Dualer Studiengang BWL

Schwerpunkt Wirtschaftsinformatik

Informatik-Anwendungen

- SNMP -

Sebastian Andersick (706953)
André Klahre (706678)
Christian Spohn (706844)
Anna-Maria Wulst (709004)

Betreuender Professor
Prof. Dr. Heinrich P. Godbersen
Lehrveranstaltung
Informatik Anwendung II (WS 2003/04)
Lehrveranstaltungs-Nr.
LV 7352

Eidesstattliche Erklärung

Hiermit erklären wir an Eides Statt, dass wir die vorliegende Arbeit selbstständig verfasst und keine anderen als die angegebenen Quellen und Hilfsmittel benutzt haben.

Sebastian Andersick _____

Andre Klahre _____

Christian Spohn _____

Anna-Maria Wulst _____

Berlin, 09.12.2003

Inhaltsverzeichnis

1 Einführung 1
2 Netzwerkmanagement 2
2.1 Definition 2
2.2 Netzwerkmanagement vs. Systemmanagement 2
2.3 Bereiche 3
2.4 Architektur 6
2.4.1 Komponenten 6
2.4.2 Kommunikation 8
3 SNMP - Grundlagen 10
3.1 Internet Management Framework 10
3.2 Managementinformationsbasis (MIB) 10
3.3 Structure of Management Information (SMI) 13
3.4 Protokoll SNMP 15
3.5 Sicherheits- und Administrationsfähigkeiten 15
3.6 ASN.1 15
4 SNMP - Entstehung 17
5 SNMP - Versionen 20
5.1 SNMPv1 20
5.2 SNMPv2 21
5.3 SNMPv3 24
6 SNMP - Nachrichten 25
7 SNMP - Sicherheitsaspekte 28
7.1 Gefahren für Netzwerksicherheit 28
7.2 Sicherheit von SNMPv1 29
7.3 Sicherheit von SNMPv2 29
7.4 Sicherheit von SNMPv3 30

8 Analyse der erzeugten PDU´s 33
8.1 PDU allgemein 33
8.2 SNMP-PDU 36
8.3 SNMP-PDU in einem Analyser 38
Quellenverzeichnis 42
9 Anhang A

Abkürzungsverzeichnis

Abb.	Abbildung
CMIP	Common Management Information Protokoll
DES	Data Encryption Standart
HEMS	High-Level Entity Management System
IAB	Internet Activities Board
IETF	Internet Engineering Task Force
IP	Internet Protocol
ISO	International Standards Organisation
MIB	Management Information Base
NM	Netzwerkmanagement
NMS	Network Management Station
OID	Object Identifier
OSI	Open Systems Interconnection
PDU	Protocol Data Unit
S	Seite
SGMP	Simple Network Gateway Monitoring Protocol
SMI	Structure of Management Information
SNMP	Simple Network Management Protokol
SNMPsec	Secure SNMP
SNMPv1	SNMP Version 1
SNMPv2	SNMP Version 2
SNMPv2*	SNMP Version 2 mit benutzerbasierender Sicherheit und zusätzlichen Funktionen
SNMPv2c	SNMP Version 2 mit benutzerbasierender Sicherheit
SNMPv2u	SNMP Version 2 ohne Sicherheit
TCP	Transmission Control Protocol
TLV	Type, Length, Value
UDP	User Datagram Protocol
USM	User Security Model
VACM	View Access Control Model
vgl.	Vergleiche
z.B.	zum Beispiel

Abbildungsverzeichnis

Abbildung 1: Netzwerkmanagementarchitektur 6
Abbildung 2: Kommunikationsvarianten im Netzwerkmanagement 8
Abbildung 3: ASN.1-Objektidentifizierungsbaum 11
Abbildung 4: SNMPv3 Packetformat 31
Abbildung 5: PDU allgemein 34
Abbildung 6: Multiplexen 34
Abbildung 7: Multiplexen einer SNMP-Nachricht 35
Abbildung 8: Die SNMP-Nachricht 36
Abbildung 9: Get-Next Funktion im Etheral-Analyser 38
Abbildung 10: Etheral-Analyser 39
Abbildung 11: SNMP-Nachricht als Hexadezimalcode 40

1 Einführung

„SNMP, das Simple Network Management Protocol, hat das Ziel, Netzwerkmanagern einen zentralen Punkt zur Beobachtung, Kontrolle und Verwaltung ihrer Installationen zu geben. Es ist dabei zunächst völlig unabhängig von herstellergebundenen Konzepten.“[1] In Kapitel 3 bis 7 wird die Rolle von SNMP im Netzwerkmanagement, die Entstehung und der Sicherheitsaspekt von SNMP sowie die Funktionalitäten von SNMP und ihre konkrete Anwendung dargestellt. In Kapitel 8 werden die von SNMP erzeugten PDU’s anaysiert. Um SNMP innerhalb von Netzwerkmanagement einordnen zu können, wird die Thematik Netzwerk-Management in Kapitel 2 zunächst losgelöst von SNMP betrachtet.

[1] vgl. [Ka92] S.260

2 Netzwerkmanagement

2.1 Definition

Für den Begriff „Netzwerkmanagement“ existieren viele verschiedene Definitionen. Netzwerkmanagement kann z.B. folgendermaßen definiert werden:

Netzwerkmanagement ist eine Vereinigung von Prozessen und Verfahren, die dazu genutzt werden, die Qualität und die Effizienz eines Netzwerkes zu pflegen, rücksichtslos auf die Größe und den Typ des Netzwerkes.[2]

„Netzwerkmanagement beinhaltet die Installation, Integration und Koordination von Hardware, Software und menschlichen Elemente zum Überwachen, Testen, Abfragen, Konfigurieren, Analysieren, Bewerten und Kontrollieren des Netzwerks und seiner Element-Ressourcen, um die Anforderungen in Bezug auf Performance im Betrieb und Dienstqualität zu angemessenen Kosten zu erfüllen.“[3]

2.2 Netzwerkmanagement vs. Systemmanagement

Ein anderer Begriff, der viele Berührungspunkte zum Netzwerkmanagement aufweist, jedoch nicht als Synonym dafür verwendet werden kann, ist Systemmanagement. So werden Geräte im Systemmanagement als unabhängige Einheiten, im Netzwerkmanagement hingegen als Mitglied des Netzwerks (erweiterte Sichtweise) betrachtet. Dennoch sind die beiden Begriffe nicht eindeutig voneinander abgrenzbar. Die Überschneidung der beiden Begriffe wird auch an folgenden beiden Beispielen deutlich:

- Das Updaten von Firmware eines Routers (Systemmanagement) beeinflusst die Netzwerkperformance positiv (Netzwerkmanagement).

[2] vgl. [Kl03]

[3] vgl. [Ku02], S.615

- Die Verkleinerung des von einem Router zu verwaltenden Umfangs innerhalb eines Netzes (Netzwerkmanagement) beeinflusst die Leistung der einzelnen Netwerkmitglieder (Systemmanagement). [4]

2.3 Bereiche

Das Netzwerkmanagementmodell der ISO (International Organization for Standards) definiert fünf Bereiche für Netzwerkmanagement, die auch in den meisten anderen Netzwerkmanagement-Philosophien zu finden sind. Diese Bereiche sind sowohl in komplexen Netzwerken als auch innerhalb nicht vernetzter Workstations vorzufinden:[5]

Konfigurationsmanagement

Mit Konfigurationsmanagement kann verfolgt werden, welche Geräte sich im verwalteten Netzwerk befinden. Des Weiteren kann die Hard- und Software dieser Geräte konfiguriert werden.[6] Es existieren drei verschiedene Aspekte des Konfigurations-Managements:

- Inventory: alle Geräte und Software, die in einem Netzwerk installiert sind und ihre statischen Informationen
- Configuration: Plan, wie die Inventory-Komponenten miteinander verbunden sind
- Provisioning bzw. provisioned Information: austauschbare Operations-Parameter, welche die Funktionsweise jeder Komponente bestimmen [7]

Fehlermanagement

Fehlermanagement, das als wichtigster Bereich des Netzwerkmanagement eingeordnet wird, ermöglicht das Finden, Identifizieren, Isolieren, Berichten und Korrigieren von Fehlern innerhalb eines Netzwerkes. Fehler sind unerwartete negative Effekte, wie z.B. ein

[4] vgl. [Kl03]

[5] vgl. [KL03]; [Ku02] S.613

[6] vgl. [Ku02] S. 614

[7] vgl. [KL03]

verloren gegangenes Signal oder die Verminderung einer Geräteleistung.[8] Fehlermanagement umfasst somit die Bereiche Fehlererkennung, Fehlerdiagnose und die Fehlerkorrektur:[9]

- Die Fehlererkennung kann reaktiv (Fehler werden erst entdeckt, wenn sie bereits entstanden sind) oder proaktiv (Früherkennung möglicher Fehlerquellen) erfolgen. Zur reaktiven Feststellung von Fehlern können auf den Geräten z.B. regelmäßig Prozesse zum Testen der Funktionalitäten ausgeführt werden. Zur proaktiven Fehlererkennung werden für die gemanagten Geräte Schwellenwerte (mögliche Fehlerquellen) festgelegt, bei deren Überschreitung ein Alarm vom Gerät an den Netzwerkmanager gesendet wird und somit vor potenziellen Fehlern warnt. So kann für die Festplatte eines Netzwerkgerätes z.B. der Schwellenwert „zu 90% gefüllt“ als mögliche Quelle für eine Überfüllung der Festplatte festgelegt werden. Ist der Schwellenwert erreicht, d.h. die Festplatte zu 90% gefüllt, wird der Netzwerkmanager über einen Fehlerbericht (Alarm) informiert. Die proaktive Fehlererkennung erfordert zusätzliches Überwachungszubehör, detaillierter Informationen über Netzwerkgeschehnisse und verursacht eine höhere Netzwerkbelastung. Die dadurch entstehenden Mehrkosten machen proaktive Fehlererkennung nur in besonders großen Netzwerken rentabel.[10]
- Die Fehlerdiagnose umfasst die Analyse der erkannten Fehler.
- Die Fehlerkorrektur beinhaltet verschiedene Maßnahmen zur Beseitigung der Fehler bis hin zur Auswechslung von Hard- und Software. Dieser Prozess wird meistens vom Konfigurationsmanagement unterstützt.[11]

Performance Management

Performance Management ist die langfristige Überwachung der Netzwerkleistung, ihre Protokollierung und die daraus resultierende notwendige Netzwerkoptimierung. Anhand des Protokolls und der darin enthaltenen statistischen Daten kann auf den möglichen Grund einer jetzigen oder vorhersehbaren Performanceminderung (z.B. bei Überschrei-

[8] vgl. [KL03]

[9] vgl. [Ka92] S.241

[10] vgl. [KL03]

[11] vgl. [Ka92] S.241

tung eines Schwellenwertes) geschlossen und notwendige Maßnahmen eingeleitet werden. Ist in einem Netzwerk keine Überlastung vorhanden, wird es als „performing well“ bezeichnet. Die gebräuchlichste Messung der Netzwerkperformance geschieht auf Paket-Ebene und beinhaltet folgendes:

- Messung der Anzahl der beschädigten Datenpakete
- Messung der Anzahl der Response Time-Outs oder der Pakete, die zurückgeschickt wurden
- Messung der Anzahl der Pakete, die nicht angekommen sind [12]

Die Grenze zwischen Performance- und Fehlermanagement ist nicht eindeutig definierbar. Während Fehlermanagement eher die unmittelbare Behandlung von vorübergehenden Netzwerkfehlern betrifft, beinhaltet Performancemanagement die langfristige Bereitstellung möglichst hoher Leistungen unter Berücksichtigung schwankender Verkehrsnachfragen und Ausfällen von Netzwerkgeräten.[13]

Sicherheitsmanagement

Sicherheitsmanagement dient der Überwachung des Zugangs zum Datennetz und des Zugriffs auf Ressourcen bzw. Services. Es wird auf zwei Ebenen ausgeführt:

- physical security: physikalische Isolation von Schlüsselkomponenten in einem Netzwerk
- logical security: Verwaltung von Systempasswörtern, Ver- und Entschlüsseln von Daten [14]

Accounting Management

Accounting Management ermöglicht die Spezifikation, Protokollierung und Kontrolle des Zugriffs auf Netzwerkressourcen durch Benutzer und Geräte. Dies kann z.B. zur Zuordnung von Zugriffsrechten auf Ressourcen, zur Berechnung von Nutzungsquoten oder zur auf Nutzung basierende Gebührenerhebung dienen.[15]

[12] vgl. [KL03]
[13] vgl. [Ku02] S.614
[14] vgl. [KL03]
[15] vgl. [Ku02] S.614

2.4 Architektur[16]

2.4.1 Komponenten

Die folgende Abbildung zeigt eine allgemeine Darstellung einer Netzwerkmanagement-architektur mit ihren drei Hauptkomponenten verwaltende Einheit, verwaltete Geräte und Netzwerkmanagementprotokoll (z.B. SNMP).

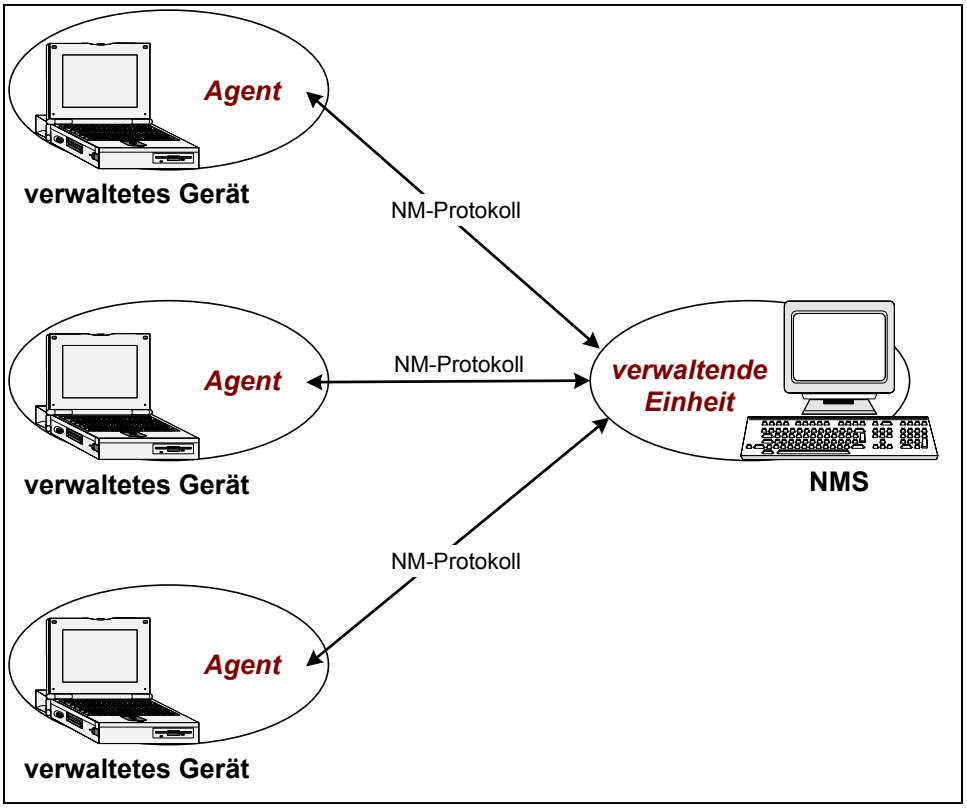

Abbildung 1: Netzwerkmanagementarchitektur

Verwaltende Einheit

Die verwaltende Einheit ist eine Anwendung, welche die Einholung, Verarbeitung, Analyse und Anzeige von Netzwerkmanagementinformationen kontrolliert und somit im Zentrum der Netzwerkmanagementaktivitäten steht. Die verwaltende Einheit läuft auf einer

[16] vgl. [Ku02] S.617

zentralen Netzwerkmanagementstation (NMS) im Network Operations Center (NOC) und wird von einem Menschen, dem Netzwerkmanager, bedient.

Verwaltetes Gerät

Verwaltete Geräte sind Teil des Netzwerkes einschließlich ihrer Software, wie z.B. Hosts, Router, Bridges, Hubs, Drucker oder Modems:

- Innerhalb eines verwaltenden Gerätes kann es mehrere **verwaltete Objekte** (auch managed objects – MO) geben, die den Hardwareteilen (z.B. Netzwerkkarte) dieser Geräte entsprechen.
- Die Netzwerkmanagementinformationen zu den verwaltenden Objekten werden in einer **Managementinformationsbasis** (MIB) erfasst und stehen der verwaltenden Einheit zur Verfügung bzw. werden von dieser vorgegeben.
- Ein **Netzwerkmanagement-Agent** ist ein Prozess, der in jedem zu verwaltenden Gerät läuft und mit der verwaltenden Einheit kommuniziert bzw. Aktionen auf dem verwalteten Gerät unter dem Kommando der verwaltenden Einheit durchführt.

Netzwerkmanagement-Protokoll

Das Netzwerkmanagement-Protokoll dient zur Übertragung der Managementinformationen und ist somit die Kommunikationsschnittstelle zwischen der verwaltenden Einheit und den Agenten der verwaltenden Geräte. Es ermöglicht

- der verwaltenden Einheit, die Status von den verwalteten Geräten abzurufen und auf diesen Geräten Aktionen über ihre Agents auszuführen.
- den Agents der verwalteten Geräte, die verwaltende Einheit über außergewöhnliche Ereignisse (z.B. bei Überschreitung eines Schwellenwertes) zu informieren.

2.4.2 Kommunikation

Aus der Beschreibung des Netzwerkmanagementprotokolls wird ersichtlich, dass zwei generelle Kommunikationsvarianten zwischen der verwaltenden Einheit und den Agenten der verwalteten Geräte stattfinden können:

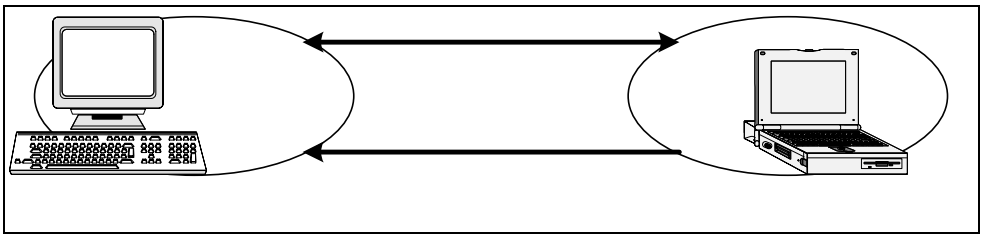

Abbildung 2: Kommunikationsvarianten im Netzwerkmanagement

1. Anfrage/Antwort[17]

Die verwaltende Einheit sendet eine Anfrage an den Agenten des verwalteten Gerätes, der Agent empfängt die Anfrage, führt daraufhin eine entsprechende Aktion aus und sendet eine Antwort an die verwaltende Einheit. Die angefragte Aktion kann das Auslesen von Geräteinformationen oder das Modifizieren von Geräteparametern sein. Die verwaltende Einheit kann somit an einem von ihr bestimmten Zeitpunkt Informationen von den verwalteten Geräten einholen bzw. die Geräte konfigurieren.

2. Alarm

Der Agent des verwalteten Gerätes sendet unaufgefordert, bei Eintreten eines bestimmten Ereignisses (z.B. Überschreitung eines Schwellenwertes) eine Alarmnachricht[18] an die verwaltende Einheit. Die verwaltende Einheit kann somit über Ausnahmesituationen informiert werden.

Aus Perspektive des Netzwerkmanagements ist die Nutzung von Alarmnachrichten in zweierlei Hinsicht vorteilhafter:

- Die Benachrichtigung bei kritischen Ereignissen ermöglicht der verwaltenden Einheit die Vorbeugung von Netzwerkstörungen (proaktives Netzwerkmanagement).

verwaltend
Einheit

[17] Üblich ist auch die analoge Verwendung der Begriffe Request/Response.

[18] Vor allem im Umgang mit SNMP wird die Alarmnachricht auch als Trap-Nachricht bezeichnet.

NMS

- Der Einsatz von Alarmnachrichten wirkt sich günstig auf die Netzwerkauslastung aus, da diese nur in eine Richtung (vom Agenten zur verwaltenden Einheit), nur bei Eintreten des definierten Ereignisses und nur vom Agenten des betroffenen Gerätes gesendet werden. Bei der Anfrage/Antwort-Variante werden Nachrichten in beide Richtungen (Anfrage von der verwaltenden Einheit zum Agenten, Antwort vom Agenten zur verwaltenden Einheit), in regelmäßigen Abständen und an bzw. von alle(n) damit zu überwachenden Geräte gesendet (höhere Netzauslastung).

3 SNMP - Grundlagen

3.1 Internet Management Framework[19]

SNMP als Netzwerkmanagementprotokoll dient der Übertragung von Managementinformationen zwischen der verwaltenden Einheit und den Agenten. Das Protokoll SNMP ist aber nur ein Teil vom ebenfalls als SNMP bezeichneten Internet Management Framework – ein Rahmenwerk für Netzwerkmanagement, deren Entstehung und heutiger Stand im Kapitel 4 beschrieben wird. Die Ziele, die SNMP als Rahmenwerk für Netzwerkmanagement verfolgt, entsprechen denen des Netzwerkmanagements - SNMP dient somit „vor allem zwei Einsatzzwecken: Der Konfiguration und Überwachung von netzwerkfähigen Geräten.“[20] Dieses Rahmenwerk für Netzwerkmanagement besteht aus vier Komponenten, die im folgenden beschrieben werden.

3.2 Managementinformationsbasis (MIB)[21]

Die Managementinformationsbasis beinhaltet die Managementinformationen der verwalteten Geräte in Form von MIB-Objekten und den ihnen zugeordneten Werten. MIB-Objekte, wie z.B. die Anzahl von verworfenen Datagrammen, die Version der Gerätesoftware, Statusinformationen bezüglich der Gerätefunktionalität oder die Gerätebezeichnung, existieren in standardisierter Form oder können vom Netzwerkmanager selbst definiert werden. Die Werte der MIB-Objekte können von der verwaltenden Einheit abgerufen bzw. gesetzt werden.
Zur Identifizierung bzw. Benennung der MIB-Objekte wurde ein von der ISO entwickeltes Objektidentifizierungsrahmenwerk übernommen, das Teil der Objektdefinitionssprache ASN.1 (siehe 3.4) ist. In diesem Rahmenwerk sind Objekte auf hierarchische Weise innerhalb einer Baumstruktur benannt – jeder Zweig im Baum hat sowohl einen Namen als auch eine Nummer. Jede Stelle in diesem Identifizierungsbaum ist somit durch eine Sequenz von Namen bzw. Nummern identifiziert, welche den Pfad von der Wurzel bis zur

[19] vgl. [Ku02] S.619
[20] [EC02] S. 46
[21] vgl. [Ku02] S.623

entsprechenden Stelle im Baum beschreibt. MIB-Objekte haben in diesem umfassenden Identifizierungsbaum ein kleine Nische, die in der folgenden Abbildung dargestellt ist.

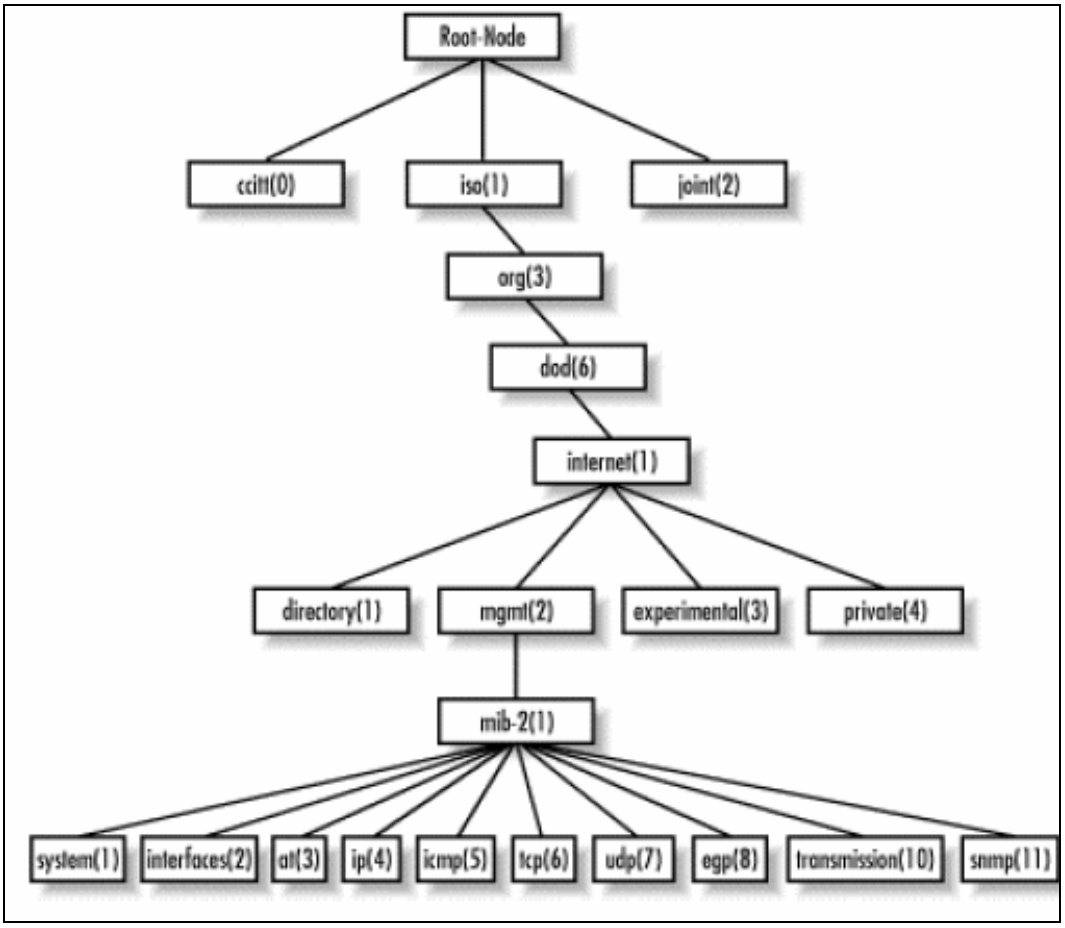

Abbildung 3: ASN.1-Objektidentifizierungsbaum[22]

Die MIB-Objekte setzen sich neben der durch die Baumstruktur eindeutig zugewiesenen Objektidentizifierungsnummer[23] aus dem Objektname, dem Datentyp und einer Objektbeschreibung zusammen. Zusammenhängende MIB-Objekte können in so genannten Modulen gruppiert werden. In der unteren Tabelle werden die MIB-Objekte des Stan-

[22] vgl. [Ma01] S.33

[23] auch bezeichnet als Objekt-ID

dardmoduls ‚system' mit den entsprechenden Beschreibungen (nach RFC 1213[24]) aufgeführt.

Obektidentifizierung	Name	Datentyp	Beschreibung
1.3.6.1.2.1.1.1	sysDescr	OCTET STRING	voller Name und Versionsidentifikation ▪ des Hardwaretyps ▪ des Betriebssystem ▪ der Vernetzungssoftware des Systems
1.3.6.1.2.1.1.2	sysObjectID	OBJECT IDENTIFIER	dem Hersteller zugeteilte Objekt-ID, die ein einfaches und eindeutiges Mittel für die Feststellung liefert, welche Art von Box verwaltet wird
1.3.6.1.2.1.1.3	sysUpTime	TimeTicks	Zeit (in Hundertstelsekunden) seit der NM-Teil des Systems zuletzt initialisiert wurde
1.3.6.1.2.1.1.4	sysContact	OCTET STRING	Kontaktperson für den verwalteten Knoten mit Kontaktinformationen
1.3.6.1.2.1.1.5	sysName	OCTET STRING	administrativ zugeteilter Name für den verwalteten Knoten (voll qualifizierter Domain-Name)
1.3.6.1.2.1.1.6	sysLocation	OCTET STRING	physische Standort des Knotens
1.3.6.1.2.1.1.7	sysServices	Integer32	kodierter Wert, der die auf diesem Knoten verfügbaren Dienste bezeichnet

Tabelle 1: MIB-Objekte des system-Moduls

[24] RFC ist eine Dokumentenreihe, welche Experimente und Neuerungen in der Internet-Protokollfamilie und ähnlichen Projekten beschreibt.

3.3 Structure of Management Information (SMI)[25]

Die Structure of Management Information ist eine auf ASN.1 (siehe 3.4) basierende Datendefinitionssprache, die zur Definition der Managementinformationen im MIB dient. SMI stellt zum einen die Datentypen für die MIB-Objekte zur Verfügung und ermöglicht es zum anderen eigene MIB-Objekte, u.a. mit Hilfe dieser Datentypen, und MIB-Module zu definieren. Die folgende Tabelle enthält die elf **SMI-Basisdatentypen** (definiert in RFC 2578).

Datentyp	Beschreibung
INTEGER	32-Bit-Ganzzahl gemäß Definition ASN.1 mit einem Wert zwischen -2^31 und 2^31-1 oder einem Wert aus einer Liste möglicher benannter Konstantenwerte.
Integer32	32-Bit-Ganzzahl mit einem Wert zwischen -2^31 und 2^31-1
Unsigned32	Vorzeichenlose 32-Bit-Ganzzahl mit einem Wert zwischen -2^31 und 2^31-1
OCTET STRING	Bytekette im ASN.1-Format, die beliebige Binär- oder Textdaten darstellt; maximal 65535 Byte
OBJECT IDENTIFIER	Strukturierter Name im ASN.1-Format (wird administrativ zugewiesen)
IPaddress	32-Bit-Internet-Adresse in der Netzwerk-Byte-Reihenfolge
Counter32	32-Bit-Zähler, der sich von 0 bis 2^32-1 erhöht und dann wieder von 0 beginnt
Counter64	64-Bit-Zähler
Gauge32	32-Bit-Ganzzahl, die nicht über 2^32-1 zählt und nicht unter 0 sinkt, wenn erhöht oder verringert wird
TimeTicks	Zeit, die seit einem Ereignis verstrichen ist (in Hundertstelsekunden)
Opaque	nicht interpretierte ASN.1-Zeichnekete, die für Abwärtskompatibilität

[25] vgl. [Ku02] S.620

	benötigt wird

Tabelle 2: SMI-Basisdatentypen

Die **Definition eines MIB-Objekts** ermöglicht SMI mit dem Sprachkonstrukt OBJECT-TYPE. Dieses Sprachkonstrukt wird an der beispielhaften Definition des Objekts ‚ipInDelivers' (32-Bit-Zähler, der die Anzahl der IP-Datagramme verfolgt, die vom verwalteten Knoten empfangen und an das höherschichtige Protokoll abgegeben wurden) erklärt:

```
ipInDelivers OBJECT-TYPE
        SYNTAX          Counter32
        MAX-ACCESS      read-only
        STATUS          current
        DESCRIPTION     "Die Gesamtzahl der Eingabe-Datagramme,
                        die erfolgreich an IP-User-Protokolle
                        abgegeben wurden."
        : := { ip 9 }
```

Das Konstrukt OBJECT-TYPE enthält, wie im Beispiel erkennbar, vier Klauseln:

- die Klausel SYNTAX definiert den Basisdatentyp des Objekts
- die Klausel MAX-ACCESS bestimmt, ob das Objekt gelesen, geschrieben oder erzeugt werden kann
- die Klausel STATUS bestimmt, ob die Objekdefinition aktuell und gültig ist oder z.B. nur für historische Zwecke benötigt wird
- die Klausel DESCRIPTION enthält ein Objektbeschreibung in Textform, die den Zweck des Objekts dokumentiert

3.4 Protokoll SNMP[26]

Entsprechend der Funktionalität von Netzwerkmanagement-Protokollen dient SNMP zur Übertragung der MIB-Informationen zwischen der verwaltenden Einheit und den Agenten der verwaltenden Geräte. SNMP wird somit auf zwei Arten genutzt:

- Am häufigsten wird der **Request/Response-Mode** (analog zu Antwort/Anfrage) genutzt, in dem die verwaltende Einheit eine Anfrage an den Agenten eines verwalteten Gerätes sendet, der eine bestimmte Aktion ausführt und eine Antwort auf die Anfrage sendet.
- Weiterhin wird SNMP dazu genutzt, die verwaltende Einheit durch das Versenden von unaufgeforderten **Trap-Nachrichten** (analog zu Alarm) seitens über außergewöhnliche Ereignisse zu informieren.

3.5 Sicherheits- und Administrationsfähigkeiten

Die vierte Komponente des Internet Management Framework umfasst die Sicherheits- und Administrationsfähigkeiten von SNMPv3 als wichtigste Erweiterung gegenüber SNMPv2 (siehe SNMP-Entstehung und Sicherheitsaspekte von SNMP).

3.6 ASN.1

ASN.1 (Abstract Syntax Notification) ist eine Sprache, die dazu dient das Format von SNMP-Nachrichten und MIB-Modulen zu definieren. Neben ASN.1 benutzt SNMP die Sprachen BER (Basic Encoding Rules) und SMI (Structure of Management Information). Die Sprachen dienen dazu die im Protokoll enthaltenen Daten in eine verständliche Sprache umzusetzen, während das Protokoll die Regeln enthält mit deren Hilfe die Daten zwischen den verschiedenen Geräten transportiert werden.
ASN.1 ist ein ursprünglich von der ISO entwickelter Standard, welcher insbesondere im Bereich Netwerkmanagement benutzt wird. Vor allem die bereits erwähnten MIB Variablen in SNMP sind unentwirrbar mit ASN.1 verbunden. Wenn zum Beispiel ein SNMP-Agent eine Response Nachricht sendet, die die ganzzahlige Anzahl der empfan-

[26] vgl. [Ku02] S.626

genen UDP-Datagramme enthält, wird der ganzzahlige Wert, der an die verwaltete Einheit zu senden ist in eine unabhängige Methode zur Beschreibung von Ganzzahlen und anderen Datentypen umgewandelt. Diese Methode wird durch SMI und ASN.1 unterstützt, in der ISO-Sprache werden die beiden Standards als Presentation Service bezeichnet. Der Presentation Service (Darstellungsdienst) dient zur Übertragung un d Übersetzung von Informationen von einem rechnerspezifischen Format in ein anderes.
Als Datenbeschreibungssprache hat ASN.1 eine eigene Syntax und eine Reihe von stilistischen Konventionen. Folgendes Beispiel stellt eine Daten-Strukturen in Form von geordneten Listen da, diese werden mit dem SEQUENCE Schlüsselwort (Befehl) erstellt:

```
ErrorCounts :: = SEQUENCE      {
circuitId                  OCTECT STRING,
erroredSeconds             INTEGER,
unavailableSeconds         INTEGER
}
```

Nun kann eine Liste, die auf dem "ErrorCounts" Datentyp basiert, erzeugt und Ihre Einträge folgendermaßen initialisiert werden:

```
circuitPerformance :: = ErrorCounts  {
"",     "Unassigned",
erroredSeconds      0,
unavailableSeconds          0
}
```

Um eine Nachricht im Netzwerk transportieren zu können, wird ASN.1 in BER (Basic Encoding Rules) umgewandelt. Die BER sind Regeln, die spezifizieren, wie Instanzen von Objekten, die mit Hilfe der Datenbeschreibungssprache ASN.1 definiert wurden, über das Netzwerk versendet werden sollen. BER basiert auf dem TLV-Ansatz (Type, Length, Value) für die Codierung von Daten als Vorbereitung auf die Übertragung. Die Übertragung findet dann in einem Bytestrom statt, welcher die nach den BER und TLV-Ansatz angeordneten binären Werte enthält.

4 SNMP - Entstehung

Kam es zu Zeiten des APANET[27] vor, dass die Verzögerung zu einem Host unverhältnismäßig lang wurde, konnte der für das Netz zuständig Administrator über den PING-Befehl ein Paket an das Ziel schicken und durch Analyse des Zeitstempels im Header[28] des zurückkommenden Paketes den fehlerhaften Punkt im Netz ausfindig machen und eine entsprechende Handlung zur Beseitigung dieses Fehlers initiieren. Dies war möglich, da das APRANET nicht sehr groß war und somit die Zahl der Router sehr gering. So war es möglich mit einfachsten Mitteln (wie zum Beispiel dem PING-Befehl) den Gesamtzustand des Netzes zu diagnostizieren.[29]

Als allerdings 1983 vom Amerikanischen Verteidigungsministeriums das TCP[30]/ IP-Protokoll zum Standard Internet Protokoll erklärt wurde war dies zum einen das Ende des APRANET und zum anderen die Geburtsstunde des Internets.

Mitte der 80 `er Jahre wuchs das Internet sprunghaft an und es bildeten sich mehrere Betreiber. Somit wuchs auch die Zahl der einzelnen Netze, die wiederum immer größer wurden und es entstanden die ersten Backbones[31]. Für die Verwaltung dieser immer weiter wachsenden Netze reicht der PING-Befehl bald nicht mehr aus und es mussten neue Verwaltungswerkzeuge gefunden werden. Außerdem wurde auch erkannt, dass man sich möglichst bald auf ein Standard Verwaltungstool einigen muß, da die einzelnen Benutzergruppen der Netzwerke unterschiedliche Verwaltungstools und/ oder Ver-

[27] Erstes größeres Netzwerk. Von amerikanischen Universitäten für das amerikanische Militär entwickelt.

[28] Kopf eines Paketes, enthält Informationen über den Status des Paketes, seinen Weg, Prüfsummen, usw. Inhalt ist aber von Protokoll zu Protokoll unterschiedlich

[29] vgl. [TA97] S. 630

[30] „Das Transmission Control Protocol (kurz: TCP) bietet sichere Vollduplex-Verbindungen zwischen zwei Hosts. Bevor TCP Daten übertragen kann, muss es eine Verbindung zwischen zwei Hosts herstellen. Dieser Vorgang ist vergleichbar, mit dem Aufbau einer Verbindung zwischen zwei Telefonen. [...] Daten lassen sich über diese virtuellen Verbindungen simultan in beide Richtungen übertragen. Nach Abschluss der Datenverbindung kann die TCP-Verbindung wieder geschlossen werden. [...] TCP überprüft, ob ein Datenpaket tatsächlich beim Empfänger eingetroffen ist. Geht unterwegs ein Paket verloren, so wird es erneut gesendet.“ [KL03]

[31] Höchste hierarchische Verbindungsebenen bei Netzwerken (z.B. Interkontinentale Verbindungen)

fahren zur Verwaltung einführten und nutzten. Diese konnte aber meistens untereinander nicht kommunizieren, waren also nicht kompatibel und konnten nicht gemeinsam genutzt werden. Gegen Ende der 80 `er Jahre fingen mehrere unabhängige Gruppen von Entwicklern mit der Erforschung von Netzwerkmanagementprotokolls an, die das Potential haben sollten sich als Standard durchzusetzen.

Das erste Tool, welches in Versuchs-Netzwerken volle Funktionalität unter Beweis stellte, war das High-Level Entity Management System (kurz: HEMS). Dieses Tool blieb aber nur ein Experiment und wurde im Internet nie angewandt.

Das zweite Modell wurde 1987 entwickelt. Dieses wurde von der Open Systems Interconnection Gruppe (kurz: OSI)[32], welche ein Teil der International Standards Organisation (kurz: ISO)[33] ist, vorgestellt. Das von der OSI vorgestellte Modell sollte ein Grundgerüst zur Internetverwaltung darstellen und wurde Common Management Information Protokoll (kurz: CMIP) genannt. Ein Nachteil des CMIP war allerdings, dass es nur für das Management von OSI-basierten Netzwerken eingesetzt werden konnte und daher nicht den Anforderungen an ein weltweit nutzbares Netzwerkmanagementtool genügt. Daher wurde das CMIP kurze Zeit später durch das CMOT (CMIP über TCP) ersetzt. Dieses Protokoll hatte die selben Funktionen wie CMIP konnte aber zum verwalten von Netzwerken eingesetzt werden, welche die Internet Protokoll Familie (TCP) benutzen. Warum sich dieses Tool nicht bewährt hat und vor allen Dingen durchsetzen konnte bleibt bis heute unklar.

Parallel zu den oben erwähnten Bemühungen ein standardisiertes Netzwerkmanagementprotokoll zu finden, begann im März 1987 eine Gruppe von Netzwerkentwicklern mit der Entwicklung des Simple Network Gateway Monitoring Protocol (kurz: SGMP). Dieses Protokoll sollte sehr einfach aufgebaut und mühelos zu integrieren sein, da man erkannt hatte, dass Komplexität viele Benutzer abschreckt. Da SGMP diese beiden sehr positiven Eigenschaften in sich vereinen konnte, wurde es von immer mehr Gruppen in ihren

[32] Eine internationale Anstrengung, die Kommunikation zwischen Rechnern unterschiedlicher Hersteller und Technologien zu erleichtern. Begründer des 7 Schichten OSI-Models.

[33] Die Organisation, die viele Standards weltweit definiert. OSI ist eines von vielen Gebieten, die von der ISO / IEC standardisiert werden.

Netzwerken zur Verwaltung eingesetzt und wurde schließlich im November 1987 im RFC[34] 1028 vorgestellt.

Nachdem es nun drei verbreitete Netzwerkmanagementprotokolle gab (HEMS, CMOT, SGMP) versammelten sich Anfang 1988 die Internet Activities Board (kurz: IAB)[35] um sich für eines dieser Protokolle zu entscheiden und somit festzulegen, welches im Internet einheitlich genutzt werden soll. Es wurde sich darauf geeinigt, dass HEMS auf Grund seiner mangelnden Akzeptanz und der eigentlich nicht vorhandenen Verbreitung fallen gelassen muß. SGMP schien wegen seiner steigenden Akzeptanz, der immer größer werdenden Verbreitung in der Internet-Gemeinde und des einfachen und leicht zu integrierenden Konzeptes die beste Wahl. Allerdings erkannte man, dass CMOT sämtliche Funktionen von SGMP ersetzen kann und daher entschied man sich für CMOT als zukünftiges einheitliches Internetprotokoll, obwohl es zu diesem Zeitpunkt noch nicht für den Vertrieb bereit war.

Aufgrund dieser Entscheidung musste ein Internet Management Framework[36] entwickelt werden, welches den Übergang von SGMP zu CMOT gewährleistet, also von beiden Modellen benutzt werden kann. Dieses Grundgerüst bekam den Namen Simple Network Management Protocol und wurde in den RFCs 1065 bis 1067 beschrieben. Der Vorsitzende der SNMP-Gruppe war Marshall T. Rose, welcher genau wie Mc-Cloghrie eine kompakte Beschreibung (Rose 1994/ Mc-Cloghrie 1995) von SNMP anfertigten.

Da kurze Zeit später viele Probleme und Unstimmigkeiten bei der Vereinigung von CMOT und SNMP wuchsen und SNMP stabiler und mittlerweile auch weiter verbreitet war als CMOT, entschied das IAB im April 1989, dass SNMP zum Standard für das TCP/ IP Internet wurde.

Zwei Monate später entschied das IAB allerdings das in Zukunft beide Modelle (CMOT und SNMP) frei weiter entwickelt werden sollten. Kurz darauf erarbeitete die SNMP

[34] RFC ist eine Dokumentenreihe, welche Experimente und Neuerungen in der Internet-Protokollfamilie und ähnlichen Projekten beschreibt.

[35] Die IAB ist die technische Organisation, die die Entwicklung der Internet-Protokollfamilie steuert.

[36] Framework bedeutet frei übersetzt Grundgerüst.

Gruppe einen neuen Entwurf der am Ende des Jahres von der IAB angenommen wurde, worauf diese im Mai 1990 SNMP als Standard Protokoll mit empfohlenen Status der Internet-Welt empfahl (siehe RFC 1157). Das von der IAB empfohlene SNMP Protokoll stellte die erste Version von SNMP (SNMPv1) dar.

5 SNMP - Versionen

5.1 SNMPv1

Wie schon im vorherigen Absatz erwähnt wurde 1990 SNMPv1 von IAB zum empfohlenen Standard für die Internetverwaltung erklärt. SNMPv1 zeichnet sich dadurch aus, dass es ein simples, auf die Bedürfnisse einfacher Netze zugeschnittenes Protokoll mit 4 Grundoperationen ist. Diese Grundbefehle sind:

- get request
- get next request
- set request
- trap

Erläutert werden diese Befehle im Abschnitt: 6 SNMP – Nachrichten.

Neben diesem Vorteil des einfach „gestrickten“ Protokolls stehen aber auch einige Nachteile, welche nicht zu vernachlässigen sind. So wird bei SNMPv1 zum Beispiel der gesamte Overhead von OSI mitgeführt und das Protokoll baut einzig und allein auf den verbindungslosen UDP-Protokoll[37] auf, wodurch das Risiko von Nachrichtenverlusten nicht ausgeschlossen werden kann. Außerdem bietet SNMPv1 keinerlei Sicherheitsoptionen da die Zugriffssteuerung einzig und allein über Community-Strings erfolgt. Da viele

[37] „UDP kann Daten übertragen, ohne vorher eine virtuelle Verbindung herstellen zu müssen. Alle Dateieinheiten, die übertragen werden, verfügen über Absender- und Ziel-IP-Adressen sowie Port-Adressen, die die Applikationen identifizieren, die beim Datenaustausch mitwirken. UDP ist wie IP ein verbindungsloses Protokoll. es besitzt weniger Overhead als TCP, garantiert aber im Gegensatz dazu keine korrekte Reihenfolge bei der Paketablieferung. UDP beinhaltet optional Prüfsummen, über die sich die Integrität der gesendeten Nachricht sicherstellen lässt.“ [KL03]

Netzwerkadministratoren nur „[...] die Standardvorgabe ´public´ und ´private´ für [...]“[38] ihre Communities verwenden öffnet dies „[...] unerwünschten Konfigurationseingriffen Tür und Tor. Och auch spezielle Community-Namen bieten keinen ernsthaften Schutz, denn SNMPv1 überträgt sie – wie auch die Daten – im Klartext. So lässt sich mit einem Netzwerksniffer schnell ein Community-String erlauschen, der dann Zugriff auf wichtige Netzwerkkomponenten erlaubt.“[39]

Da die oben erwähnten Befehle äußerst einfach gehalten wurden und immer nur einen Request zulassen ist SNMPv1 nicht in der Lage Massentransfers durchzuführen. SNMPv1 verfügt auch nur über einen stark begrenzten Bereich an MIB-Variablen haben.

All diese Unzulänglichkeiten führten dazu, dass man 1992 mit der Entwicklung einer zweiter SNMP-Version begann.

5.2 SNMPv2

Den ersten Schritt zur Entstehung von SNMPv2 bildete die Version Secure SNMP (kurz: SNMPsec). Ausgangspunkt für die Entwicklung dieser Version, war die Einsicht, dass SNMP nicht ohne Sicherheitsmechanismen auskommen würde. So wurden dem Grundgerüst von SNMPv1 Sicherheits- und Administrationsmechanismen hinzugefügt. Beschrieben werden diese Neuerungen in den RFCs 1351 bis 1353. Diese Neuerungen stellten den offiziellen Beginn der Entwicklung von SNMPv1 zu SNMPv2 dar. Anwendung hat SNMPsec nie richtig gefunden, da die meisten Benutzer es vorzogen komplett auf SNMPv2 zu warten um dann einen direkten Wechsel von SNMPv1 u SNMPv2 vorzunehmen.

Für die eigentliche Entwicklung von SNMPv2 haben sich die Programmierer große Ziele und nur ein geringen Zeitraum gesetzt. Sie wollten sämtliche Schwachstellen aus SNMPv1 beseitigen, indem sie das Konzept (Grundoperationen und Mechanismen) von

[38] [EC02] S. 46

[39] [EC02] S. 46

SNMPv1 beibehielten aber den Kernteil des Protokolls vollständig neu konzipierten. Für diese Aufgabe wurde ein Entwicklungszeitraum von nur einem Jahr angesetzt.[40]

Die erste Neuerung war, dass die Grundbefehle von SNMPv1 um vier weitere erweitert wurden. So gab es nun (neben den Alten get request, get next request, set request, trap) zusätzlich:

- 1 get bulk request
- 2 inform request
- 3 report request
- 4 notification

Durch diese neue Befehle besteht die Möglichkeit SNMP-Nachrichten zu bündeln. Erläutert werden diese Befehle im Abschnitt: 6 SNMP – Nachrichten.

Um einen weiteren Nachteil der ersten Version zu beheben wurde eine neue MIB-Generation eingeführt (MIB II), durch welche eine eindeutige und weltweite Zuordnung zwischen OID's und Herstellern von Netzwerkgeräten mit Hilfe einer Standarddefinition erreicht werden sollte. Die wir erreicht indem wichtige Informationen, wie die IP-Adresse, der Herstellername usw., an einem vorgeschriebenen Ort untergebracht werden. Nun war es möglich unabhängig vom Hersteller und der Geräteart wichtige, netzwerkspezifische Daten von jedem SNMP-fähigen Netzwerkgerät auszulesen. Außerdem wurde eine Erweiterung der Definitionsbereiche der MIB-Variablen gewährleistet.[41]

In der ersten Version von SNMPv2, der SNMPv2p, wurden sehr komplexe Sicherheits- und Administrations-Mechanismen eingebaut. Diese Version hat aber nie ihre Akzeptanz in der Industrie gefunden, was wahrscheinlich daran liegt, dass die eingebauten Mechanismen zu komplex waren und daher wurden sie bald wieder fallengelassen.

[40] vgl. [FR02]

[41] Vgl. [FR02]

Weiterhin sollte erwähnt werden, dass mit dem SNMPv2 Modell eine Performancesteigerung erreicht wurde und die neue Version Ansätze für die Unterstützung mehrerer Protokolle enthält.

Natürlich gab es auch in der neuen Version Schwachstellen, so ist besteht eine völlig „[...] Inkompatibel zu SNMPv1 [...]“[42], dies bedeutet, dass Geräte, welche nur SNMPv1 unterstützen nicht mit Geräten kommunizieren können, welche nur SNMPv2 unterstützen.

Weitere Schwachstellen sind, dass der Speicherplatz für SNMP Objekte gestiegen ist, was damals in erster Linie eine Kostensteigerung wegen hören Speicherbedarfs darstellte, und das SNMPv2 nicht erkennt und somit auch verhindert das Abfragen mehrmals ins Netz eingespielt werden.[43]

Ein weiterer wichtiger Punkt ist, dass nachdem vom Internet Engineering Task Force (kurz: IETF)[44] im April 1993 SNMPv2p offiziell SNMPv1 abgelöst hat und zum empfohlenen Standard wurde (RFCs 1441, 45-47) und es in der SNMP-Gemeinde Einigkeitsbestrebungen gab trotzdem ab 1996 drei verschiedene Implementationen von SNMPv2 parallel entwickelt wurden und existieren. Dies sind:

- SNMPv2u
- SNMPv2*
- SNMPv2c

SNMPv2u und SNMPv2* sind beides Versionen die Benutzerbasierende Sicherheitsmerkmale haben, wobei SNMPv2* dies noch durch einige Zusatzfunktionen komplettiert. SNMPv2c bietet gar keine Sicherheitsmerkmale wird aber erstaunlicher Weise am meisten von der Industrie genutzt und ist somit am meisten verbreitet und akzeptiert.[45] Dies

[42] [FR02]

[43] vgl. [FR02]

[44] Ein Komitee des IAB der die Aufgabe hat kurzfristige Bedürfnisse der Internet-Gemeinde zu befriedigen.

[45] vgl. [KL03]

führt aber dazu, dass im Sicherheitsbereich wieder die gleichen Probleme auftauchen wie bei SNMPv1.

5.3 SNMPv3

Die dritte SNMP-Version entstand unter der Leitung des IETF-Komitee. Ziel der Entwicklung war es SNMPv2u und SNMPv2* zu vereinen. Das heißt man wollte die Konzepte der jeweiligen SNMPv2-Versionen (insb. SNMPv2u und SNMPv2*) zu einem SNMP-Standard zusammenführen und weiterentwickeln. Außerdem sollten in der dritten Version von SNMP endlich ein Sicherheitsmechanismus hinzugefügt werden, welcher Verschlüsselungen und Authentisierungen ermöglicht aber nicht so komplex ist, dass ihn wieder niemand annehmen will. In diesem Zuge sollte auch der MIB II, aus SNMPv2, um „[...] sicherheitsrelevante Elemente [...]“[46] ergänzt werden.

Neben diesen Hauptzielsetzungen gab es für SNMPv3 noch einige weiteren Ideen oder auch Features. So wollte man mit SNMPv3 ein Tool herausbringen, welches aus verschiedenen Modulen besteht, so dass der Benutzer immer den für sich nötigen Bereich nutzen kann und nicht gezwungen ist das ganze Tool zu benutzen. Die Zielsetzung war, dass SNMPv3 in den verschiedenen Versionen von der minimalen Funktionalität bis hin zum vollem Funktionsumfang zu bekommen ist. Sehr eng verbunden damit ist auch die Idee, dass SNMPv3 mit mehreren (älteren) Versionen von SNMP zu benutzen sein soll, die neuste Version soll parallel alle anderen Versionen unterstützen. Dies wird ermöglicht durch die Einführung von „[...] „mehrsprachigen Manager und Agenten“ [...]“[47].

Da SNMPv3 ein noch sehr neues Tool ist, ist es in realen Netzen, außerhalb von Versuchsnetzwerken, noch nicht sehr verbreitet. Daher werden sich die Schwachstellen dieser Version erst in einiger Zeit zeigen und dann höchst wahrscheinlich als Vorlage zur Entwicklung von SNMPv4 dienen.

[46] [FR02]

[47] [FR02]

6 SNMP - Nachrichten

Mittels SNMP sollen Informationen gesammelt und gemanaged werden. Um diese Aufgabe erfüllen zu können stellt SNMP ein Set von Nachrichten bereit die zum Austausch von Informationen von den Agents und Managern genutzt werden können. In diesen Kapitel wird deren Funktionsweise näher erläutert.
Wie bereits in Kapitel: 5 SNMP - Versionen erwähnt wurde das SNMP Protokoll seit seiner Einführung ständig weiterentwickelt. Um die Kompatibilität der einzelnen Versionen die in den verschiedenen Hard- und Softwarekomponenten verwendet wurden zu gewährleisten konnte man die Struktur der älteren Nachrichten nicht einfach modifizieren um neue Funktionalitäten hinzuzufügen. Mit SNMPv2 wurden deshalb einige neue Nachrichten definiert welche das Informationsmanagement vereinfachen.
Folgende Übersicht zeigt eine zusammengefasste Auflistung der einzelnen Nachrichten und die SNMP-Version ab der sie Verfügbar sind.

- get request SNMPv1
- get next request SNMPv1
- get bulk request SNMPv2
- set request SNMPv1
- get response SNMPv1
- trap SNMPv1
- notification SNMPv2
- inform SNMPv2
- report SNMPv2

get request

Der get request wird von der NMS an den Agent gesendet. Der Agent empfängt den Befehl und bearbeitet diesen. Wenn der Agent die angeforderte Information findet sendet er ein get response zu der NMS. Was für Informationen müssen in der Nachricht an den Agenten gesendet werden damit man die angeforderten Informationen erhält?

Einen Community-String, einen Object Identifier (OID).und die IP-Adresse zur Identifizierung des Agenten. Der Community-String ist eine Zeichenkette die in allen Nachrichten enthalten ist und Mitglieder einer Gruppe identifiziert. Somit stellt der Community-String eine Art Passwort dar. Und letztendlich der OID mit dessen Hilfe man dem Agenten mitteilt welche Information von ihm anfordert wird. Wie zum Beispiel: 1.2.6.1.2.1.1, welcher für die system group steht. Die Antwort, get response, des Agents auf den get request beinhaltet die gleichen Informationen: Community-String, angefragte OID und den Wert (Value) der Variable, der in dem hier verwendeten Beispiel Auskunft über den Standort des Agents gibt.

get-next request

Ähnlich wie get request wird get-next request zum Anfordern von Daten verwendet. Dieser Nachrichttyp weist das selbe Format wie der get request auf. Der Unterschied ist das der get-next request nicht die angeforderte Variabel zurückgibt sondern die die im MIB-Baum nach ihr kommt. Deshalb wird get-next hauptsächlich für das auslesen von Tabellen genutzt.

set request

Der set request wird von der NMS genutzt um bei einem Agenten einen oder mehrere Werte zu setzten oder zu verändern. Der Agent antwortet auf diesen request mit einem get response, die den Fehlerstatus: NoErrors enthält wenn der Wert gesetzt wurde[48]. Die Praxis hat jedoch gezeigt das man nach jedem set request den Wert des OID nochmals mit einem get request überprüfen sollte um sicherzugehen das er auch wirklich gesetzt wurde. Somit stellt der set request die einzige Möglichkeit von SNMP dar Agenten zu konfigurieren.

get bulk request

Get bulk request fordert die Werte einer Folge von Objekten an. Dadurch kann mit einer Anfrage z.B. eine ganze Tabelle gelesen werden. Die Anzahl der zurückgegebenen Objekte wird nur durch die maximale Framelänge des Transportmediums begrenzt.

[48] vgl. [KU02] S.628

trap

Traps sind Ereignismeldungen vom Agenten zu NMSs deren Empfang nicht bestätigt wird. In SNMP sind sechs Ereignisse standardisiert. In der Praxis reichen diese jedoch selten aus. Deshalb können zusätzliche, gerätetypische, nicht standardisierte, trap-Meldungen erzeugt werden. Für nicht standardisierte Ereignisse hat der Hersteller oder Manger die Möglichkeit Enterprise- oder Specific-traps zu definieren.

notification

„In SNMPv2 wurde notification definiert um das PDU Format von SNMPv1 traps zu standardisieren, die ein anderes Format haben als get- und set-Befehle.“ [49] Somit ist notification das SNMPv2-Format für Traps, welches noch zusätzliche Statusmeldungen definiert.

inform

„Inform ist ein Befehl der Manager zu Manager Kommunikation ermöglicht. Wenn ein inform von einer NMS zu einem anderen gesendet wird schickt der Empfänger einen response, der den Erhalt bestätigt. Außerdem kann inform genutzt werden um SNMPv2 traps zu einem NMS zu senden.“ [50]

report

„Hierbei handelt es sich um eine Nachricht die es SNMPv3-Engines ermöglicht miteinander zu kommunizieren, hauptsächlich um Probleme bei der Verarbeitung von SNMP-Nachrichten zu melden.“ [51].

[49] [MA01] S.46

[50] [MA01] S.46

[51] [MA01] S.46

7 SNMP - Sicherheitsaspekte

7.1 Gefahren für Netzwerksicherheit

Es gibt verschiedene Gefahren für die Netzwerksicherheit, im Folgenden wird zuerst eine Übersicht über die einzelnen Gefahren und deren Auswirkungen gegeben. Anschließend wird erläutert welche Sicherheitsmechanismen in den verschiedenen SNMP-Versionen implementiert wurden um die Gefahren für die Netzwecksicherheit zu eliminieren oder reduzieren.

- Masquerating bedeutet das es einem Angreifer gelingt in die Rolle eines anderen zu schlüpfen und in seinem Namen zu handeln. Wenn es einem Angreifer gelingt sich zum Beispiel als ein Netzwerkmanager auszugeben hat er all die Rechte die auch der Netzwerkmanager hat.[52]
- Modification of Information heißt das es jemanden unautorisierten gelingt eine Nachricht abzufangen und diese zu verändern, um sie dann an den eigentlichen Empfänger wieder weiterzuleiten. Dieser glaubt nun das die Nachricht von dem ursprünglichen Sender stammt.[53]
- Massage Stream Modification ist die Gefahr das der Nachrichtenstrom verändert wird. Das beinhaltet die Neuordnung, Verzögerung und wiederholtes einspielen von aufgezeichneten Nachrichten.[54]
- Disclosure ist die Gefahr das vertrauliche Informationen an Angreifer gelangen. Beispielsweise kann ein Angreifer die Managementdatenverkehr ausspionieren und die so gewonnen Informationen für andere Angriffe wie Masquerating nutzen[55].
- Denial of Service: bedeutend das Netzwerkdienste auf irgend eine Weise blockiert werden. Ein Angreifer könnte beispielsweise versuchen ständig TCP Verbindungen zu einem Agenten aufzubauen und ihn damit so auszulasten / überlasten das er andere Anfragen nicht mehr bearbeiten kann[56].

[52] Vgl. [US99] S.4
[53] Vgl. [SE03] S.2
[54] Vgl. [US99] S.4
[55] Vgl. [SE03] S.2
[56] Vgl. [SE03] S.2

- Trafic patern analysis bei dieser Art des Angriffs wird die eigentliche Inhalt der Nachricht ignoriert. Es werden die Sicherheitsrelevanten Informationen über das System extrahiert mittels Analyse des normalen Informationsflusses.[57]

7.2 Sicherheit von SNMPv1

Bei der Entwicklung von SNMPv1 wurde der Aspekt der Sicherheit nicht viel Beachtung geschenkt. Die einzige Form der Zugriffsberechtigung erfolgt über den Community-String. Der Community-String ist eine Zeichenkette die in den einzelnen PDUs enthalten ist und Mitglieder einer Gruppe identifiziert. Somit stellt der Community-String eine Art Passwort dar über das sich der Manager identifiziert; nur wenn der Community-String in der PDU und der Community-String der in dem Agenten hinterlegt ist übereinstimmen erlaubt dieser den Zugriff. Die meisten Herstellern von SNMP managebaren Agenten implementieren zwei Communitys: read-only und read-write um eine weitere Zugriffsrechtsdifferenzierung vornehmen zu können. Eine große Sicherheitslücke in vielen Netzwerken ist das die von den Herstellern Standart Community-Strings public und „private“ von vielen Netzwerkadministratoren nicht geändert werden. Doch auch wenn die Communitys geändert werden bieten diese keinen ausreichenden Schutz vor unberechtigten Zugriffen da SNMPv1 alle UDPs in Klartext überträgt. Wenn sich ein Angreifer in dem Netzwerk befindet kann er mittels eines Sniffers eine SNMP UDP abfangen und den Community-String auslesen, um diesen dann in seinen UDPs zu verwenden. Einige neuere Agents haben als zusätzliches Sicherheitsfeature Access Control Lists (ACL). In den ACLs lassen sich IP-Adressen hinterlegen die zur Authentifizierung eines Managers dienen. Jedoch lassen sich auch IP-Adressen fälschen.

7.3 Sicherheit von SNMPv2

Aufgrund der bereits beschriebenen Mängel von SNMPv1 und weiteren Anforderungen die nicht die Sicherheit betreffen war eine Weiterentwicklung unabdingbar. Wie bereits in Kapitel 5.2 SNMPv2 aufgeführt gab es bei der SNMP Version 2 einige Streitpunkte unter den Entwicklergruppen wie die Sicherheit verbessert werden sollte. Diese verschiedenen Ansätze zur Gewährleistung der Daten- und Netzwerksicherheit führten zu den unter-

[57] vgl. [SE03] S.2

schiedlichen SNMP2-Versionen und diese wiederum dazu das die meisten Hersteller SNMPv2 nicht in ihren Agenten Implementierten und weiterhin die unsichere Version 1 verwendeten oder die Kompromisslösung SNMPv2c nutzten die keine neuen Sicherheitsfeatures enthielt.

7.4 Sicherheit von SNMPv3

Version drei ist nicht als Nachfolger von eins und zwei gedacht sondern als Ergänzung zu diesen und ist Gegensatz zu SNMPv2 ein sauber definierter Standart, welcher der Zugangskontrolle, Authetifizierung und Datensicherheit Rechnung trägt. Dies wird durch zwei Sicherheitsmodelle umgesetzt, dem User Security Model (USM) und dem View Access Control Model (VACM).

User Security Model

USM ist das Sicherheitsmodell welches in SNMPv3 Datensicherheit und Authentifikation gewährleistet. Dazu benötigt jede Einheit (Nutzer) verschiedene secret keys für Authentifikation und Verschlüsselung. Die Authentifikationsprotokolle die Verwendung finden sind HMAC-MD5 und HMAC-SHA und zur Verschlüsselung wird das CBC-DES Protokoll genutzt. Wie in Abbildung 1 dargestellt findet eine Authentifikation für die gesamte Nachricht statt, Verschlüsselung beschränkt sich jedoch auf die eigentliche SNMP PDU. Das heißt wenn ein Agent mit SNMPv3 verwaltet wird entspricht die eigentliche PDU dem SNMPv1 oder v2 Standart und ist in einem SNMPv3-Packet eingekapselt. Vorraussetzung ist das jeder Manager und Agent eine eigene SNMPv3-Engine enthält, welche die Nachricht verarbeitet. Wenn ein MNS eine Nachricht an einen Agenten schicken will passiert folgendes: Die SNMPv3-Engine empfängt das Datagramm von der SNMP Anwendungsebene, führt die Sicherheitsfunktionen aus und verkapselt die PDU in eine SNMPv3-Nachricht bevor sie über das Netzwerk geschickt wird. Der Agent empfängt die Nachricht, führt die entsprechende Decodierung und Authentifizierung mit seiner SNMPv3-Engine aus und reicht die PDU an die SNMP Anwendung weiter.[58] Die SNMP Paketstruktur wurde für Version drei geändert um die Verwendung des oben beschrie-

[58] vgl. [SE04] S.4

benen User Security Models (USM) zu ermöglichen. Abb. 1 stellt die Struktur der SNMPv3 PDU dar.

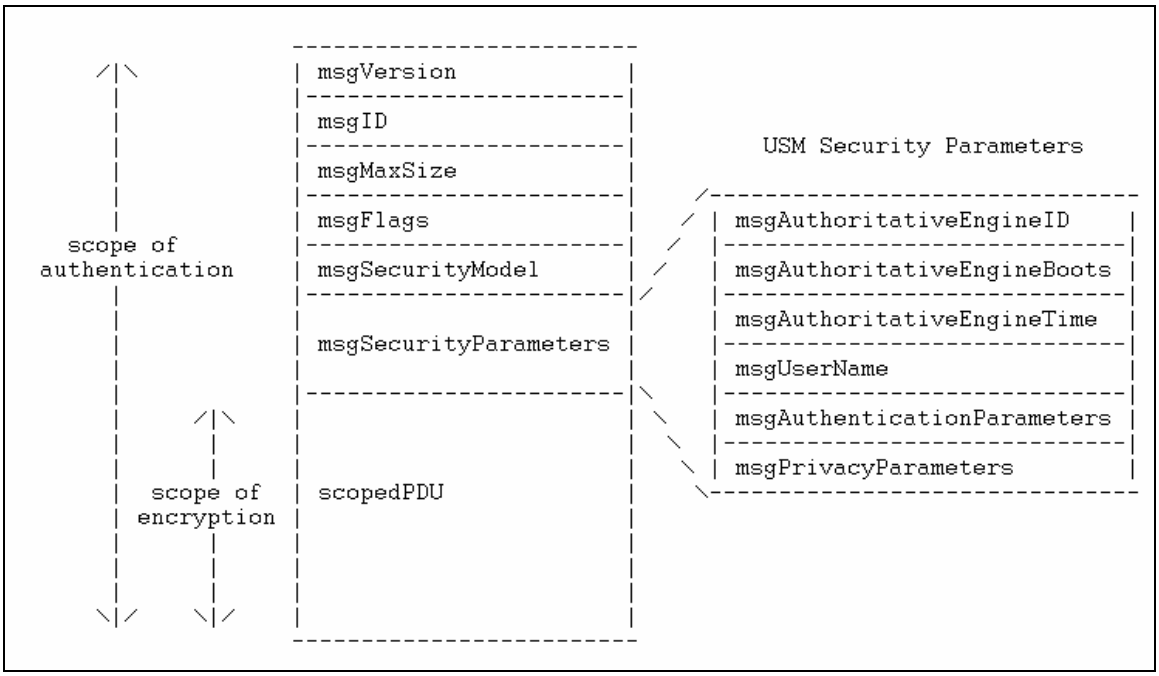

Abbildung 4: SNMPv3 Packetformat[59]

- MsgVersion gibt die SNMP-Version des Paketes an. [60]
- MsgID wird genutzt um Anfragen und Antworten zwischen Manager und Agent zu koordinieren; die MsgID einer Antwort muss identisch mit der Anfrage sein. [61]
- MsgMaxSize gibt die maximale Nachrichten Größe an die ein Sender fähig ist von einer anderen SNMP-Engine zu empfangen. [62]
- MsgFlags ist ein einzelnes Oktett, welches mitteilt wie die Nachricht Bearbeitet werden muss[63]

[59] [DA03]

[60] Vgl.: [DA03] S.3

[61] Vgl.: [DA03] S.3

[62] Vgl.: [DA03] S.3

[63] Vgl.: [DA03] S.3

- MsgSecurityModel definiert das Sicherheitsmodell mit dem diese SNMPv3-Nachricht erstellt wurde und beinhaltet somit die Information nach welchem Modell die Nachricht zu decodieren authentifizieren ist.[64]
- MsgSecurityParameters beinhaltet die spezifischen Informationen zu dem Verwendeten Sicherheitsmodell. Diese Daten werden durch das verwendete Sicherheitsmodell definiert, und auch nur von diesem genutzt. [65]

View Access Control Model

View Access Contol Model (VACM) ist das Sicherheitsmodell das den Zugriff auf die einzelnen OIDs der Agenten regelt. Dazu vergibt es im Gegensatz zu dem User Access Model Zugriffsrechte für Gruppen, nicht für einzelne Nutzer. Das heißt das es zum Beispiel eine Gruppe von Managern geben kann die uneingeschränkte Lese- und Schriebrechte hat und eine Gruppe die nur Leserechte für einen Teil des MIB-Baumes hat. Diese Zugriffsrechte werden in verschiedenen Tabellen auf jedem Agenten gespeichert. Die Parameter MsgFlags, MsgSecurityModel, und die eigentliche PDU werden an den AccessContol-Mechanismus übergeben der anhand der Informationen aus den Tabellen für jede OID eeinzeln entscheidet ob zugriff gewährt wird oder nicht. Falls für ein OID der Zugriff verweigert wird, wird eine Fehlermeldung ausgegeben und die Bearbeitung der Nachricht gestoppt.[66]

Weitere Gefahren

Eine Gruppe von Sicherheitsspezialisten der Universität von Oulu in Finnland haben SNMP und SNMP managebare Agenten auf ihre Sicherheit hin untersucht – „mit verheerendem Ergebnis“[67]
SNMP-Agenten und –Manager wurden mit „[...]ungültigen SNMP-Paketen, überlange Community-Namen, undefinierte oder extrem große Einträge in die Felder für den Nach-

[64] Vgl.: [DA03] S.3
[65] vgl. [DA03] S.3
[66] vgl. [DV03] S. 1-4
[67] [EC02] S.46

richtentyp, ASCII-Zeichenketten mit Sonderzeichen oder ohne Inhalt [...]“[68] bombardiert und so die Reaktion der SNMP-Implementierung getestet.
Sehr viele SNMP-Agenten erwies sich als anfällig für diese Angriffsversuche. Zu den fatalen Reaktionen der Agenten zählten einfache Abstürze, Denial-of-Service-Symptome und Ausführung beliebigen Codes durch die Systeme, auf denen die Agenten laufen.[69]

8 Analyse der erzeugten PDU´s

8.1 PDU allgemein

PDU-Definition:
Protokoll-Dateneinheiten (PDU) dienen der Kommunikation zwischen gleichberechtigten Protokollschichten und werden deshalb auch als Kommunikationsprotokolle bezeichnet. Die PDUs sind durch die Schichtenbildung in jedem Knoten ineinander verschachtelt. In der Datenquelle fügt jede Schicht den durchlaufenden Datensätzen Kontrollinformationen hinzu. Diese werden in der Datensenke in den entsprechenden Schichten abgearbeitet. Das bedeutet, dass in der Datenquelle den Daten zuerst in der Transportschicht eine Transport-PDU, danach in der Netzwerkschicht eine Netzwerk-PDU und zuletzt in der Sicherungsschicht eine Sicherungs-PDU hinzugefügt wird.[70]

PDU bedeutet Protocoll Data Unit. In den meisten Fällen ist die PDU ein einfacher Header Block oder bzw. "Einleitung", in der einige Identifikations-Informationen, gefolgt von einem Payload of Data, enthalten sind. Die Payload ist ebenfalls eine PDU, die von der nächst höheren Layer empfangen wird. Je mehr Layer eine Nachricht passieren muß, um so größer ist die Zahl der Header die ihr angefügt werden. Die PDUs dienen zur Kommunikation zwischen den gleichberechtigten Protokollschichten. Jede Schicht produziert dabei ein n-PDU, desen Format und Inhalt sowie die Art in der die PDUs zwischen den Netzwerkelementen ausgetauscht werden vom Schicht-n-Protokoll definiert wird. PDUs sind durch die Schichtenbildung in jedem Knoten ineinander verschachtelt.

[68] [EC02] S.46
[69] vgl. [EC02] S.46
[70] [SL03]

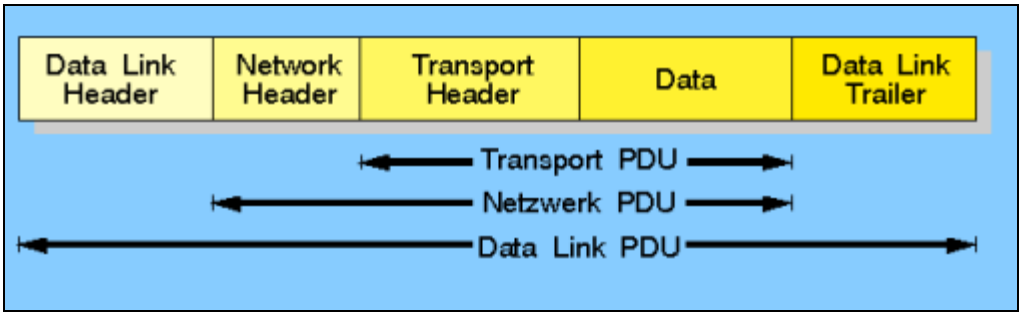

Abbildung 5: PDU allgemein[71]

In der Datenquelle sind die reinen Nutzdaten – Daten eines Datenpakets die keine Steuer- oder Protokollinformation beinhalten – als PDU anzusehen. In der Application Layer wird ein Header H7 erzeugt. Die Nutzdaten und der Header H7 werden der Presentation Layer zugefügt, die neu entstandene PDU wird auch Application-PDU genannt. In jeder Schicht entstehen neue PDUs, jeweils ergänzt um den Header der darüber liegenden Schicht. In jeder Schicht werden so den durchlaufenden Datensätzen Kontrollinformationen zugefügt. Die Header erhalten die zusätzlichen Informationen, die von der sendeten und der empfangenden Seite der Schicht benötigt werden, um den Dienst zu implementieren, der von der Schicht für die nächst höherer Schicht bereitgestellt wird.

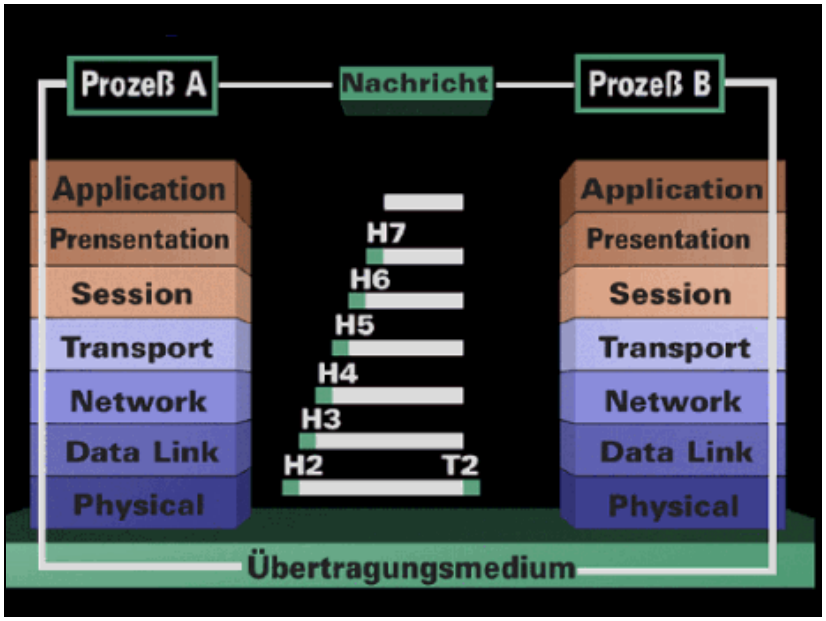

Abbildung 6: Multiplexen[72]

[71] vgl. [SL03]

Aufgaben einer PDU

Alle Informationen die in einem Ethernet Netzwerk verkehren sind in Rahmen eingekapselt. Ein Rahmen selbst ist eigentlich nur ein Datenpaket. Solche Pakete werden Frames (Rahmen) genannt, da die speziellen Synchronization and Error Detection Bits den Daten voraus und hinterher gehen um sie zu "Framen" (einzurahmen).
Bevor Daten über das Netzwerk gesendet werden können, müssen sie Stück für Stück in das jeweilige PDU-Format einer jeden Schicht, die sie passieren, gekapselt bzw. multigeplext werden. Jede neue PDU-Kapsel legt sich um die vorherige. Erst wenn dies erledigt ist, passieren die Daten die Physical Layer und werden in Netzwerk gesendet.

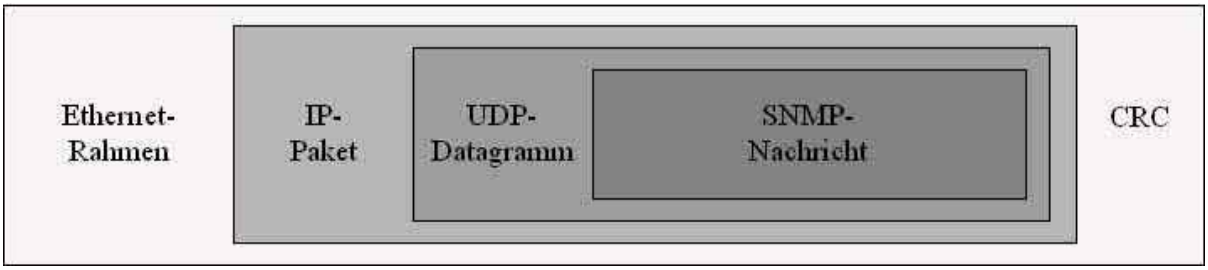

Abbildung 7: Multiplexen einer SNMP-Nachricht

Wenn die Daten die Zieladresse erreicht haben, müssen sie wieder entkapselt bzw. demultigeplext werden, bevor sie verarbeitet werden können => während die Daten eine Schicht nach der anderen passieren, werden sie Stück für Stück entblättert. Die Informationen, die die PDUs enthalten, teilen den Protokollen der einzelnen Schichten mit, was sie mit den Daten machen sollen (Error Detection und Correction, message routing, usw.)

Inhalt einer PDU

Jede PDU hat andere Inhalte. Generell kann man jedoch sagen, welche Inhalte auf welcher Schicht der PDU zugefügt werden. Wie schon oben erwähnt wird auf jeder Schicht eine neue PDU zugefügt. Mit Hilfe des Programms Etheral wurden die verschiedenen Rahmen einer SNMP-Nachricht dargestellt. Die SNMP-Nachricht ist durch vier Schichten gelaufen.

[72] vgl. [SL03]

8.2 SNMP-PDU

Eine SNMP-Nachricht besteht aus einer Preamble und einer angehängten Applicatioin-PDU, die auch SNMP-PDU genannt wird. Kurose beschreibt die eigentliche Nachricht ebenfalls als PDU, nach Kurose ist die gesamte SNMP-Nachricht eine PDU. Klein fängt erst mit der SNMP-PDU an von einer PDU zu reden und bezeichnet die Teile Length, Version und Community als Preamble der SNMP-Nachricht, die folgenden Darstellungen beziehen sich auf die letztere Definition.

Preamble			**SNMP-PDU**	
Length	Version	Community	PDU-Header	PDU-Body

Abbildung 8: Die SNMP-Nachricht

Die weitere Beschreibung der PDU soll an Hand einer übermittelten SNMP-Nachricht beispielhaft dargestellt werden.
Dafür wurde zwei PCs mit einem Router verbunden. Auf den PCs wurde mit Microsoft Windows XP Professional als Betriebssystem gearbeitet. Nach dem das SNMP-Protokoll installiert wurde, sowie der Router für SNMP konfiguriert war, konnte man über SNMP Nachrichten versenden und die Vorgänge mit einem Analyser „Ehtereal“ mitschreiben.
In Abbildung XY sieht man das Fenster des Analysers. Hier erkennt man die Befehlte mit denen die Nachrichten zum Router oder zum Computer geschickt wurden und die Antworten zurückgesendet wurden. Es wurde die Funktionen GET benutzt um Werte der MIB-Varibalen abzurufen, GET NEXT um den nächsten Wert in Bezug auf das gerade bearbeitete Objekt abzurufen und schließlich erhielten man die Meldung RESPONSE als Antwort auf die vorher genannten Requests an das NMS.

GET-Request

Der GET-Request wird von NMS ausgelöst, welcher die Anfrage an den Agent sendet. Der Agent erhält ihn und führt ihn bestmöglich aus. Einige Netzwerkkomponenten die stark beansprucht werden wie Router zum Beispiel, lassen den GET-Request einfach untergehen. Üblicherweise sollte der Agent jedoch ein GET-Response zu dem NMS zurücksenden.

Im GET-Request ist ein so genannter Varbind enthalten. Das ist eine Liste von MIB-Objekten, die dem Empfänger mitteilen was der Sender wissen möchte. Die Varbinds sind verantwortlich, dass der Sender genau die Informationen erhält, die erhabne möchte.
Zum Beispiel führt man einen Unix Befehl aus snmpget, um die Management Daten einer bestimmten Maschiene auszulesen. Zusätzlich werden community string und die OID 1.3.6.1.2.1.1.6.0 als Abfragedaten angegeben. Die .6 ist die OID Variable die ausgelesen werden soll, welche lesbar sysLocation genannt wird. Als Antwort erhält man system.sysLocation.0="", das bedeutet, dass die Variable nicht gesetzt ist.
Nun zur ebenfalls zusätzlich angehängten .0. Bei SNMP werden MIB Objekte nach der Konvention XY definiert. X ist die eigentilcihe OID des gemanagten Obejektes und Y ist der Instance Identifier. Bei Scalaren Obejekten ist Y immer 0. Scalare Objekte sind Ojekte die nicht als Reihe in einer Tabelle definiert sind. Falls es sich um eine Tabelle handelt, kann man mit dem Instance Identifier eine spezielle Zeile auswählen.[73]

[73] [MA01]

8.3 SNMP-PDU in einem Analyser

Um einen RESPONSE bzw. RESPONSE-PDU zu erhalten, muss man erst eine Anfrage schicken. Diese wird durch eine GET und danach eine GET-Next Funktion gestellt.

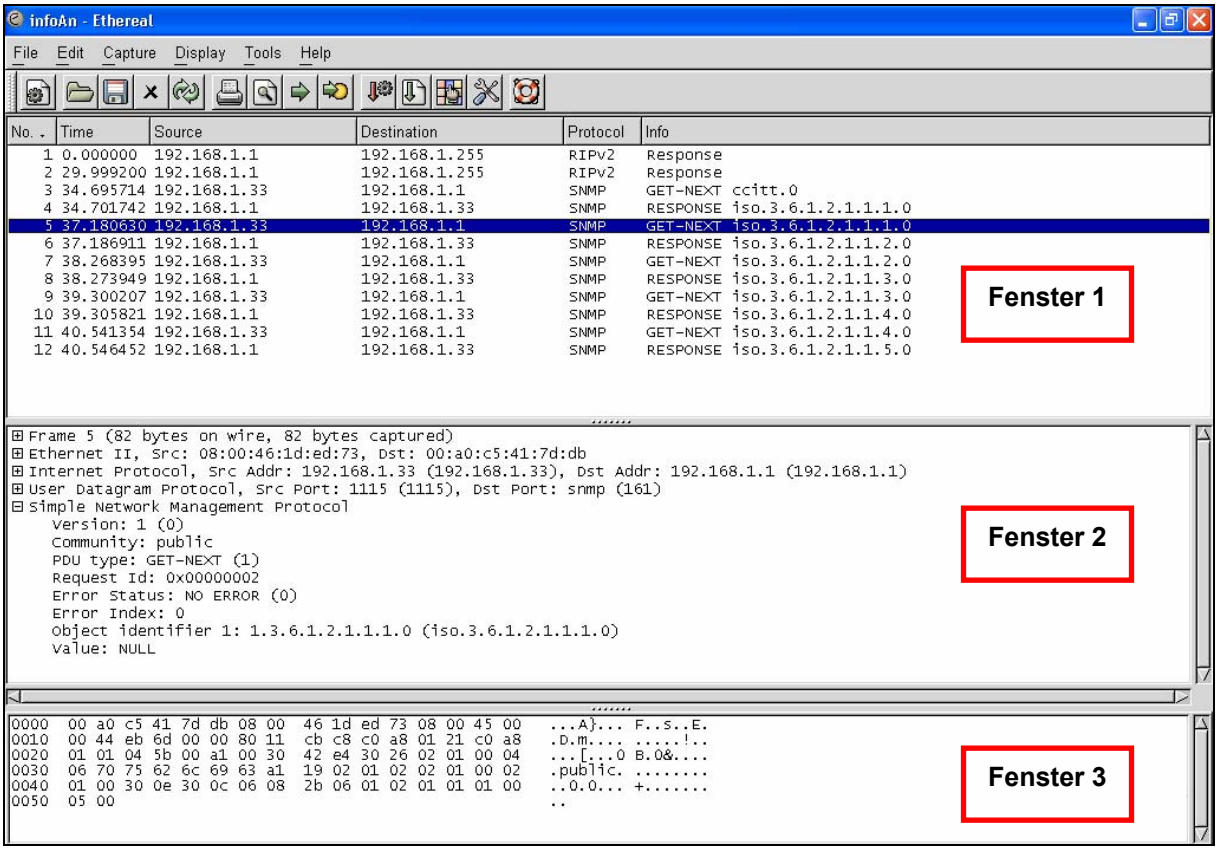

Abbildung 9: Get-Next Funktion im Etheral-Analyser

In Abbildung 9 ist mit dem Etheral-Analyser eine GET-NEXT Funktion analysiert worden. In Fenster 1 sind die üblichen Basisinformationen aller gesendeten Nachrichten abgebildet, hier erkennt man, dass es sich dabei um eine GET-NEXT Nachricht handelt.
In Fenster 2 kann man die Unterschiede zur später folgenden RESPONSE erkennen. Während Version und Community natürlich noch die Gleichenden wie in der RESPONSE sind, ändert sich jedoch die PDU Inhalte. Der PDU Type steht nun auf GET-NEXT. Mit der GET-NEXT Operation wird der nächste Wert gerettet und gespeichert. Vielen Daten, die aus Tabellen und MIB-Variablen zurückgegeben werden sollen, kann man keinen speziellen Namen zuweisen. Daher kann man sie nur mit dem GET-NEXT-Befehl ansprechen. Da der GET-NEXT Befehls auf 2 gesetzt ist, wird der folgende REPONSE die Request ID 3 enthalten. Der OID verweist bei GET-NEXT auf einen anderen Zweig

der Variablen des gemanagten Objektes innerhalb der MIB als bei der RESPONSE Nachricht.

Das Value der PDU steht auf NULL, was bedeutet, dass die MIB Variable einen uninteressanten Wert enthält. Er fungiert als Platzhalter, da ASN.1 NULL kein echter Datentyp ist.

Die folgende RESPONSE-PDU ist eine Antwort auf den GET-NEXT Befehl respektive die GET-NEXT-PDU.

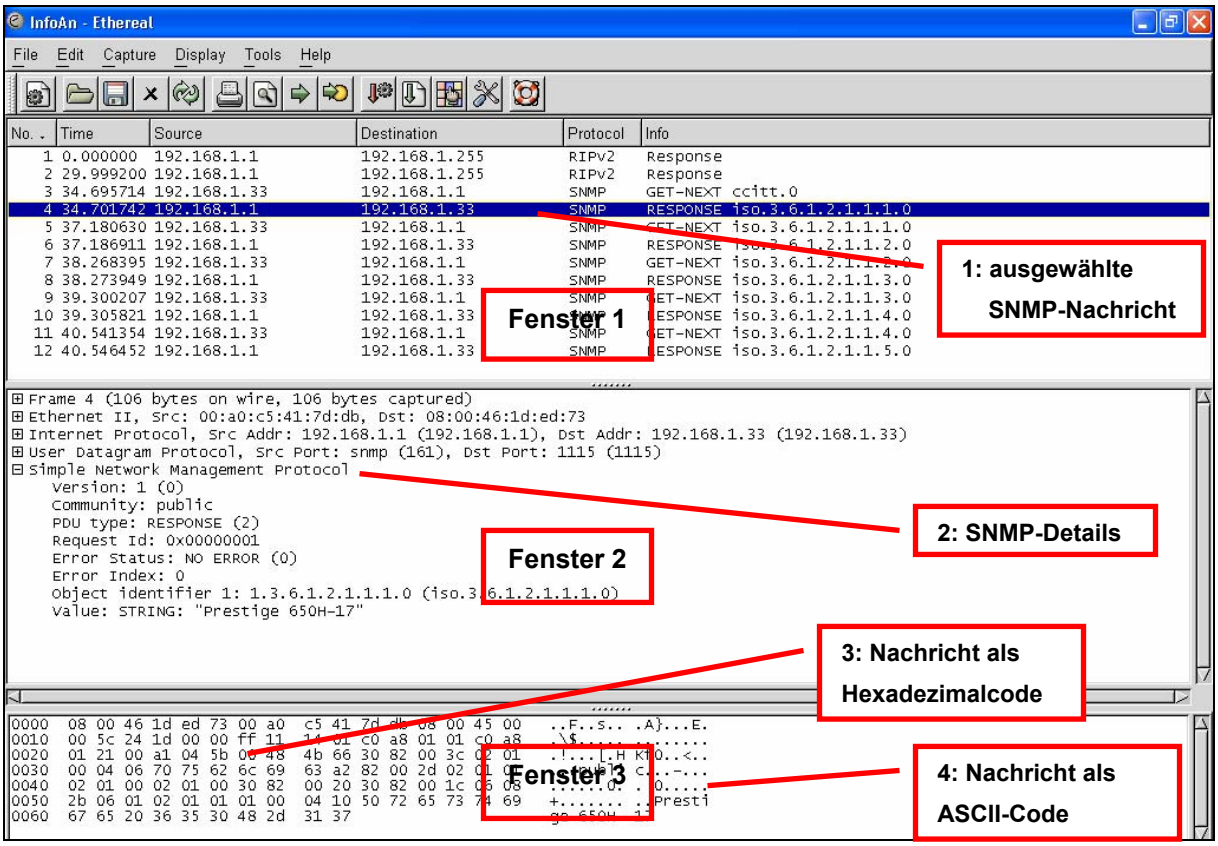

Abbildung 10: Etheral-Analyser

In Abbildung 10 ist der Etheral-Analyser abgebildet. Dieser ist in drei Abschnitte aufgeteilt. Im Abschnitt 1 wurde eine Nachricht markiert, welche blau unterlegt ist. Die Nachricht, hier durch eine gekennzeichnet ist ein RESPONSE. In der markierten Zeile kann man außerdem die IPs, OID und Zeit entnehmen. In Abschnitt 2 kann man diese markierte Nachricht detailliert betrachten. In Fenster 3 ist diese als Hexadezimalcode und interpretierter ASCII-Code abgebildet. Der Hexadezimalcode wird in Fenster 2 für jeden lesbar in Worten dargestellt.

Ähnlich der Kapslung werden in Abschnitt 2 die vier Schichten der Nachricht angezeigt. Die äußerste Kapsel ist der Ethernet-Rahmen (siehe hierzu auch Abbildung Nr. 3), welcher immer die MAC-Adresse der kommunizierenden Maschinen erhält. Im Ethernet-Rahmen findet sich das IP-Packet, welches unter anderem die IP-Adressen der kommunizierende Maschinen erhält. Wenn man eine Stufe weiter nach innen geht kommt man zum UDP-Datagramm, welches die Ports des sendenden und des empfangenden Hosts enthält. Da SNMP das verbindungslose UDP Transportprotokoll benutzt und UDP selbst nicht zuverlässig ist, muss SNMP durch seine eigenen Sicherheitsfunktionen, Zuverlässigkeit beweisen.
Im innersten finden wir die SNMP-Nachricht (siehe Markierung 2: SNMP-Details). Diese wurde hier detailliert darstellt. Nun wird zuerst die benutzte SNMP Version angezeigt. Oben ist geschrieben 1 (0), was bedeutet, dass SNMPv2 benutzt wurde. Gefolgt von dem Community String von dem Host der die Nachricht empfangen soll. Nun folgt die Protocoll Data Unit der SNMP-Nachricht, auch SNMP-PDU genannt. Diese wird anhand des Hexadezimalcodes erklärt und ist in Abbildung 11 blau und grün dargestellt. Der Hexadezimalcode ist der Code aus dem Etheral-Analyser farblich aufbereitet.

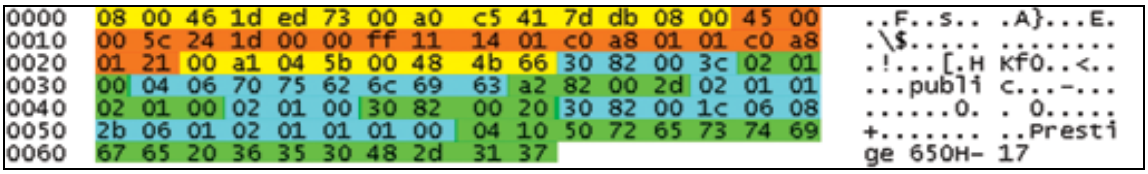

Abbildung 11: SNMP-Nachricht als Hexadezimalcode

Die erste gelbe Reihe stellen die MAC-Adressen auf der Ethernet-Schicht da. Die folgenden orangnen Zahlen sind der IP-Header, welche übersetzt die IP-Adressen des Senders und Empfängers und die Header-Länge in Bytes enthält. Die zweite gelbe Reihe stellt das UDP-Datagramm da.

In der folgenden Tabelle wird die Bedeutung der einzelnen Hexadezimalzahlen der SNMP-Nachricht dargestellt:

30	Start der SNMP-Nachricht (Sequence)
82 00 3c	Länge der Daten in der Sequenz: 60 Bytes
02 01 00	Version: SNMPv1
04 06 70 75 62 6c 69 63	Community String: Public
a2	PDU-Type a2 steht für GET-RESPONSE
82 00 2d	Gibt die Länge an: 45 Bytes
02 01 01	Reqeust ID
02 01 00	Error-Status 0
02 01 00	Error-Index 0
30 82 00 20	Der Payload hat Sequenz mit Länge 32…
30 82 00 1c	…enthält eine Sequenz der Länge 28…
06 08	…darin ist der OID von der Länge 8
2b 06 01 02 01 01 01 00	Object Identifier: 1.3.6.1.2.1.1.1.0
04 10 50 72 65 73 74 69 67….	Gibt den Value String zurück

Tabelle 3: Interpretierter Hexadezimalcode

Quellenverzeichnis

Literatur

[KA89]	Kauffels, F. J.; *Lokale Netze,* DATACOM, 1989
[KA92]	Kauffels, F. J.; *Netzwerk-Management,* DATACOM, 1992
[KU02]	Kurose, J. F.; *Computernetze Ein Top-Down-Ansatz mit Schwerpunkt Internet,* Addison-Wesley, 2002
[EC02]	Eckel P.; *Stille Helfer unter Beschuss,* C't Ausgabe 05/2002 Seite 46
[TA97]	Tanenbaum A. S.; *Computernetzwerke 3. Auflage*, Prentice Hall Verlag GmbH München, 1997
[FR02]	FROX communication: *R&D – Einführung in SNMP 1.2*, 2002
[MA01]	Mauro D., Schmidt K.; *Essential SNMP*, O'Reily, 2001

Internet

[US99]	*User-based Security Model (USM) for version 3 of the Simple Network Management Protocol (SNMPv3)*, April 1999
[DA03]	Davis E.; *SNMPv3 – User Security Model*, http://www.foobargeek.com/docs/usm.html
[SE03]	*Security in SNMPv3 versus SNMPv1 or v2c*, htp://www.aethis.com/solutions/snmp_research/snmpv3_vs_wp.pdf
[KL03]	Klein J.; http://www.jklein.de/techniker_arbeit/tech_html/snmp_allgemein.htm
[SL03]	http://w3.siemens.de/solutionprovider/_online_lexikon

9 Anhang

- Arbeitsaufteilung.....Seite B
- Programme und Tools.....Seite C

Arbeitsaufteilung

Kapitel	Bearbeitet von:
1	Wulst
2	Wulst
2.1	Wulst
2.2	Wulst
2.3	Wulst
2.4	Wulst
2.4.1	Wulst
2.4.2	Wulst
3	Andersick, Spohn, Wulst
3.1	Andersick, Wulst
3.2	Wulst
3.3	Wulst
3.4	Wulst
3.5	Wulst
3.6	Spohn
4	Andersick
5	Andersick
5.1	Andersick
5.2	Andersick
5.3	Andersick
6	Klahre
7	Klahre
7.1	Klahre
7.2	Klahre
7.3	Klahre
7.4	Klahre
8	Spohn
8.1	Spohn
8.2	Spohn
8.3	Spohn

Programme und Tools

- Etheral Analyser
- SolarWinds 2001 Engeneers Edition
- SNMP Manager Lorio Pro V200
- SNMP Manager SNMPC600 Eval
- TrapGEM