebssystemen und ist in vielen Distributionen als fertiges Paket enthalten.

Neben Nagios sollen exemplarisch folgende weitere von Nagios abgeleitete Projekte genannt werden:

Icinga ist ein 2009 abgespaltener Nagios-Fork, der eine modernere Weboberfläche bietet, zusätzliche Datenbanken unterstützt und einen schnelleren Entwicklungsprozess aufweist. Die ursprünglichen Nagios-Entwickler kommen vorwiegend aus dem deutschsprachigen Raum.[11]

Shinken ist eine von Jean Gabès ebenfalls 2009 begonnene Neuprogrammierung des Nagioskerns, die einige Mängel beheben sollte.[12]

Naemon ist ein 2013 vom ehemaligen Hauptentwickler Andreas Ericson veröffentlichter Fork des Nagioskerns.[13]

OMD bündelt verschiedene, ausgewählte Teile der Nagioswelt zur Vereinfachung des Einsatzes als Distribution und stellt ein geschlossenes System dar.[14]

Op5 Monitor ist eine durch die Firma op5 AB weiter entwickelte Version, die verschiedene

[11] URL: `http://www.icinga.org/`
[12] URL: `http://www.shinken-monitoring.org/`
[13] URL: `http://www.naemon.org/`
[14] URL: `http://omdistro.org/`

Erweiterungen zur Verbesserung bietet, wird als geschlossenes System angeboten.[15]

Opsview ist eine durch die Firma Opsview Limited weiter entwickelte Variante, die verschiedene freie Technologien integriert und vor allem eine verbesserte Weboberfläche bietet.[16]

openITCockpit ist ein Nagios-basiertes Monitoring-Framework, das sich an den ITIL-Prozessen orientiert und Monitoring mit Ticketing, Cloud Computing und Konfigurationsmanagement verbindet.[17]

Check_MK ist aus der Entwicklung eines Plugins entstanden, das die Prüfung mehrerer Services vereinfachen sollte und hat sich mittlerweile zu einem eigenen System entwickelt.[18]

4.1.5 Auswahl

Für die bisherigen Untersuchungen wurde eine Distributionslösung basierend auf Nagios verwendet (Fully Automated Nagios FAN[19]). Damit konnten bereits einige Erfahrungen gesammelt werden. Allerdings stellt die zugrundeliegende Linuxdistribution CentOS einen Exoten in der IT-Landschaft des TIETZ dar, hier wird sonst im wesentlichen openSUSE verwendet. Dazu kommen die Ereignisse um die derzeitige Weiterentwicklung von Nagios (vgl. (Lau10, S. 13)). Diese Lösung soll deshalb nicht weiter eingesetzt werden.

Die anderen bereits vorhandenen Lösungen sind zwar für ihren speziellen Einsatzzweck gut geeignet, bieten aber durch den Fokus auf nur ein Produkt keine Plattform- und Herstellerunabhängigkeit. Dazu kommt, dass durch Lizenzprobleme wie beim HP ProCurveManager oft nicht einmal diese Geräte vollständig abgebildet werden können.

Die kommerziellen Produkte kommen für einen Einsatz ebenfalls nicht in Frage, da sowohl für die Produkte selbst als auch für Windows als Betriebssystem Lizenzkosten anfallen. Entsprechende Mittel stehen nicht zur Verfügung, eine Beantragung könnte frühestens im nächsten Jahr berücksichtigt werden und überschreitet damit den zur Bearbeitung verfügbaren Zeitraum.

Durch die Lizenzpflicht entsteht aber noch ein weiteres Problem. Während der Bearbeitung wurde eine virtuelle Maschine mit Windows als Gastsystem aufgesetzt, um die Überwachung von Windowsservern zu testen. Die Maschine wurde geklont, um verschiedene Szenarien testen zu können, ohne das Ausgangssystem zu beeinträchtigen. Obwohl ein gültiger

[15] URL: `http://www.op5.com/`
[16] URL: `http://www.opsview.com/`
[17] URL: `http://www.it-novum.com/monitoring.html`
[18] URL: `http://mathias-kettner.de/check_mk.html`
[19] URL: `http://www.fullyautomatednagios.org/`

Lizenzschlüssel vorhanden war, musste die Maschine danach jedesmal neu freigeschalten und registriert werden, was unnötigen zusätzlichen Aufwand erforderte.

Bild 4.1: Fälschungshinweis von Windows

Diese Probleme entfallen beim Einsatz von Open-Source-Produkten. Die betrachteten Systeme erfüllen laut den Beschreibungen alle funktionalen Anforderungen. Allerdings erfüllt ein System aus der Nagios-Klasse die Forderung nach Nachhaltigkeit am besten. Es sind ausreichend Alternativen gegeben, um nicht in eine Herstellerabhängigkeit zu geraten oder andererseits Stillstand befürchten zu müssen. Die Frage ist nur, welche Alternative gewählt werden soll.

Den Ausschlag gab der Besuch eines Vortrags während der Chemnitzer Linuxtage, der den Nagios-Ableger Icinga näher vorstellte. Durch den Vortrag bestätigte sich, dass das System die Anforderungen erfüllt. Es hat den Anspruch, vollständig kompatibel zu Nagios zu bleiben[20] und bietet außerdem zahlreiche neue Dinge, die eine Beschäftigung mit Icinga lohnen. Die Mängel, die zur Abspaltung von Nagios geführt haben, sind behoben und es ist als offenes System angelegt. Außerdem vereinfacht die Tatsache, dass viele Unterlagen und Dokumentationen in deutsch vorliegen, die Einarbeitung ungemein. Icinga erfüllt die gestellten Anforderungen und soll deshalb genauer untersucht werden.

[20] "Alles was mit Nagios geht, soll auch mit Icinga funktionieren"
Brendel, Jens-Christoph : Monitoring-Gipfel. ADMIN: IT-Praxis und Strategie.
URL: `http://www.admin-magazin.de/News/Monitoring-Gipfel/`

4.2 Icinga

4.2.1 Arbeitsweise

Die Arbeitsweise von Icinga ist stark objektorientiert. Die zu überwachende Struktur wird deshalb in einzelne Objekte zerlegt, denen bestimmte Eigenschaften zugeordnet werden. Die möglichen Objekte und ihre Beziehungen untereinander sind genau definiert. Die wichtigsten Objekte in Icinga sind (Bar09, S. 57 ff):

Host beschreibt den zu überwachenden Netzwerkknoten, das können Computer, Router oder andere Netzwerkgeräte sein, aber auch elektrische Sensoren, die Messdaten erfassen.

Service stellen die zu überwachenden Eigenschaften eines Hosts dar, das können "echte" Dienste sein wie ein Webserver- oder Mailserverdienst, Betriebsparameter wie Speicherauslastung oder Netzwerkverkehr oder Messdaten eines Sensors. Ein Service ist immer an einen Host gebunden, wobei es den gleichen Service auf mehreren Hosts geben kann.

Kommando ist ein Objekt, mit dem externe Programme aufgerufen werden. Das können Plugins zur Überwachung oder Benachrichtigung sein.

Kontakt enthält eine Person, die über bestimmte Ereignisse benachrichtigt werden soll. Außerdem wird über dieses Objekt die Darstellung in der Weboberfläche gesteuert. Die Person bekommt nur das zu sehen, für das sie als Kontakt eingetragen ist.

Neben diesen grundsätzlichen Objekten gibt es noch eine Reihe weiterer Objekte, die der Darstellung der Beziehungen in der beschriebenen Struktur dienen. So können Hosts, Services und Kontakte zusammengefasst werden und bilden dann Host-, Service- oder Kontaktgruppen.

Zur Festlegung von Zeiten, in denen eine Überwachung oder Benachrichtigung stattfinden soll oder nicht, können Zeitfenster definiert werden.

Wenn ein bestimmtes Objekt nicht ohne ein anderes funktionieren kann, können diese Beziehung als Abhängigkeiten definiert werden. Funktioniert beispielsweise ein Router nicht mehr, lassen sich alle Objekte in dem Netz hinter dem Router nicht mehr beobachten und das Monitoringsystem meldet sie dann als *nicht erreichbar* und statt als *nicht in Ordnung* und macht keine Abfragen mehr.

Durch die Objektorientierung beherrscht Icinga auch Vererbung. Dadurch können Eigenschaften von übergeordneten Objekten an untergeordnete Objekte weiter gegeben werden. Das vereinfacht die Konfiguration des Systems, muss aber andererseits in bestimmten Fällen entsprechend beachtet werden.

Eine vergleichbare Vereinfachung wie durch die Beschreibung der Infrastruktur als Objekte wird bei der Beschreibung des Zustandes der Objekte vorgenommen. Anstatt wie bei anderen Programmen den Verlauf genau darzustellen, werden wie bei einer Ampel nur die Zustände grün = *alles in Ordnung* (*OK*), gelb = *bedenklich* (*WARNING*) oder rot = *kritisch* (*CRITICAL*) unterschieden. Was als kritisch oder bedenklich gilt, wird in der Konfiguration der Objekte festgelegt.

Falls keine Aussage getroffen werden kann, etwa bei abhängigen Objekten, oder ein Fehler vorliegt, gibt es außerdem die Zustände *nicht erreichbar* (*UNREACHABLE*) beziehungsweise *unbekannt* (*UNKNOWN*).

Für eine etwas genauere Darstellung gibt es noch zwei weitere Zustände. Wenn ein Objekt nach einem Fehlerzustand wieder in Ordnung ist, wird es als *zurück nach Fehler* (*RECOVERY*) ausgewiesen. Ein Objekt, welches seinen Zustand über einen bestimmten Zeitraum wechselt, wird durch *Zustand wechselt* (*FLAPPING*) beschrieben.

4.2.2 Core

Icinga besteht aus verschiedenen Komponenten (siehe Bild 4.2). Die zentrale Komponente, der sogenannte Core, plant und veranlasst die Ausführung von Tests und empfängt und verarbeitet deren Ergebnisse. Anhand der Ergebnisse stellt der Core fest, welche Zustände die überwachten Objekte haben, stellt diese für die Darstellung bereit und veranlasst entsprechende Benachrichtigungen bei Änderung von Zuständen.

Dem Core ist dabei nicht bekannt, welche Kontrollen durchgeführt werden, um Hosts und Services zu testen. Ebenso kennt er nicht die Methoden, mit denen die Benachrichtigungen erfolgen. Er ruft die definierten Kommandos lediglich auf und verarbeitet deren Rückkehrwerte.

Alle anfallenden Informationen werden durch den Core in Logfiles und wahlweise in einer Datenbank gespeichert. Von dort können sie für die Darstellung auf einer Weboberfläche, zur Analyse in Form von Statistiken und Berichten oder als grafische Darstellung abgerufen werden. Informationen können Ausfallzeiten, Überwachungsergebnisse, Alarmverläufe, Befehlsprotokolle und ähnliches sein.

Über die Weboberfläche können durch den Bediener verschiedene Aktionen ausgelöst werden. Diese nimmt der Core entgegen und führt wiederum entsprechende Kommandos aus. Solche Aktionen können beispielsweise die Bestätigung von Alarmen, Kommentierung von Ereignissen, Planung von Ausfallzeiten oder das Aussetzen von Überprüfungen sein.

Der Core läuft auf dem Überwachungsserver als Dienst. Die Konfiguration wird in einem einfach zu verstehenden Format als Textdatei abgelegt und beim Start eingelesen. Diese

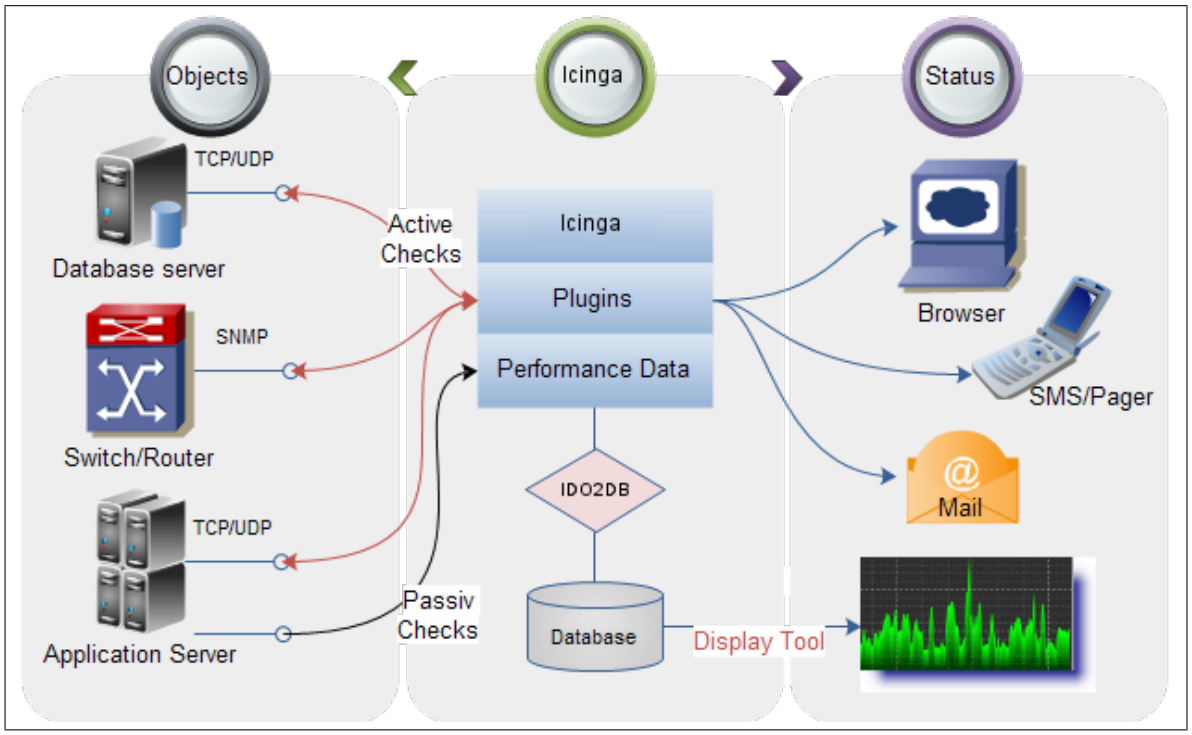

Bild 4.2: Komponenten Icinga

Konfiguration muss die Beschreibung jedes Objektes enthalten, das bei der Überwachung eine Rolle spielen soll.

Icinga läuft auf dem Überwachungsserver unter den meisten aktuellen Linuxdistributionen. Je nach Distribution sind verschiedene Programme und Bibliotheken notwendig, die installiert werden müssen. Unbedingt jedoch werden folgende Programme vorausgesetzt:

- Web Server
- Mailserver
- GCC Compiler

Die Installation kann über verschiedene Wege erfolgen. Am einfachsten ist die Verwendung der fertigen Pakete der jeweiligen Distribution. Da es im Moment noch eine sehr rasche Entwicklung gibt und in kurzer Zeit neue Versionen erscheinen, ist zu empfehlen, das Programm direkt aus den Quellen zu installieren. Dadurch können beseitigte Fehler und neue oder verbesserte Funktionen genutzt werden. Zum Testen gibt es außerdem fertige Appliances, die als virtuelle Maschine mit einer Virtualisierungssoftware oder auf einem Virtualisierungshost sofort benutzt werden können.

4.2.3 Plugins

Die zweite wichtige Komponente von Icinga sind die sogenannten Monitoring-Plugins. Sie führen die eigentlichen Überwachungsaufgaben aus. Es handelt sich dabei um kleine Programme oder Skripte, die für jeweils eine bestimmte Aufgabe zuständig sind. Die Plugins können in verschiedenen Programmiersprachen programmiert sein und müssen sich dabei an bestimmte Richtlinien halten, die in den sogenannten Monitoring Plugins Development Guidelines[21] festgehalten sind. Diese Richtlinien sorgen beispielsweise dafür, dass alle Plugins eine einheitliche Schnittstelle aufweisen und Aufruf sowie Auswertung immer nach dem gleichen Schema erfolgen. Auch die Funktionsfähigkeit in verschiedenen Umgebungen wird so gewährleistet.

Für die Integration in Icinga wird der Aufruf des Plugins in einem Kommandoobjekt gekapselt. Beim Aufruf des Kommandos können entsprechende Parameter für Schwellwerte oder -bereiche übergeben werden. Das Kommando liefert dann sowohl einen Rückgabewert als auch eine Zeile mit den Testergebnissen zurück. Sowohl die Übergabe der Parameter als auch der Resultate genügen einer einheitlichen Syntax. Folgende Rückgabewerte sind vereinbart:

Wert	Status	Beschreibung
0	OK	Service ließ sich überprüfen, es scheint alles in Ordnung zu sein
1	Warning	Service ließ sich überprüfen, Warnschwelle war überschritten
2	Critical	Service läuft nicht oder kritischer Schwellwert war überschritten
3	Unknown	Fehler beim Aufruf des Plugins

Die Rückgabewerte erscheinen als Text noch einmal auf der Ausgabezeile des Plugins. Diese kann weitere Ergebnisse der Überprüfung wie Informationstext oder Leistungswerte enthalten. Die Ausgabe der Leistungswerte folgt einem festgelegten Format und ermöglicht so eine vielseitige Auswertbarkeit.

Bei der Installation von Icinga sind eine Reihe von Plugins für grundlegende Tests bereits enthalten. Eine wichtige Erweiterung der Testmöglichkeiten lässt sich mit der Installation der Monitoring-Plugins[22] erreichen. Diese Plugins werden von einer eigenen Entwicklergemeinschaft entwickelt und funktionieren mit allen Nagiosvarianten und -ablegern. Sie bilden auch die Basis für die Installation auf den hier betrachteten Hosts.

Eine weitere Möglichkeit, um an Plugins für spezielle Aufgaben zu gelangen, stellt die Plattform MonitoringExchange[23] dar. Hier werden für fast alle denkbaren Fälle Lösungen angeboten und eine aktive Gemeinschaft sorgt für Unterstützung bei Fragen.

[21] URL: `https://www.monitoring-plugins.org/doc/guidelines.html`
[22] URL: `https://www.monitoring-plugins.org/`
[23] URL: `https://www.monitoringexchange.org/`

Nicht immer werden Plugins an diesen zentralen Stellen veröffentlicht. Sie lassen sich dann nur durch eine Suche im Internet oder Hinweise in der Literatur finden. Schließlich gibt es noch die Möglichkeit, eigene Plugins zu erstellen. Entsprechende Hinweise findet man unter anderem in (Bar09), (KS13) oder (Meh13).

4.2.4 Addons

Die dritte Komponente von Icinga bilden die sogenannten Addons. Das sind Zusatzprogramme für verschiedene Einzelaufgaben, die oft von eigenständigen Entwicklern erstellt und zur Verfügung gestellt werden. Sie sind kein Bestandteil des Icingacodes, für die Funktion des Monitoringsystems Icinga aber wesentlich.

NRPE oder Nagios Remote Plugin Executor ist ein Dienst zur Überwachung eines Linuxrechners. Er erlaubt es, die Monitoring-Plugins auf entfernten Maschinen auszuführen. Die Ergebnisse werden über SSL[24] an Icinga übertragen. Zur Nutzung müssen OpenSSL, NRPE und die Plugins auf dem entfernten Computer installiert sein, außerdem ist OpenSSL auf dem Icingaserver notwendig. Aus Sicherheitsgründen ist die Einrichtung eines speziellen Benutzers auf dem entfernten Host sinnvoll.

Das jeweils benötigte Plugin auf dem zu überwachenden Host wird von Icinga durch ein spezielles NRPE-Plugin aktiviert und dessen Ergebnisse abgerufen. Deshalb muss NRPE auch auf dem Icingaserver installiert sein. Da die Abfrage aktiv durch Icinga ausgelöst wird, spricht man von aktiver Prüfung.

NSClient++ ist ähnlich wie NRPE ein Dienst zur Überwachung eines entfernten Rechners, hier für Windowsrechner. Zur Nutzung genügt es, NSClient++ auf dem entfernten Rechner zu installieren, der Dienst enthält bereits verschiedene, unter Windows sinnvolle Prüfungen. Die Aktivierung der verschiedenen Prüfungen und der Abruf der Ergebnisse erfolgt von Icinga aus mit einem speziellen Plugin, es kann auch das NRPE-Plugin benutzt werden. Es handelt sich also ebenfalls um eine aktive Prüfung.

NSCA oder Nagios Service Check Acceptor ist ein Dienst auf dem Icingaserver, der Überwachungsergebnisse entgegen nehmen kann, die von entfernten Hosts geschickt werden und diese an Icinga zur Verarbeitung weitergibt. Da solche Prüfungen nicht von Icinga ausgelöst werden, spricht man von passiver Prüfung.

[24] SSL: Secure Socket Layer

PNP4Nagios ist ein Addon, mit dem die bei der Überwachung gewonnenen Performancedaten grafisch dargestellt und analysiert werden können (KS13). Die gesammelten Daten werden von PNP4Nagios in einer Round-Robin-Datenbank gesammelt. Für die richtige Funktion müssen deshalb neben dem Addon die rrd-Tools installiert werden. Die Ausgabe wird in die Weboberfläche von Icinga integriert und von hier aus verwendet.

NagVis ist ein Addon, mit dem die überwachten Hosts und Services als Symbole auf beliebigen Hintergründen wie Karten, Grundrissen oder Rackschemas dargestellt werden können (KS13). Dadurch wird eine intuitivere Übersicht erreicht. Die Ausgabe des Addons kann ebenfalls in die Weboberfläche integriert werden.

4.2.5 Konfigurationstools

Die Konfiguration von Icinga erfolgt wie bei Linux üblich mit einfachen Textdateien. Struktur und Syntax dieser Konfigurationsdateien entsprechen zur Wahrung der Kompatibilität der von Nagios. Erst mit Icinga 2 wird eine neue, weiter entwickelte Syntax eingeführt, um vorhandene Mängel zu beseitigen und fehlende Möglichkeiten zu ergänzen. Diese Version befindet sich derzeit in der Entwicklung und ist für einen produktiven Einsatz noch nicht zu empfehlen.

Eine Konfiguration mit textbasierten Dateien ist einfach zu handhaben und lässt sich auch durchführen, wenn nur eine Textkonsole wie beispielsweise im Serverraum zur Verfügung steht. Aus diesem Grund wird unter anderem in den Monitoring Plugins Development Guidelines gefordert, eine Bildschirmgröße von 80 mal 25 Zeichen zu respektieren. Für eine große Anzahl von Hosts und Services wird diese Form der Konfiguration aber schnell unübersichtlich, aufwendig und fehleranfällig. In diesem Fall kann ein Konfigurationstool genutzt werden, von denen es inzwischen mehrere gibt. Im folgenden werden einige ausgewählte kurz aufgelistet.

NagioSQL ist ein webbasiertes Tool, das die Konfigurationsdaten in einer MySQL-Datenbank speichert, aus der dann entweder lokal oder entfernt die Konfigurationsdateien erzeugt werden können. Die Konfigurationsobjekte lassen sich über ein Menü auswählen, dass alle wesentliche Objekte enthält. NagioSQL ist mit allen Icinga-Versionen kompatibel, es bietet Unterstützung für große und verteilte Umgebungen. Für die Installation werden Webserver, PEAR-Modul, MySQL-Datenbank, PHP[25] und Erweiterungen vorausgesetzt, ein Assistent vereinfacht die Installation. Das Tool wird im Moment nicht aktiv weiter entwi-

[25] PHP: PHP: Hypertext Preprocessor

ckelt, es gibt aber Hilfe bei Fragen und Fehlerbereinigungen[26].

NConf ist ebenfalls ein webbasiertes Tool und speichert die Konfigurationsdaten in einer MySQL-Datenbank. Es wurde für große verteilte Umgebungen konzipiert. Über ein Menü lassen sich die Konfigurationsobjekte aus Grund-, Zusatz-, Server- und Administrationsgruppen auswählen. NConf besitzt eine webbasierte Vorinstallations-Prüfung. Es ist mit allen Icinga-Versionen kompatibel. Für eine Installation werden Apache Webserver, PHP, MySQL-Datenbank und Perl vorausgesetzt.

Icingen ist ein Bash-Skript zur Konfiguration von Icinga. Es verwendet Vorlagen für Hosts und Hostgruppen und unterstützt verteilte Umgebungen. Es ist mit allen Icinga-Versionen kompatibel. Als Voraussetzungen sind SNMP-Plugins und catdoc notwendig. Das Projekt ist nicht mehr aktiv.

LConf speichert die Konfigurationsdaten in einer LDAP-Datenbank, aus der die Textkonfigurationsdateien im laufenden Betrieb exportiert werden können. Als Schnittstelle kann ein beliebiger LDAP-Browser verwendet werden. Die Verwendung von LDAP unterstützt die Strukturierung der Konfigurationsobjekte nach Host- oder Anwendungsgruppen, Orte oder Instanzen als Baumstruktur. LConf bietet Unterstützung für große verteilte Umgebungen und ist mit allen Icinga-Versionen kompatibel. Als Voraussetzungen für die Installation sind OpenLDAP-Server und Perl-Bibliotheken für LDAP notwendig, die Installation wird durch Shellskripte unterstützt. Das Addon enthält ein spezielles Modul für die Integration in die Icinga-Weboberfläche.

LConf empfiehlt sich durch die Integration in die Icinga-Weboberfläche und eine aktive Weiterentwicklung. Es lässt sich intuitiv bedienen und bietet durch das verwendete LDAP-Schema per Design eine logische Prüfung der Konfiguration. Gleichzeitig verstärkt LDAP die Hinwendung zu einer objektorientierten Denkweise und ermöglicht damit die Trennung von Struktur und Konfiguration. Deshalb wird dieses Tool für die Konfiguration eingesetzt.

4.3 Testszenario

Einen wesentlichen Teil bei der Bearbeitung des Themas nahm die Beschäftigung mit dem zu installierenden Monitoringssystem, dessen Zusammenspiel mit den Plugins und der Verwendung von Addons in einer realen Umgebung ein. Es sollte vor allem dem Kennenlernen

[26]URL: `http://www.nagiosql.org/forum8/general-questions/3308-is-the-nagiosql-\ \project-inactive.html`

des Systems, aber auch möglicher Fehlerquellen und Stolpersteine dienen. Da es sich um die Einführung eines komplexen Systems handelt, das schließlich die gesamte IT-Infrastruktur umfasst, sollte das nicht in der produktiven Umgebung geschehen. Stattdessen wurden diese Tests zuerst in einer kleineren Testumgebung modellhaft durchgeführt. Für die Testumgebung wurde dazu ein Icinga-Monitoringsystem auf einem realen Rechner eingerichtet, außerdem wurde eine fertig eingerichtete Icinga-Appliance in einer virtuellen Maschine genutzt.

Für den realen Rechner wurde ein Raspberry Pi, Modell B mit dem Betriebssystem Raspbian verwendet. Dieser Rechner ist für Lehr- und Lernzwecke entwickelt worden, aus diesem Grund äußerst kostengünstig zu bekommen und besitzt eine SD-Karte als zentrales Speichermedium. Durch die Verwendung verschiedener SD-Karten können damit sehr einfach verschiedene Konfigurationen miteinander verglichen werden. Das eingesetzte Betriebssystem entspricht dem in der Icinga-Dokumentation[27] als Referenz dienenden Debian. Dadurch gleichen viele Schritte bei der Installation dem dort beschriebenen Vorgehen. Bei der Umsetzung auf openSUSE muss das später an die dort verwendeten Installationswerkzeuge angepasst werden.

Auf dem Testsystem wurden Icinga, Icinga-Web sowie LConf installiert und eingerichtet. Die Verwendung von Icinga-Web wurde wieder rückgängig gemacht, da diese Oberfläche auf dem Raspberry Pi eine zu hohe Last erzeugt. Deshalb wurde in einem zweiten Schritt eine virtuelle Maschine mit einer Icinga-Appliance aufgesetzt, die diese Probleme nicht aufwies. Allerdings läuft die virtuelle Maschine nur solange wie der als Virtualisierungshost genutzte Arbeitsplatzrechner, was sie für Untersuchungen des Verhaltens im Dauerbetrieb ungeeignet macht. Dafür wurde weiterhin der durchlaufende reale Rechner mit dem Raspberry Pi verwendet. Auf der virtuellen Maschine konnten dafür die verschiedenen Weboberflächen und insbesondere der Umgang mit dem Konfigurationswerkzeug LConf ausprobiert werden.

Da verschiedene Varianten der Appliance zur Auswahl standen, wurde eine unter openSUSE laufende Version ausgewählt. So konnten bereits in der Testumgebung verschiedene Icingainstallationen ausprobiert und die gemachten Erfahrungen beim Einrichten der produktiven Umgebung umgesetzt werden, was sich damit wesentlich einfacher gestaltete.

Als Überwachungsobjekte wurden die im lokalen Netz erreichbaren Geräte genutzt und für das Monitoring eingerichtet. Weitere benötigte Objekte ließen sich als virtuelle Maschinen aufsetzen. Dabei wurden verschiedene Varianten getestet und Durchführung und Ergebnisse entsprechend dokumentiert, um sie bei der Umsetzung im produktiven Icingasystem berücksichtigen zu können.

In der Testumgebung standen natürlich nicht alle in der späteren realen Umgebung vor-

[27] URL: `http://docs.icinga.org/`

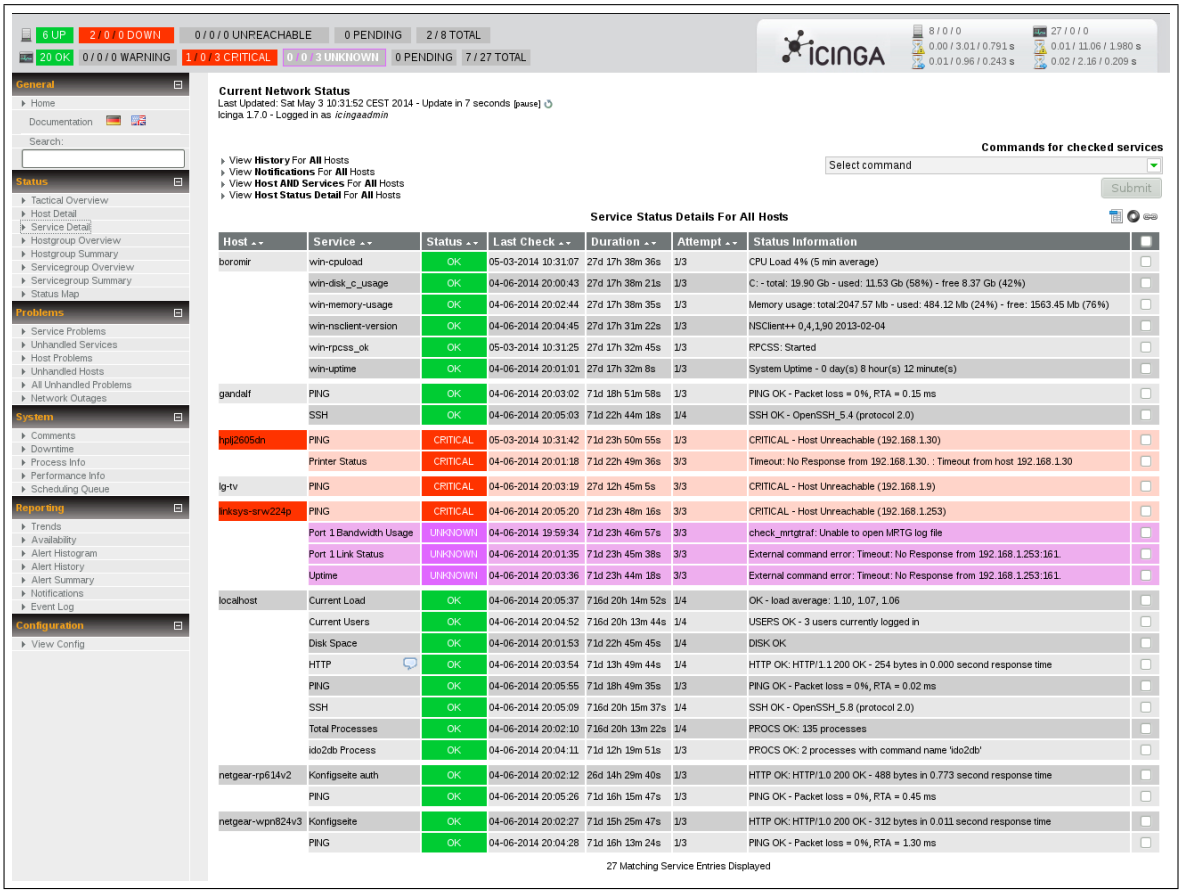

Bild 4.3: Testumgebung Icinga

kommenden Objekte zur Verfügung. Das ist aber auch nicht notwendig, da sich die Überwachungsmethoden auf einige typische Arten zurückführen lassen. Als wichtigste Objekte standen sowohl Rechner unter Linux als auch unter Windows zur Verfügung, außerdem Netzwerkgeräte. Damit konnten die wichtigsten Typen prinzipiell ausprobiert und getestet werden. Sie werden im folgenden näher erläutert.

4.3.1 aktive Prüfung

Standardmäßig werden von Icinga aktive Prüfungen durchgeführt. Das heißt, das Monitoringsystem fragt das zu überwachende System regelmäßig ab, ob es erreichbar ist und die definierten Dienste entsprechend funktionieren. Dieses Verfahren wird auch Polling genannt. Es ist für die meisten Überwachungsaufgaben empfehlenswert, insbesondere für Geräte und deren Dienste, die im Dauerbetrieb laufen und ständig zur Verfügung stehen müssen. Abhängig von den konkreten Erfordernissen werden die Abfrageintervalle und die Bewertung der Prüfungsergebnisse festgelegt.

Prüfung externer Dienste Allgemein im Netzwerk verfügbare Dienste lassen sich prüfen, indem über das Netzwerk eine Verbindung zu dem Dienst aufgebaut und eine entsprechende Testabfrage gestartet wird. Beispiele dafür sind DNS, SSH, Web- oder Maildienste. Die Prüfung externer Dienste sagt nichts über die Ursache von Ausfällen aus, sie werden nur als Blackbox betrachtet. Dafür erfordert diese Prüfung keine weiteren Voraussetzungen, sofern der entsprechende Dienst über das Netzwerk erreichbar ist. Falls der zugehörige Host in einem Netzsegment aufgestellt ist, das vom Monitoringsystem nicht direkt erreichbar ist, muss die Erreichbarkeit entsprechend eingerichtet werden. Bei der Anforderungsanalyse konnten diese Fälle bereits identifiziert werden.

Prüfung interner Dienste Viele zur Beurteilung des Hostzustandes wichtigen Eigenschaften sind von außen nicht zugänglich, zum Beispiel Systemressourcen und Leistungsdaten. Der Blick auf die lokalen Ressourcen erlaubt eine Korrektur von Problemen, bevor sie nach außen sichtbar werden. Um solche Eigenschaften in das Monitoring einzubinden, werden sie als Service betrachtet. Die Prüfung muss innerhalb des Gerätes erfolgen und das Ergebnis nach außen transportiert werden. Der zu überwachende Host muss entsprechend eingerichtet sein. Je nach Gerätetyp gibt es dafür verschiedene Möglichkeiten.

SNMP wurde speziell für das Management von Netzwerkgeräten entwickelt. SNMP ist für die meisten netzwerktauglichen Geräten verfügbar oder lässt sich entsprechend einrichten. Da ein entsprechender Dienst laufen muss, der sehr schlank ist und im Netzwerk über UDP abgefragt werden kann, ist SNMP sehr ressourcenschonend. Auf Netzwerkgeräten ist es oft die einzige Möglichkeit, um an interne Leistungsdaten zu gelangen. Problematisch sind die fehlende Verschlüsselung der Datenübertragung und die nur rudimentäre Zugangsprüfung, zumindest in den Versionen 1 und 2c. Diese ist erst in der wenig verbreiteten Version 3 vorhandenen.

Wenn SNMP nur mit der vorhandenen MIB verwendet wird, ist die Nutzung auf das eingeschränkt, was der Hersteller vorgesehen hat. Dafür muss auf dem Host nichts weiter eingerichtet werden und die Prüfung kann wie bei einem externen Dienst über eine Abfrage von außen erfolgen.

Eine andere Möglichkeit ist die Nutzung von SNMP Extends. Dazu wird der SNMP-Dienst auf dem Host entsprechend eingerichtet. Im MIB-Baum wird ein spezieller Zweig angelegt, in dem die Ergebnisse der internen Prüfungen abgelegt werden. Die Prüfungen können durch systemeigene Programme erfolgen, aber auch durch die Monitoring-Plugins. Diese müssen vorher auf dem Host installiert sein. Diese Variante ist sehr elegant, da sie sehr flexibel und ausbaubar ist und kaum Last auf dem Host erzeugt.

Beide Varianten wurden in der Testumgebung ausprobiert und haben sich als funktional

erwiesen. Aus der Anforderungsanalyse sind entsprechende Anwendungsfälle ablesbar. Da SNMP bisher nicht verwendet wird, ist eine Erreichbarkeit von Hosts nicht von vornherein gegeben. Das ist besonders bei solchen Hosts zu beachten, für die es keine Alternative zu SNMP gibt. Das sind alle Netzwerkgeräte wie Router oder Switche sowie die Drucker.

Für Linuxserver stellt die Nutzung von SNMP Extends eine elegante Möglichkeit dar, die wenig Last erzeugt und die Möglichkeit der Trennung zwischen Monitoring- und Serververwaltung bietet[28].

SSH stellt eine verschlüsselte und authentifizierte Verbindung zwischen zwei Hosts zur Verfügung. Über das Protokoll können Kommandos auf dem entfernten System ausgeführt werden. Das können systemeigene sein, aber auch Aufrufe der Monitoring-Plugins.

Der SSH-Zugriff wird in der Regel auf allen Hosts, auf denen er verfügbar ist, bereits für die Systemverwaltung genutzt, muss also nicht extra installiert werden. Aus diesem Grund ist die Firewall bereits entsprechend eingerichtet, auch in Netzsegmente, die ansonsten für andere Zugriffe gesperrt sind. Eine Erreichbarkeit der Hosts ist damit gegeben.

Der Zugriff über SSH hat auch Nachteile. So verursacht SSH auf dem Host mehr Last als beispielsweise SNMP, da es einen größeren Overhead durch die Verschlüsselung und die Verwendung von TCP gibt. Die notwendige Verwendung von Schlüsseln ohne Passphrase verringert die Sicherheit, deshalb ist für das Monitoring unbedingt ein spezieller Nutzer ohne weitere Rechte einzurichten. Aufgrund der Public-Key-Authentifizierung ist SSH empfindlich gegenüber Änderungen der Schlüssel, zum Beispiel wenn der Host neu installiert wird.

Am ehesten bietet sich diese Möglichkeit für die Überwachung von Linuxservern an. Hier stört die zusätzliche Last kaum und die notwendigen Voraussetzungen sind bereits vorhanden oder können einfach ergänzt werden.

NRPE, NSClient++ sind spezielle Agenten auf dem zu überwachenden Host, die die Prüfungen übernehmen und für die Verbindung zum Monitoringserver sorgen. Sie werden vor allem für die Serverüberwachung eingesetzt. Die Agenten erfordern eine Installation auf dem Host und müssen als Dienst laufen.

NRPE wird zur Überwachung von linuxbasierten Hosts verwendet. Die Verbindung zwischen Icinga und dem Agenten erfolgt verschlüsselt über SSL und verwendet spezielle Ports. Auf dem zu überwachenden Host müssen OpenSSL, NRPE und die Monitoring-Plugins installiert sein, auf dem Icingaserver OpenSSL. Eine Installation kann schwierig werden, wenn auf Produktivmaschinen kein Compiler installiert ist. Es empfiehlt sich, auf

[28] URL: `https://blog.netways.de/2014/03/28/icinga-2-und-auto-discovery-mit-snmp/`

dem überwachten Host aus Sicherheitsgründen einen gesonderten Benutzer einzurichten. NRPE erzeugt weniger Last auf dem Host als die Verwendung von SSH, ist dafür etwas aufwendiger bei der Einrichtung. Ein detaillierter Vergleich ist auf dem Monitoring-Portal zu finden[29].

NSClient++ wird zur Überwachung von windowsbasierten Hosts verwendet. Für die Verbindung zu Icinga können verschiedene Ports und Modi verwendet werden. Auf dem zu überwachenden Host muss NSClient++ installiert und als Dienst gestartet werden. Weitere Installationen sind nicht notwendig, die Prüfbefehle sind bereits enthalten. NSClient++ kann auch als NRPE-Dienst fungieren, die Verwendung der Windowsvariante von NRPE empfiehlt sich nicht.

Der Aufruf der Agenten kann grundsätzlich auf zwei verschiedene Arten erfolgen. Entweder werden die Prüfbefehle mit allen Parametern auf dem überwachten Host konfiguriert und von Icinga nur aufgerufen. Die Einstellung von Schwellwerten und Parametern erfolgt dann auf dem geprüften Host. Oder die Prüfung und deren Parameter werden von Icinga übergeben, dann können diese an zentraler Stelle gepflegt werden. Für den geplanten Einsatz wird die zweite Variante bevorzugt, um den Vorteil von Templates für eine Gruppe von Hosts nutzen zu können.

Während es bei der Überwachung von windowsbasierten Servern nur wenig Spielraum gibt, ist die Entscheidung bei linuxbasierten Servern etwas schwieriger. Hier stehen mit SNMP, SNMP Extends, über SSH und NRPE mehrere Möglichkeiten zur Auswahl. Aufgrund der einfachen Einrichtung und der kleinstmöglichen Änderung am überwachten System wird im folgenden auf die Prüfung über SSH orientiert.

4.3.2 passive Prüfung

Im Gegensatz zur aktiven Prüfung testet bei passiver Prüfung das entfernte System selbständig den Status und sendet die Ergebnisse oder kritischen Ereignisse an das Monitoringsystem. Dafür muss auf dem Monitoringsystem ein spezieller Dienst laufen, der auf die Übertragung der Ergebnisse wartet. Ein solcher Dienst ist NSCA.

NSCA wird vor allem für verteiltes Monitoring verwendet. Hier werden mehrere Icingaserver eingesetzt, um beispielsweise eine Lastverteilung auf mehrere Server bei sehr vielen zu überwachenden Systemen zu ermöglichen oder um bei getrennten Netzen wie beim Einsatz eines Proxy nur einen Zugang zu benötigen. Die verteilten Icingaserver prüfen selbständig die ihnen zugeteilten Hosts und Services und senden die Ergebnisse an den

[29] URL: `http://wiki.monitoring-portal.org/nagios/howtos/nrpe_vs_ssh`

zentralen Icingaserver. Bei entsprechender Konfiguration lässt sich damit auch ein hochverfügbares Monitoringsystem aufsetzen.

Im vorliegenden Rahmen werden zunächst keine passiven Prüfungen verwendet.

4.4 Überwachungsplanung

4.4.1 Überblick

Nach Auswahl und Test des Monitoringsystems lassen sich im Vergleich mit den Ergebnissen aus der Anforderungsanalyse eine Reihe von grundlegenden Vorgaben für die Umsetzung in die produktive Umgebung ableiten. Für die praktische Bearbeitung sollen sie mit drei verschiedenen Hilfsmitteln festgehalten werden.

Zu beachtenden Randbedingungen und Grundeinstellungen werden als Richtlinien zusammengefasst und im Wiki dokumentiert. Dort können diese als Standard für die Bearbeitung nachgeschlagen und gepflegt werden.

Wiederkehrende Abläufe werden als Workflows dokumentiert und in Form von Checklisten im Wiki abgelegt.

Für die Parametrisierung der Prüfungen werden in LConf entsprechende Vorlagen angelegt.

Durch diese Standardisierung von Abläufen und Einstellungen lässt sich die Erstellung der Konfiguration deutlich vereinfachen.

Vor dem Einrichten des ersten Hosts muss das Monitoringsystems installiert und grundlegend eingestellt werden. Danach erfolgt die Übernahme von Hosts und Services in mehreren Schritten:

- Hosts und Services identifizieren
- Hosts zugänglich machen
- Hosts für Monitoring vorbereiten
- Hosts in LConf einrichten
  - in Struktur einsortieren und eintragen
  - schrittweise Überwachung aktivieren
  - zusätzliche Services einrichten
- Benachrichtigung einrichten
- Betrieb beobachten

Dieser Ablauf wiederholt sich, bis alle Überwachungsobjekte eingepflegt sind. Er ist auch gültig bei späteren Änderungen und Erweiterungen. Die einzelnen Schritte werden im folgenden näher erläutert und die für jeden Schritt abgeleiteten Vorgaben wie Richtlinien, Abläufe und Vorlagen exemplarisch dargestellt.

4.4.2 Installation Monitoringssystem

Die Installation von Icinga erfolgte in einer virtuellen Maschine unter ESXi. Es wird zwar empfohlen, für den Icingaserver dedizierte Hardware einzusetzen, die möglichst wenig Berührungspunkte mit der restlichen Infrastruktur hat (Lau10, S. 17–18). Allerdings liegt die Zielrichtung bei dieser Arbeit ein wenig anders und es erscheint vertretbar, diesen Weg zu gehen. Vor allem entstehen im Moment keine weiteren Kosten für Hardware, da entsprechende Ressourcen vorhanden sind. Wenn es zu einem späteren Zeitpunkt erforderlich oder gewünscht ist, kann die virtuelle Instanz mit allen Einstellungen auf andere Hardware umziehen.

Als Betriebssystems wird openSUSE verwendet, es wurde zunächst mit dem Schema *minimale Serverauswahl* installiert. Danach wurden die Schemata *Web-* und LAMP-Server sowie *grundlegende Entwicklungsumgebung* ergänzt. Damit sind die grundlegenden Voraussetzungen gelegt. Für Icinga und LConf sind außerdem die Pakete Net-SNMP und OpenLDAP sowie einige Bibliotheken notwendig. Diese wurden ebenfalls installiert. Nach Anlegen eines Benutzers und einer Gruppe *icinga* wurde das Monitoringssystem aus den Quellen installiert. Folgende Pakete waren dafür erforderlich: Icinga und Icinga-Web, LConf, LConf für Icinga-Web, NRPE sowie die Monitoring-Plugins. Bei allen Paketen wurden jeweils die zum Installationszeitpunkt aktuellen Versionen ausgewählt.

Nach der grundlegenden Einrichtung und dem Test aller Dienste ist das Monitoringsystem betriebsbereit. Da bei der Installation von Icinga eine Basiskonfiguration zur Überwachung des lokalen Rechners eingerichtet wird, konnte das System bereits in Betrieb genommen werden. Es überwacht zu diesem Zeitpunkt allerdings nur sich selbst.

4.4.3 Auswahl Hosts und Services

Der nächste Schritt ist die Identifizierung der einzurichtenden Hosts und Services. Dafür werden die Ergebnisse aus der Anforderungsanalyse verwendet. Die dort vorgenommene Klassifizierung lässt sich gut mit dem in LConf verwendeten baumartigen LDAP-Schema darstellen (siehe Bild 4.4). Bei der Analyse wurden einige Objekte aus inhaltlichen Gründen mehreren Klassen zugeordnet. Die streng hierarchische Gliederung bei LDAP erlaubt allerdings nur die Einordnung in einem Strukturzweig. Deshalb wird bei der Strukturzuordnung

eine Klasse als Hauptklasse festgelegt.

Alle Klassen, denen die Objekte zugeordnet sind, werden außerdem als Hostgruppen definiert. So werden die Hosts in der Weboberfläche, Berichten und grafischen Auswertungen übersichtlicher dargestellt. Außerdem werden allen Klassen über Vorlagen spezifische Prüfungen und Parameter zugeordnet.

Servicegruppen werden für solche Dienste gebildet, die in der Anforderungsanalyse als wichtig erachtet wurden und für die eine Überwachung als Service möglich ist. Die Services sind dabei den Hosts zugeordnet, die den jeweiligen Dienst erbringen.

Die Klasseneinteilung aus der Anforderungsanalyse spiegelt sich also an drei Stellen in der Konfiguration wieder:

- in der Struktur
- in den Host- und Servicegruppen
- in den Vorlagen

Die Übernahme der in der Anforderungsanalyse ermittelten Objekte in das Monitoring erfolgt schrittweise entsprechend der festgelegten Priorität. Außerdem hängt die Übernahme insbesondere einzelner Services vom Aufwand ab, der für die Erstellung eines Prüfbefehls notwendig ist. Ein Beispiel dafür sind die Überwachung der Umgebungsparameter oder die Prüfung von Logereignissen.

Strukturentwurf Ausgehend von einem Wurzelobjekt werden zunächst Infrastrukturobjekte angelegt, die die Struktur weiter gliedern. Die Gliederung der Infrastrukturobjekte folgt dabei der Netzwerkstruktur. Das jeweils letzte Infrastrukturobjekt in einem Zweig wird von der Hauptklasse gebildet. Unterhalb der Hauptklasse werden die enthaltenen Hosts angelegt. Unterhalb jedes Hosts können noch hostspezifische Services angelegt sein. Die für alle Hosts einer Hauptklasse geltenden Services werden unterhalb der Hauptklasse angelegt. Über den Vererbungsmechanismus von LConf erben alle zugeordneten Hosts deren Eigenschaften.

Alle Vorlagen, zum Beispiel für die klassenspezifischen Services, werden als Templates außerhalb der Struktur angelegt und über Aliase in die Struktur verlinkt. Die Vorlagen werden hauptsächlich für die Hauptklassen benutzt, können aber in bestimmten Fällen auch einzelnen Hosts zugeordnet werden.

Bild 4.4 zeigt einen Ausschnitt aus der angelegten Struktur.

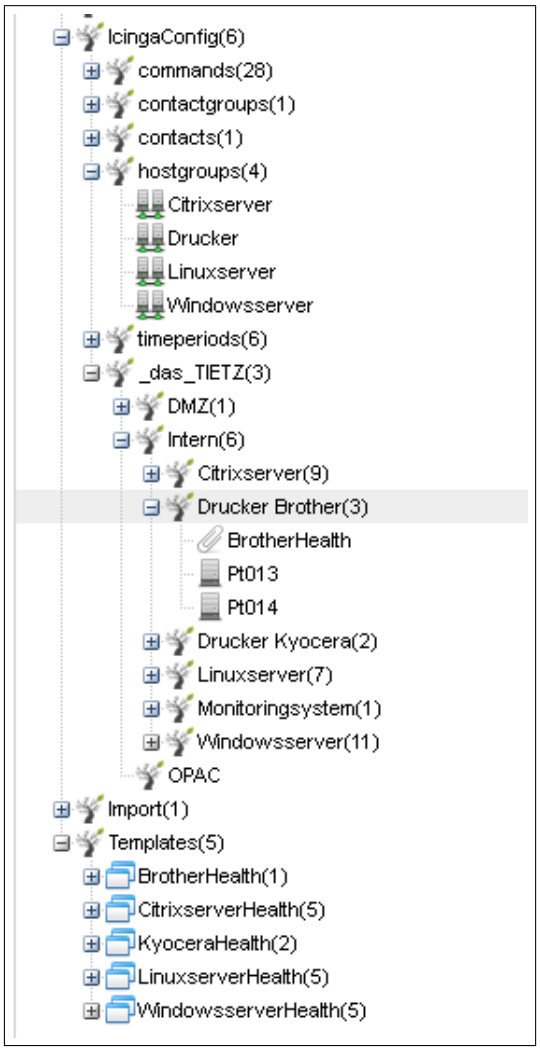

Bild 4.4: Struktur in LConf

Hostgruppen Die Bildung von Hostgruppen dient vor allem der besseren Übersichtlichkeit. Sie lassen sich von den gebildeten Klassen ableiten, wobei Klassen auch zusammengefasst werden können. Beispielsweise wird aus den verschiedenen Druckerklassen eine Hostgruppe Drucker gebildet, ebenso bei den Routern. Für Klassen, die nur ein Objekt enthalten, wird keine eigene Hostgruppe gebildet, das ist zum Beispiel beim Bandlaufwerk oder der USV[30] der Fall. Bei diesen Objekten sind eher die bereit gestellten Dienste wichtig, die besser über die Services abgebildet werden.

Für folgende Klassen wurden Hostgruppen angelegt:

- Windowsserver
- Linuxserver
- Citrixserver

[30] USV: unterbrechungsfreie Stromversorgung

- Router
- Switche
- Drucker
- ESXi-Server
- VM-Server

Servicegruppen Ähnlich wie die Hostgruppen erhöhen Servicegruppen die Übersichtlichkeit. So können wichtige Services, die von mehreren Hosts bereitgestellt werden, zusammengefasst werden.

Sinnvolle Servicegruppen können für folgende Funktionsklassen abgeleitet werden:

- Terminaldienste
- Active Directory
- Webserverdienst
- Mailserverdienst
- Applikationen

4.4.4 Zugänglichkeit herstellen

Um Hosts und Services überwachen zu können, müssen sie über das Netzwerk erreichbar sein. Für netzfähige Geräte und Rechner, die im gleichen Netzsegment wie der Monitoringserver angeschlossen sind, ist das in der Regel gegeben. Falls das nicht der Fall ist, gibt es je nach Situation verschiedene Lösungsansätze. Häufig genügt es, wenn in der Firewall eine entsprechende Verbindung freigeschaltet wird. Bei erhöhtem Sicherheitsbedarf kann es notwendig sein, in dem betreffenden Netz einen eigenen Monitoringsserver einzurichten, um die Zahl der Verbindungen zu reduzieren.

Ein Problem anderer Art gibt es bei der Verwendung von ISDN-Routern, wenn diese kostenpflichtige Verbindungen aufbauen. Dann kann es sein, dass vom Monitoringserver aus keine Verbindung möglich ist, da sie nur bei Bedarf von außen aufgebaut wird. Selbst wenn es erlaubt ist, verursacht die regelmäßige Abfrage zusätzliche Kosten. Auch hier muss eine individuelle Lösung gefunden werden.

Bei direkt erreichbaren Hosts ist dann keine ständige Zugänglichkeit gegeben, wenn sie nur zeitweise eingeschaltet sind. Damit Fehlalarme während der Ausschaltzeit vermieden werden, muss bei der Überprüfung oder der Benachrichtigung ein entsprechendes Zeitintervall eingetragen werden. Für Geräte, die unregelmäßig eingeschaltet sind und trotzdem geprüft

werden sollen, gibt es noch keine zufriedenstellende Lösung.

Bei den ausgewählten Hosts und Services ist die Zugänglichkeit jeweils gegeben und muss lediglich bei der Einrichtung der Verfügbarkeitsprüfung beachtet werden.

4.4.5 Host vorbereiten

Wie unter 4.3.1 beschrieben, sind viele zur Beurteilung des Hostzustandes wichtigen Eigenschaften nur von innen zugänglich. Für den notwendigen Zugriff müssen die Hosts entsprechend eingerichtet werden. Innerhalb einer Klasse ist der dafür erforderliche Ablauf gleich, auch zwischen verschiedenen Klassen bestehen große Ähnlichkeiten. Bei den gebildeten Klassen lassen sich drei verschiedene Prüfungsmethoden einrichten:

- die Monitoring-Plugins werden auf den Serverklassen eingerichtet, auf denen sich Programme installieren lassen
- SNMP wird bei den Klassen eingerichtet, die Netzwerkgeräte wie Router und Switche oder Drucker enthalten
- herstellerspezifische Schnittstellen werden bei den Klassen ESXi- und Fujitsu-Server benutzt

Da bei linux- und windowsbasierten Servern jeweils eine andere Übertragungsweise verwendet wird, unterscheiden sich die beiden Abläufe. Damit lassen sich vier grundlegende Ablaufpläne aufstellen.

Der prinzipielle Ablauf für einen linuxbasierten Server erfordert folgende Schritte:

- Einrichten eines Monitoring-Users
- Kopieren der Monitoring-Plugins
- Einrichten des SSH-Zugriffs

Die meisten Schritte lassen sich skripten, so dass sich der Aufwand für die Einrichtung weiter verringern lässt. Für einen windowsbasierte Server ist der Ablauf etwas anders:

- Installation von NSClient++
- Einrichten und Starten des Dienstes

Die Installation erfolgt toolgestützt[31], so dass sich auch hier der Aufwand verringern lässt.

Bei den Hosts, die über SNMP abgefragt werden, muss SNMP aktiviert und eingerichtet werden. Insbesondere ist der Zugriff zu beschränken und die Standardcommunity zu

[31] URL: `http://docs.nsclient.org/tutorial/core/index.html\#deploy-massively-manage`

ändern. Um die gewünschten Parameter abfragen zu können, werden die passenden Object Identifier benötigt. Sie müssen entweder über die Dokumentation oder eine Suche mit einem SNMP-Browser herausgefunden werden.

Die Abfrage der herstellerspezifischen Schnittstellen erfordert in der Regel ein spezielles Plugin. Die Schnittstelle muss aktiviert und entsprechend eingerichtet werden.

Die detaillierten Ablaufpläne einschließlich der benutzten Skripte sind als Checkliste im Wiki dokumentiert.

4.4.6 Überwachung einrichten

Die Überwachung der Hosts erfolgt schrittweise auf mehreren Ebenen. Erst die gesamten Antworten ergeben ein vollständiges Bild zur Beurteilung der Infrastruktur. Dementsprechend müssen mehrere Konfigurationsschritte ausgeführt werden, um Host und Services in das Monitoring aufzunehmen.

Verfügbarkeit Zuerst wird eine Prüfung, ob der Host verfügbar ist, eingetragen. Dafür wird die bei Icinga standardmäßige Host-Alive-Prüfung mit dem Ping-Kommando verwendet, welche über ICMP prüft, ob der Host antwortet. Gleichzeitig wird dadurch die Netzwerkverbindung zum Host überprüft. Wenn die Verwendung von ICMP nicht möglich ist, weil beispielsweise das Protokoll von der Firewall geblockt wird, wird alternativ geprüft, ob der SSH-Dienst antwortet. Die Prüfung wird direkt im Hostobjekt eingetragen, es handelt sich bei LConf um eine Pflichteintragung.

Health Dann wird die Beurteilung des Hostzustandes, sozusagen seiner “Gesundheit”, über die Prüfung von internen Parametern eingerichtet. Für diese Prüfung werden Meßgrößen wie Last, Anzahl der Prozesse und Nutzer sowie Speicher- und Festplattenbelegung verwendet. Die Meßgrößen und Parameter unterscheiden sich je nach Klasse und werden als Vorlage in einem LConf-Template abgelegt. Über Schwellwerte wird eingestellt, wann eine Warnung erfolgt oder eine Größe als kritisch eingestuft wird. Die Templates werden über Aliases in den Hauptklassenzweig verlinkt, die Health-Prüfung vererbt sich so an die enthaltenen Hosts. Wenn in einzelnen Fällen andere Parameter für diese Prüfung notwendig sind, kann die Prüfung unterhalb des betreffenden Hosts eingerichtet werden, die vererbten Parameter werden überschrieben.

Dienste Schließlich werden die Prüfungen aller auf dem Netzknoten laufenden oder ihm zugeordneten Dienste eingetragen. Das betrifft sowohl Prüfungen von außen, also aller Dienste, die nach außen verfügbar sein müssen, als auch Prüfungen von innen. So wird

zum Beispiel der Webserverdienst sowohl von außen auf Verfügbarkeit als auch von innen über den Status der laufenden Prozesse auf dem Webserver geprüft. Die Serviceprüfungen werden unter dem jeweiligen Hosts eingetragen, entweder direkt oder über ein Template, wenn Prüfungen für mehrere Hosts verwendet werden.

Gruppenzuordnung Nach dem Eintragen der Hosts und den zugehörigen Services erfolgt die Zuordnung zu den Host- und Servicegruppen. Die Zuordnung kann in LConf an verschiedenen Stellen vorgenommen werden, zur besseren Übersicht sollte es aber einheitlich an einer Stelle gemacht werden. Hier wird dafür die Auswahlmöglichkeit in den Host- und Servicegruppenobjekten benutzt.

Eltern-Kind-Beziehung Aus der Sicht des Monitoringsystems verläuft eine Netzwerkverbindung zu einem bestimmten Host oft über andere, ebenfalls im Monitoring eingetragene Hosts. Falls auf diesem Weg einer der Hosts einen Fehler aufweist, lassen sich alle Objekte dahinter nicht mehr erreichen und würden als fehlerhaft ausgewiesen. Um das zu vermeiden, muss über die sogenannte Eltern-Kind-Beziehung eingetragen werden, wie die Hosts miteinander verbunden sind. Im Hostobjekt ist dafür das parents-Feld vorgesehen. Wenn es zwischen Host und Monitoringsystem weitere Hosts gibt, wird hier der nächstliegende Host eingetragen. Dafür ist die Netzwerkstruktur aus Sicht des Monitoringsystems zu betrachten.

Bei den Switch- und Routerklassen ist die Abhängigkeit zu den anderen an das Netzwerk angeschlossenen Klassen offensichtlich und ergibt sich aus der vorliegenden Netzstruktur. Zu beachten sind aber auch weniger offensichtliche Beziehungen, so zwischen gestackten Switchen sowie zwischen ESXi-Servern und den darauf laufenden virtuellen Maschinen.

In Icinga gibt es noch eine weitere Möglichkeit, um Abhängigkeiten zwischen Hosts und Services zu definieren. Diese Möglichkeit wird im Moment nicht verwendet.

4.4.7 Benachrichtigung, Visualisierung

Für die Benachrichtigung ist bereits bei der Installation des Monitoringssystems ein Kontakt sowie eine Kontaktgruppe eingerichtet worden. Diese verwendet die Standardoptionen von Icinga und enthält eine speziell für diesen Zweck eingerichtete Mailadresse. Die Zustellung der Mails wird im Mailsystem des TIETZ dem Postfach des Systemadministrators zugeordnet. Weitere Benachrichtigungen per Mail sind zur Zeit nicht eingerichtet.

Die Beobachtung der Infrastruktur und die Bedienung des Monitoringsystems erfolgt vorrangig über die Weboberfläche von Icinga, wobei sowohl Icinga-Classic als auch Icinga-Web verwendet wird. Die Oberflächen unterscheiden sich in Ansicht und Bedienung voneinan-

der, wobei die neue Icinga-Web-Oberfläche mehr Möglichkeiten zur individuellen Konfiguration bietet. Für den Umgang damit ist aber entsprechende Erfahrung notwendig, die sich erst aus dem täglichen Betrieb ergeben wird.

Eine sinnvolle Ergänzung zur Weboberfläche im Browser ist die Verwendung einer App für mobile Geräte. Es gibt verschiedene Apps, getestet wurde aNag für Android. Die App wurde zwar für Nagios entwickelt, arbeitet aber sehr gut mit Icinga zusammen. Damit lässt sich das Monitoringsystem von unterwegs beobachten und falls erforderlich, bedienen.

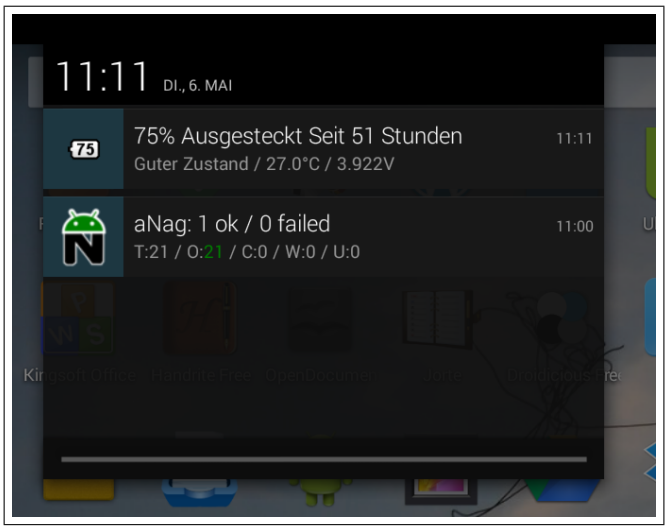

Bild 4.5: Benachrichtigung über aNag

4.5 Betrieb und Evaluation

4.5.1 Inbetriebnahme

Nach der Installation wurde das Monitoringsystem gestartet. Es überwacht zu diesem Zeitpunkt über die enthaltene Musterkonfiguration den Server und seine eigenen Prozesse, ist aber bereits voll funktionsfähig. Zu beachten ist, dass Icinga und alle dafür notwendigen Dienste auch nach einem Neustart des Servers automatisch gestartet werden. Dazu müssen entsprechende Einträge in der Startkonfiguration des Betriebssystems gemacht werden, bei der verwendeten openSUSE-Version wird das im entsprechenden Sysconfig-Zweig vorgenommen. Mit einem Neustart wurde der erfolgreiche Eintrag noch einmal überprüft.

Mithilfe der Musterkonfiguration von Icinga ließ sich die Funktionsfähigkeit von LConf überprüfen. Dazu wurden die Objekte aus der Musterkonfiguration über ein mitgeliefertes Skript in LConf importiert und waren danach sofort in LConf sicht- und bearbeitbar. Die Konfiguration von Icinga wurde nun so geändert, dass nur noch die über LConf erstellte Konfiguration wirksam ist. LConf wurde so eingestellt, das von der Weboberfläche aus der Export der Konfiguration in die Textdateien der Icingakonfiguration ausgelöst werden kann und ein Reload von Icinga zur Aktivierung erfolgt. Über die Weboberfläche und anhand der erzeugten Konfigurationsdateien wurde überprüft, ob es funktioniert.

Nach der Inbetriebnahme erfolgte entsprechend den aufgestellten Regeln nacheinander eine klassenweise Übernahme der Hosts und Services in das Monitoring. Dazu wurden zuerst einzelne Hosts aufgenommen und der Umfang nach erfolgreicher Übernahme allmählich vergrößert, bis alle Hosts einer Klasse übernommen waren. Nach einigen Tagen Betrieb folgte dann die nächsten Klasse.

Der Prozess der Übernahme von Hosts und Services ins Monitoring ist noch nicht abgeschlossen und wird sich nach dem Bearbeitungszeitraum dieser Arbeit weiter fortsetzen.

4.5.2 Wirksamkeit

Die Übernahme der bisher aufgenommenen Hosts erfolgte ohne größere Auffälligkeiten. Dafür erwies sich bereits an mehreren Fällen die Nützlichkeit des Monitoringssystems für die Systemadministration.

So wurde unmittelbar nach der Aufnahme bei einem Server eine Warnung bezüglich der Dateisystemauslastung angezeigt. Bei näherer Untersuchung stellte sich heraus, dass das Dateisystem zu klein angelegt war und vergrößert werden musste. Bei einem anderen Server wurde durch das kontinuierliche Monitoring eine hohe Speicherauslastung während der Nachtstunden entdeckt. Die Ursachen konnten daraufhin genauer untersucht werden. Beide

Probleme wären ohne das Monitoringsystem erst durch einen Ausfall bemerkt worden.

In einem weiteren Fall konnte durch die Alarmierung ein Routerausfall kurzfristig bemerkt und behoben werden, bevor er Auswirkungen auf den Betrieb zeigte. Der Ausfall blieb deshalb außerhalb der Systemadministration unbemerkt.

Die gezeigten Fälle deuten bereits darauf hin, dass sich das Monitoringsystem wie gefordert einsetzen lässt. Für eine fundiertere Aussage ist aber eine Beobachtung über einen wesentlich längeren Zeitraum notwendig.

4.5.3 Bewertung

Mit dem gewählten Open-Source-System und den gefundenen Einstellungen lassen sich die Forderungen nach geringem Aufwand und Komplexität recht gut erfüllen. Die Lösung ist wegen der fehlenden Lizenzkosten und der Möglichkeit, das System selbst einzurichten, sehr kostengünstig.

Durch die offene Architektur lässt sich die IT-Infrastruktur umfassend abbilden, für die meisten Probleme gibt es nachnutzbare Lösungen, eigene lassen sich leicht ergänzen. Die Vielfalt der gebotenen Lösungen erhöht zwar die Komplexität und muss deshalb durch die beschriebene standardisierte Vorgehensweise ausgeglichen werden.

Sämtliche benutzte Software wird aktiv weiterentwickelt und bietet über die gemeinsamen Wurzeln eine Reihe von Alternativen, so dass von einer nachhaltigen Lösung gesprochen werden kann. Die funktionalen Forderungen werden durch das System umfassend erfüllt.

5 Fazit und Ausblick

5.1 Zusammenfassung

Mit der vorliegenden Arbeit konnten die Anforderungen an die Überwachung der IT-Landschaft im “Das TIETZ” mit einem Open-Source-Monitoringsystem genauer spezifiziert werden. Die dafür notwendigen Daten wurden mit Methoden der Anforderungsanalyse gewonnen. Für das Anforderungsmanagement wurde ein wikibasiertes Werkzeug entworfen. Mit der bei der detaillierten Analyse gewonnenen tieferen Kenntnis des IT-Systems lassen sich bereits Verbesserungen bei der täglichen Arbeit der Systemadministration im TIETZ erzielen.

Mit Icinga wird eine für die vorhandene IT-Infrastruktur sinnvolle Variante aus einer Vielzahl von Möglichkeiten für das Monitoring ausgewählt und eine mögliche praktische Umsetzung in ein funktionierendes Monitoringsystem gezeigt. Damit wird die Voraussetzung für das Monitoring der IT-Infrastruktur im “Das TIETZ” geschaffen und der produktive Betrieb vorbereitet.

Die festgestellte Wirksamkeit der gewählten Lösung legt nahe, die begonnene Übernahme der IT-Infrastruktur in das Monitoring weiter zu führen und im täglichen Betrieb zu etablieren. Mit den erarbeiteten Richtlinien, Abläufen und Vorlagen ist eine einfache Integration vorhandener wie neuer Technik möglich. Durch die offene und verbreitete Architektur ist das System ausbau- und erweiterbar.

Die eingangs gestellten Fragen ließen sich damit beantworten und die angestrebten Ergebnisse wurden weitestgehend erreicht. Sie machen auf die Leistungsfähigkeit von Open-Source-Software aufmerksam und bestätigen die Nutzbarkeit in kommunalen Einrichtungen trotz notwendiger Kostenminimierung.

Die in der Projektskizze vorhergesagte steile Lernkurve bei der Beschäftigung mit Icinga hat sich bestätigt. Monitoring heterogener Systeme ist umfangreich und unübersichtlich, von dynamischer Entwicklung geprägt und bietet viele Möglichkeiten, sich in Detailfragen zu verlieren. Mehr als einmal erinnerte die Beschäftigung mit dem Thema an das Märchen vom süßen Brei.

Die Einarbeitung kann als gelungen bezeichnet werden und zeigt, eine weitere Beschäfti-

gung mit dem Thema ist lohnend und für die Systemadministration kleinerer Umgebungen unverzichtbar.

5.2 weiterführende Arbeiten

Aufgrund des umfangreichen Themas und der begrenzten Zeit sind einige Detailfragen offen geblieben. Die rasche Weiterentwicklung im IT-Bereich bringt außerdem ständig neue Anforderungen hervor, auf die im IT-Service-Management reagiert werden muss. Deshalb gibt es auf mehreren Ebenen Ansätze für weiterführende Arbeiten.

Zunächst ist es wichtig, das Monitoringsystem vollständig in den produktiven Betrieb zu überführen. Neben der vollständigen Übernahme der bereits spezifizierten Hosts und Services und dem Sammeln von Erfahrungen lassen sich folgende Arbeiten ausmachen:

- Verfeinern der Parameter und Schwellwerte
- Einbeziehen weiterer Geräte
- Überwachen anderer Dinge wie Logs oder Zertifikate[1]
- Berücksichtigen des Monitorings bereits bei der Beschaffung neuer Technik

Während der Bearbeitung kam es bei der Nutzung von Werkzeugen immer wieder zu Medienbrüchen. Es wurden für die Inventarisierung, das Anforderungsmanagement und die Monitoringkonfiguration Werkzeuge verwendet, die aber nicht zusammenarbeiteten. Durch die Integration der bereits existierenden Open-Source-Software ließe sich eine vollständige Unterstützung des IT-Service-Management-Prozesses erreichen.[2]

Durch das Monitoring und die damit verbundene Aufteilung der IT-Infrastruktur in viele kleine Teile verlieren diese ihren Zusammenhang. Mit einer geeigneten Verknüpfung von Prüfungen kann der Zusammenhang als sogenannte Business Process Views wieder hergestellt werden. Somit wird Monitoring auch für den Anwender interessant, denn er erfährt, ob das IT-System für seinen Anwendungsfall funktioniert.[3]

[1] URL: `http://word.bitly.com/post/74839060954/ten-things-to-monitor`
[2] URL: `http://www.netways.de/index.php?id=3444`
[3] URL: `http://my-plugin.de/wiki/de/projects/check_multi/process_views`

Literaturverzeichnis

[Bar09] BARTH, Wolfgang: *Nagios : System- und Netzwerk-Monitoring*. 2., aktualisierte und erw. Aufl., Jub.-Ausg. 10 Jahre Nagios. München : Open Source Press, 2009 (Roots reading). – ISBN 978–3–937514–91–8

[Des12] DESCOVICH, Philipp: *Die fünf größten Herausforderungen beim Netzwerk-Monitoring | it-administrator.de*, 2012. `http://www.it-administrator.de/themen/netzwerkmanagement/fachartikel/111010.html`. – Abruf am 04.03.2014

[Jur07] JURZIK, Heike (Hrsg.): *Monitoring : Probleme erkennen bevor sie entstehen*. Ausg. 2007. München : Linux New Media AG, 2007 (Linux-Magazin : Technical review ; 2). – ISBN 978–3–939551–04–1; 3–939551–04–X978–3–93955104–01

[Kau03] KAUFFELS, Franz-Joachim: *Lokale Netze*. 15. Aufl. Bonn : mitp-Verl., 2003 (Netzwerke). – ISBN 3–8266–0994–8. – Gb.

[Kra14] KRAAZ, Matthias: Abgeklopft: Requirements Engineering für eingebettete Systeme. In: *iX Developer* (2014), Nr. 2/2014, S. 62–65. – ISSN 3–944099–19–2, 978–3–944099–19–4

[KS13] KUCZA, Timo ; STAUDEMEYER, Ralf: *Das Nagios/Icinga-Kochbuch*. 1. Aufl. Beijing : OReilly, 2013 (Monitoring für Systemadministratoren). – ISBN 978–3–86899–346–2; 3–86899–346–0

[Lau10] LAUSSER, Gerhard: *Nagios - Das Praxisbuch: Open Source-Monitoring im Unternehmen*. Addison-Wesley Verlag, 2010

[Lim06] LIMONCELLI, Tom: *Zeitmanagement für Systemadministratoren : [Techniken, Strategien, Beispiele]*. Dt. Ausg., 1. Aufl. Beijing : OReilly, 2006. – ISBN 3–89721–465–2978–3–89721–465–1

[Lyo09] LYON, Gordon: *Nmap : Netzwerke scannen, analysieren und absichern; [official Nmap project guide]*. München : Open Source Press, 2009 (Roots reading). – ISBN 978–3–937514–82–6

[Meh13] MEHTA, Viranch: *Icinga Network Monitoring*. Packt Publishing, 2013. – ISBN 1783282290

[Mie11] MIES, Christian: Industriestandard fürs Monitoring. In: *ADMIN : IT-Praxis und Strategie* 4 (2011), Juni, Nr. 6, S. 21–22

[Mig12] *Migrationsleitfaden : Leitfaden für die Migration von Software*. [Elektronische Ressource], 4., überarb. Aufl. 2012. Berlin : Bundesministerium des Innern, Referat IT 2 (KBSt), 2012 (Schriftenreihe der KBSt). – Online–Ressource, 196 S. (pdf, 196 S. = 2 MB, Text). `http:`

//www.cio.bund.de/Web/DE/Architekturen-und-Standards/Migrationsleitfaden-und-Migrationshilfen/migrationsleitfaden_node.html. – Abruf am 27.04.2014

[MS05] MAURO, Douglas R. ; SCHMIDT, Kevin J.: *Essential SNMP : [help for system and network administrators]*. 2. ed. Beijing : OReilly, 2005. – ISBN 0–596–00840–6

[Pes12] PESCHLOW, Patrick: Die DevOps-Bewegung | codecentric AG. In: *Javamagazin* (2012), Nr. 1/2012, 12 S. https://www.codecentric.de/kompetenzen/publikationen/die-devops-bewegung/. – Abruf am 21.03.2014

[Sch10] SCHWANTNER, Holm: *Werkzeuge für die Systemadministration : Berichte Hauptpraktikum*, 2010

[Win09] WINKLER, Lutz: *Vorlesungsunterlagen Grundlagen der Kommunikationstechnik*. Mittweida, 2009

[Zie05] ZIEGLER, Matthias Clauß ; Thomas Müller ; C.: *Management und Sicherheit von IT-Services : Vorlesungsskript*. URZ der TU Chemnitz, 2005 http://www.tu-chemnitz.de/urz/lehre/msis/script05/. – Abruf am 17.04.2010

Selbstständigkeitserklärung

Ich erkläre, dass ich die vorliegende Arbeit selbstständig und nur unter Verwendung der angegebenen Literatur und Hilfsmittel angefertigt habe.

Mittweida, den 9. Mai 2014

_____

Holm Schwantner