SNMP

Zhizhong Chen

Zusammenfassung-

Um die Probleme, die in einem Netzwerk auftreten können, schnell und sicher mit SNMP beseitigen zu können, muss man SNMP fest im Griff haben.

SNMP ist ein Kommunikationsprotokoll,das für Management von TCP/IP-Netzwerken eingesetzt wird und ist Teil der TCP/IP- Protokollsuite

Dieser Vortrag befasst sich mit den betrieblichen Merkmalen von SNMP sowie mit der logischen Struktur von Management-Informationen.

SNMP Grundlagen

Das SNMP ist ein Protokoll der Anwendungsschicht, wurde für das Managen von verteilten und heterogenen Netzwerken entwickelt und gehört zu der TCP/IP Protokollfamilie.das erlmöglicht den Austausch von Managementinformationen zwischen den in einem Netzwerk befindenen Geräte .SNMP erlaubt dem Netzwerk-Administrator die Netzwerk-Verwaltung durchzuführen und hilft ihm die Netzwerkleistungen zu beobachten,um eventuelle Netzwerkprobleme schon im Vorfeld aufzufinden und um das Wachstum des Netzwerks zu planen.

.

SNMP wurde für das Managen von verteilten und heterogenen Netzwerken entwickelt.es wird auch als Standardmanagement angesehen.Es gehöhrt zu der Gruppe der TCP/IP-Anwendungsprotokolle und ist in erster Linie für den Transport von Kontrolldaten zuständig.Das Protokoll ermöglicht den Informationsfluss von Status- und Statistikdaten zwischen einzelnen

Netzwerkelementen und einem Netzwerkmanagementsystem.

Das Netzwerkmanagement mit SNMP besteht aus einem Modell mit verschiedenen Komponenten.davon sind Management-Station und Management-Agent die beiden Kern-Komponenten.

Management-Station

Die Management-Station ist ein Netzwerkknoten mit besonderen Funktionen,der dem Netzwerk-Management-System als Schnittstelle zum Netzwerkadministrator dient.
die sind im Allgemeinen universelle Rechner mit einer speziellen Managementsoftware,die als Management-Stationen eingesetzt werden.

Diese Management-Station führen eine oder mehrere Prozesse aus ,die mit den Management-Agenten über das Netzwerk kommunizieren ,dabei werden Befehle erteilt oder Antworten erhalten.

Mittels des Managementprotokolls kann die Management-Station den Zustand von lokalen Objekten der Agenten abfragen bzw. Verändern,d.h. die Management-Station erledigt die Hauptverarbeitung der Netzwerkinformationen ,die durch die Agenten gesammelt und bereitgestellt werden.

Der Netzwerkadministrator hat mit der Management-Station ein Interface mit dem er das Netz kontrollieren und beobachten kann.

Management-Agent

Der Management-Agent ist eine weitere,zentrale Komponente in der Netzwerkmanagement-Architektur.

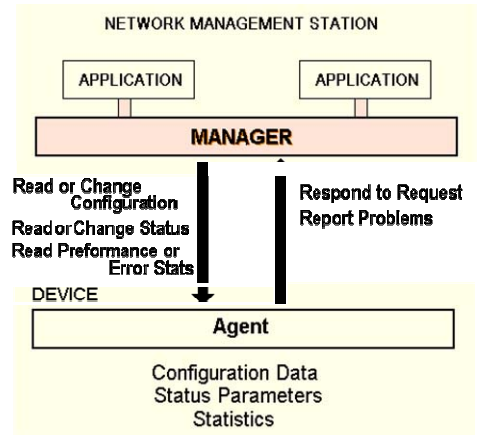
Die Aufgabe eines Agents besteht darin, einen oder mehrere Networkknoten zu überwachen, Daten zu sammeln (Management Information), und diese Daten an eine Management-Station zu senden.

Die wichtigen Netzwerkknoten sind z.B. Hubs,Bridges,Router,Drucker oder Host usw.

Ein Agent ist genau genommen eine Software,die auf dem verwalteten Gerät läuft.Er ist für die Erfassung und Speicherung der Informationen über das verwaltete Gerät zuständig.

Management-Agenten sind also verwaltete Netzwerkknoten,die in der Lage sind intern mittels SNMP mit der Management-Station zu kommunizieren.

Abb. Interactions between a Manager und a Agent



Managed Objekte

Die von der Agenten gesammelten direkt zur den verwalteten Geräte relevanten Informationen werden als Managed Objekte genannt.Die Managed Objekte können z.B. folgendes sein:

-die Konfigurationsparameter

-das auf dem verwalteten Rechner läufte Betriebssystem und

Systeminformationen

-Anzahl der vom verwalteten Rechner empfangenen UDP-Datagramme

-u.s.w

Die Aktivität der management-Station besteht aus der Auswertung und ggf.der Aktualisierung oder Veränderung von einem oder mehreren managed Objekten des verwalteten Geräte.Deswegen müssen solche Objekten logisch zugriffbar sein.d.h. diese managed Objekte müssen organisiert und irgendwo gespeichert werden. Die Datenbank,wo die managed Objekte gespeichert sind ,nennt man Management Information Base(MIB).um die managed

Objekte in der MIB organisieren und identifizieren zu können,wird SMI verwendet.

Die Kommunikation zwischen der Management-Station udn dem Management-Agenten erfolgt mit Hilfe von SNMP.

SMI

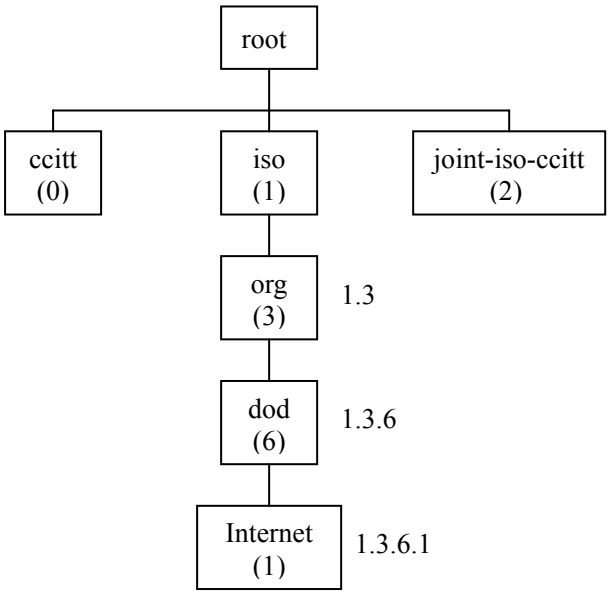
Das SMI ist ein Modell zur Beschreibung der Struktur der benötigten Informationen im Netzwerkmanagement.

D.h. das SMI stellt ein Verfahren zum Benennen und Organisieren von Objekten bereit.

Es kennzeichnet Datentypen,die in der MIB verwendet werden und es definiert noch, wie Ressourcen in der MIB dargestellt und bezeichnet werden.

Die SMI definiert die Regeln für die Definiton der Managed Objekte.Der Zugriff auf diese Objekte erfolgt mit Hilfe eines Netzwerkmanagementprotokolls ,wie beispielsweise dem SNMP.

Die SMI muss standardisierte Darstellungsformen anbieten, ,um eine standardisierte Darstellung für Daten zu gewährleisten.



Die SMI verwendet eine Baumstruktur,in dem die diversen Objekte die Blätter darstellen.Das ermöglicht eine einfachere,bildliche Vorstellung vom Internet.

Den Objekten wird hierbei eine Folge von ganzzahligen Werten ,die sogennanten object identifier zugewiesen. Diese object identifier dienen der Kennzeichnung seiner Positon im Baum.

Der Baumwurzel hat keine Bezeichnung,sondern verfügt über drei untergeordnete Äste.

Diese drei äste werden von verschiedenen Einrichtungen,die Standards festlegen,verwaltet.

Die Verwaltung des Astes 0 wird von der ITU(International Telecommunication Union),die Verwaltung vom Ast 1 wird von der ISO und die Verwaltung vom Ast 2 wird von beiden Institutionen gemeinsam durchgeführt.

Die ISO selber unterteilt ihren Ast für mehrere Organisationen.So hat sie ihren Ast 3 mehreren internationalen Organisationen zugeordnet (org).

Eines dieser Äste hat sie dem amerikanischen Verteidigungsministerium zugeordnet(dod).

Für Internet-Objekte (internet) wird der erste Zweig dieses Astes verwendet.

Um ein Internet-Objekt zu erreichen,beginnen alle Objekt-Identifier mit [1.3.6.1].

Der Knoten internet enthält weitere Knoten:

directory -reserviert für zukünftige OSI-Anwendungen

mgmt -für vom IAB beschäftigete Objekte { IAB (Internet Architecture Board)}

experimental -experimentelle Objekte

private -private Objekte

Im mgmt –Unterbaum ist auch die MIB zu finden,welche mit 1.3.6.1.2.1 referenziert wird.

SMI-Datentypen

Im SNMP verwendete SMI-Datentypen:

Counter: Dies ist ein nicht-negativer Integerwert,der Werte von 0 bis 2(hoch 32)-1 annimmt und beim Überlauf auf 0 gesetzt wird.
Gauge: Dies ist ein nicht-negativer Integerwert,der Werte von 0 bis 2(hoch

32)-1 annimmt.Die Werte können inkrementiert oder dekrementiert werden. Bei Erreichen des Maximalen Wertes verändert sich sein Wert beim weiteren Inkrementieren nicht.
Time Ticks : ein Counter,der die hundestel-Sekunden eit einem Zeitpunkt misst.

Die Syntax für die Definiton von einem Objekt:

Im SMI wird ein Makro definiert ,um die Definition von Objekten zu vereinheitlichen.Dieses Makro definiert die Syntax eines Objektes,die Zugriffsmöglichkeiten auf ein Objekt und den Status eines Objektes.Jedem Objekt wird zusätzlich eine Beschreibung in Textform hinzugefügt.

Diese Beschreibung dient der nähreren Erklärung des Objektes

(objectname) OBJECT-TYPE

SYNTAX (syntax)

ACCESS (access)

DESCRIPTION (description)

::= { (parent)(number)}

(Objektname) beinhaltet den offiziellen Namen des SNMP-Objektes.Er sollte mit einem Kleinbuchstaben anfangen.

OBJECT-TYPE ist Schlüsselwort,bedeutet hier ist ein neu Objekt-Type zu definieren.

SYNTAX ist auch ein Schlüsselwort ,die gibt den Typ des Objektes an.z.B. Integer,IpAddresse,Counter u.s.w.

(parent) gibt die Objekt-Identifier des Vorgänger von diesem Objekt an,(number) bezeichnet die Position von diesem Objekt innerhalb des Zweigs von seinem Vorgänger.

Mit dem Schlüsselwort ACCESS wird der Zugriff auf das definierte Objekt festgelegt.

Die management Information Base (MIB)

Eine Management Information Base ist eine Datenbank,die die zu verwaltenden Elemente einthält.Die Management Information Base ist in jedem Netzwerkmanagementsystem ein wesentliches Element.

Es wurden Standard-MIBs definiert und veröffentlicht. Zudem kann die Standard-MIB durch herstellereigene MIBs ergänzt werden.Anbieter können also eigene Zweige definieren ,in denen sie die verwalteten Objekte für ihre eigenen Produkte anbieren.

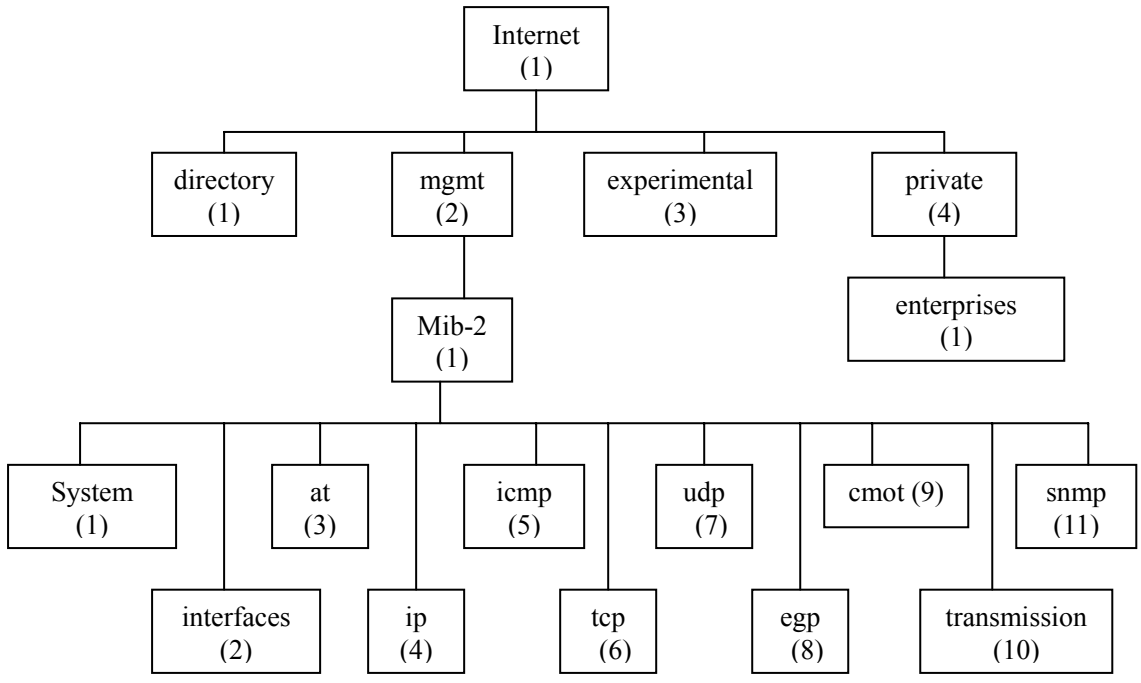
Jedes verwaltbare Gerät enthält eine MIB.Die Informationen,die in diesen MIBs verwaltet werden,werden als MIB-Objekte bezeichnet.

Jedes dieser Objekte wird über die Objekt-ID eindeutig gekennzeichnet und stellt eine von den Eigenschaften eines verwalteten Gerätes dar.

Durch die Object-ID wird das verwaltete Gerät in der MIB-Hierarchie eindeutig gekennzeichnet.

Die Hierarchie des MIBs kann dabei als ein Baum betrachtet werden. Die einzelnen Ebenen dieses Baumes werden durch verschiedene Oragnisationen verwaltet.

z.B. die folgende Abbildung zeigt den internet-Zweig der Standard-MIB:



Der Zweig snmp kann entweder über die Objektbezeichnung
Iso.org.dod.internet.mgmt.mib-2.snmp

Oder über die eindeutige Objekt-ID
1.3.6.1.2.1.11
referenziert werden

Die seit SNMPv2 verwendete MIB wird als MIB-II bezeichnet und die ursprüngliche als MIB-I bezeichnet.

Das Netzwerkmanagementsystem kann aus der MIB den Gerätetyp und die von diesem zur Verfügung gestellten Funktoinen auslesen.

Die MIB beschreibt formal Netzwerkobjekte.die seit SNMPv2 verwendete MIB wird als MIB-II und die ursprünliche als MIB-I bezeichnet.

Community Namen

eine Community-basierendes Administrations-Framework wird definiert, um mit ihm die verschiedenen SNMP Elemente verwalten zu können. Jede SNMP-Community ist eine Gruppe von Geräten, die mindestens einen Agent und ein Management-System beinhaltet. Den Namen, den diese Gruppe bekommt wird als Community-Name bezeichnet. Der Community-Name wird jeder SNMP Nachricht kodiert beigefügt und wird dann als Communutiy-String bezeichnet. Der Community-String informiert dann den Empfänger, für welche Community diese Nachricht bestimmt ist.

Ein gemanagter Node zeigt durch Annehmen oder Ablehnen der SNMP-Nachricht an, ob er zur deren Community gehört.
Ein Beispiel:
wenn ein Node alle Nachrichten, die den Community-String "*public*" enthalten, annimmt, zeigt er somit an, dass er zu der Community "*public*" gehört. Wenn er alle Nachrichten mit dem Community-String "*private*" ablehnt, zeigt er somit, dass er nicht Mitglied in dieser Community ist.

Die Bezeichnungen der einzelnen Communities werden vom Netzwerkbetreuer festgelegt und sollten eindeutig sein. So wäre der Name für eine Community, die die Geräte einer Entwicklungsabteilung beinhaltet, "Entwicklung". Der Namen sollte sorgfältig bedacht werden, da er später nicht mehr so leicht geändert werden kann.

Wird ein Node keiner Community zugewiesen, nimmt er normalerweise alle SNMP-Nachrichten mit einem beliebigen Community-String an. Dieser Node gehört somit zu allen SNMP-Communities in einem Netzwerk.

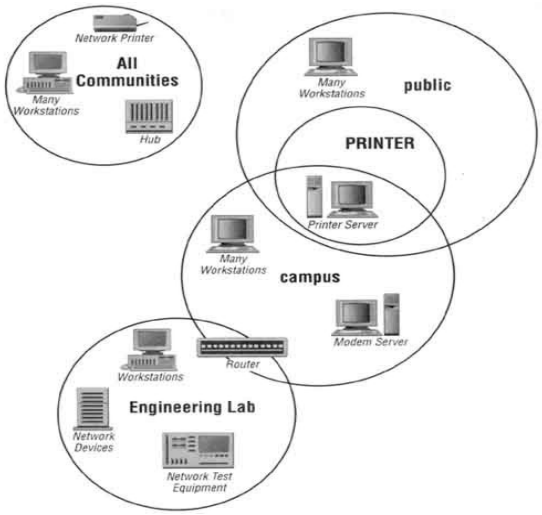


Abb. Beispiel Community

Diese Abbildung zeigt Ihnen die grafische Darstellung eines mit SNMP verwalteten Netzwerkes. Alle Nodes in diesem Netzwerk unterstützen einen Agent. Die meisten Nodes sind nur in einer Community vorhanden, aber einige existieren in zwei oder sogar drei Communities. Der Router z.B. gehört sowohl zur "campus" wie zur "Engineering Lab" Community. Die Community die "All Communities" genannt wird, beinhaltet alle Geräte, denen kein Community-Name zugewiesen wurd. Diese Geräte gehören somit automatisch zu allen anderen Communities im Netz.

Die SNMP-Message

Die SNMP-Messages kann man in fünf Arten unterteilen:

--get-Request-Message
-- get-Next-Message
--get-Response-Message
--set-Message
--trap-Message

Die get-Request-Message

Diese Message wird vom Manager an den Agenten gesendet.Sie wird für das Anfordern von Daten von einem managed Object verwendet.Mit jeder Message wird ein Wert aus einer oder mehreren MIBs abgefragt.

Die get-Next-Message

Diese Message ist eine Erweiterung der get-Request-Message und erleichtert die Anfrage von nachfolgenden Objekten.GetNextRequest wird primär für das durchlaufen von Tabellen und das Anfordern von speziell ausgewählten Werten (die natürlich von Tabellenmitgliedern stammen) verwendet.

Wenn Sie eine GetNext-Operation mit der OID einer *scalar variable* (ohne Angabe ihres *instance identifier*) ausführen, bekommen Sie den korrekten Wert dieser Variablen geliefert.

Wie Sie aus der OID erkennen können, handelt es sich um eine *scalar variable* und daher gibt es auch nur eine *instance*.
Wenn wir nun noch den *instance identifier* mit angeben, würde der Wert der nächsten Variablen ausgelesen :

Die get-Response-Message

Diese Message wird von einem Agenten an einen Manager als Antwort auf eine get-Request-Message gesendet.Sie beinhaltet die Daten,die von der get-Request-Message angefordert worden sind.Im value-Feld ist der angeforderte Wert enthalten.

Set-Message

Die Set-Message stellt die einzige Möglichkeit für ein Management System dar, Daten, die von einem Agent verwaltet werden, zu verändern.

Um eine Set-Operation auszuführen, wird eine SetRequest-Nachricht, mit einer varbind pro MIB Variablen, kreiert. Jede varbind beinhaltet den instance identifier und den neuen Wert der Variablen. Das Management System sendet die SetRequest-Nachricht an den Agent. Dieser durchläuft die varbinds und speichert alle Werte in einem Zwischenspeicher. Wenn der Durchlauf keine Fehlermeldungen, wie z.B. noSuchName, badValue, readOnly, tooBig oder genErr, liefert, werden die neuen Werte in die Variablen geschrieben. Kommt jedoch auch nur eine Fehlermeldung vor, wird alles verworfen, die alten Werte bleiben erhalten und eine Fehlermeldung wird an das Management System gesendet.

SNMP ist nur ein Bestandteil der Mechanismen, die für das Lesen und Schreiben von Variablen angewandt werden. Das kein Fehler durch die *GetResponse*-Nachricht angezeigt wurde, bedeutet nur, daß der SNMP-Dienst des Agents die Daten an weitere Dienste übergeben konnte. Was danach mit den für SNMP unbekannten Diensten passiert, wird nicht überprüft. Es kann daher vorkommen, daß die neuen Werte nicht übernommen werden.

Da ein SNMP Management System eine erfolgreiche GetResponse-Nachricht nicht als Beweis für die korrekte Ausführung der Set-Operation nutzen kann, muss jede Set-Operation in vier Schritte unterteilt und ausgeführt werden:

1. Das Management System erfragt den aktuellen Wert der Variablen mit einer Get-Operation.
2. Es sendet die neuen Werte mit der Set-Operation.
3. Es überprüft die neuen Werte mit Hilfe der GetResponse-Nachricht, die auf die Set-Operation folgt.
4. Es startet eine neue Get-Operation, um die Variablen erneut auszulesen. Sind alle Variablen richtig, wird hier abgebrochen. Wenn nicht, wird eine bestimmte Anzahl von Get-Operationen ausgeführt. Ändert sich nichts, werden die Variablen erneut gesetzt.

Nun wird Ihnen dies alles sehr umständlich und unausgereift vorkommen. Bedenken Sie aber, daß der Agent als ein Low-Level Prozess auf dem Netzwerk-Gerät ausgeführt wird. Die Aufgaben des Agents sollen stets so einfach wie möglich gehalten werden. Treten Probleme auf, sollen diese von dem Netzwerk Management System und nicht von dem Agent gelöst werden.

Die trap-Message

trap-Messages sind unaufgefordert versandte nachrichten.Sie werden von einem Agenten an das Management-System verchickt,sobald etwas Unvorhergesehenes und Bedeutendes für das Management-System passiert.

Ein Trap wird vom Agenten ausgelöst,sobald einvorbestimmtes Ereignis(dies kann auch ein unbekanntes sein) aufgetreten ist.Die Message wird dann an die Trap destination (eine bestimmte Netzwerk-Adresse) gesendet.

Zwei Versionen von SNMP existieren.die beiden besitzen viele Eigenschaften von gemeinsamen.aber SNMPv2 ist eine Weiterentwicklung von SNMPv1.

SNMPv1

SNMP –Messages sind immer in einem UDP-Datagramm eingekapselt.

Aufbau einer SNMP-Message

Eine SNMP-Message ist nach einfachen Schema aufgebaut.Sie besteht aus der SNMP Message Preamble und der SNMP Protocol Data Unit(PDU).

Nachfolgende Abbildung zeigt dieses Schema auf.

Message Length	Message Version	Community String	PDU-Header	PDU-Body
SNMP Message Preamble			SNMP Ptotocol Data Unit	

Die SNMP-Message Preamble besteht aus den Feldern Message Length,Message Version und dem Community String.
die Message Length beinhaltet die absolute Länge der Message.
die Message Version kennzeichnet die verwendete Version des Protocolls, wobei der Wert „0“ für SNMPv1,der Wert „1“ für SNMPv2 steht.
der Community String kennzeichnet die Community,zu der diese Message gehöhrt.

Die SNMP-Protocol Data Unit (PDU)

Die SNMP-PDU enthält die beiden Abschnitte PDU-Header und PDU-Body.

Der *PDU header* beinhaltet Informationen speziell zur SNMP Nachricht. Das könnten Informationen wie z.B.welche Operation ausgeführt werden soll usw. sein. Der *PDU body* (PDU Körper) beinhaltet die aktuellen Daten, die zur Ausführung eben dieser Operation notwendig sind.

PDU header

Bei den Request- und Response-Nachrichten ist der Header stets der gleiche. Der Trap-Header ist jedoch sehr unterschiedlich zu dem der anderen Messsages. Dies ergibt sich aus der speziellen Nutzung der Trap-Message

PDU Type
Zeigt auf den Typ, der von dieser Message dargestellten

SNMP-Operation.oai
Folgende Werte sind erlaubt und von RFC 1157 definiert :

0 getRequest
1 getNextRequest
2 getResponse
3 setRequest
4 trap

PDU Length
Die Länge, der nach dem P*DU length field* folgenden

SNMP-Nachricht. Dieser "Rest" der Nachricht wird in Oktetten angegeben.

Die folgenden Felder kommen nur in einer Trap-Nachricht vor.

Enterprise MIB OID

Die OID der management enterprise, welche die Trap-Nachricht definiert. Dieser Wert stellt sich

als ein OBJECT IDENTIFIER Wert mit variabler Länge dar.

Agent IP Adress

Die Netzwerk Adresse des Agents, der die Nachricht erstellt hat. Dieses Feld beinhaltet die IP Adresse

in Form eines OBJECT IDENTIFIERS. Wenn der Agent sich in einem Netzwerk befindet, daß nicht

mit IP sondern mit z.B. IPX arbeitet, stellt sich der Wert der Adresse mit Nullen dar (0.0.0.0).

Standard Trap Type

Identifiziert den Typ der Trap-Nachricht. Folgende Werte sind definiert :

0 coldStart
1 warmStart
2 linkDown
3 linkUp
4 authenticationFailure
5 egpNeighborLoss
6 enterpriseSpecific

Specific Trap Type

Zeigt auf die besondere Trap, die in einer besonderen Händler MIB definiert wird. Auf dieses Feld wird verwiesen, wenn der Wert des *"Standard Trap Type*"-Feldes 6 ist. In diesem Fall ist der Wert im *"Specific Trap Type*"-Feld größer Null, und zeigt mit ihrem Inhalt auf die besondere MIB, die die Trap definiert. Ist der Wert im *"Standard Trap Type*"-Feld ungleich 6, so wird der Wert im *"Specific Trap Type*"-Feld auf Null gesetzt und nicht weiter beachtet.

Time Stamp

Ein positiver 32-Bit Wert, der die Anzahl der 1/100 Sekunden angibt, die seit dem (Neu)Start des

SNMP-Agents und dem Senden dieser Trap-Nachricht vergangen sind. Dieses Feld wird mit dem Wert

der *sysUpTime*-Variablen zum Zeitpunkt des Sendens der Trap initialisiert.

Diese folgenden Felder sind in den PDUs aller SNMP Request- und Response-Nachricht definiert (also nicht in den PDUs der Trap)

Request ID

Ein Handshake-Wert, mit dessen Hilfe eine Response-Nachricht mit einer Request-Message aufeinander abgestimmt werden. Dies ist vor allen dann sehr hilfreich, wenn eine Management-System mehrfach Request-Nachrichten zur selben Zeit sendet und die empfangenen Response-nachrichten den verschiedenen Request-Nachrichten wieder zuordnen muss (auf dieses Kapitel werden wir später noch genauer eingehen).

Error Status

Der Wert in diesem Feld informiert über den Erfolg oder den Fehlschlag der SNMP Operation. Werte ungleich Null weisen auf ein Fehlschlag hin. Folgend die möglichen Werte (definiert von RFC 1157), die in diesem Feld enthalten sein dürfen.

Wert	Typ
0	noError
1	tooBig
2	noSuchName
3	badValue
4	readOnly
5	genErr

Error Index

Dieser Index hat nur einen einzigen Eintrag. Der Wert gibt die Nummer der Variablen an, in der der Fehler aufgetreten ist. Hat der Eintrag den Wert "1", handelt es sich um die erste Variable. Bei "3" ist es die dritte Variable usw. Ist der Wert von Error Status ungleich "0", der Wert von Error Index jedoch "0", so konnte der Fehler nicht lokalisiert, bzw. es handelt sich um einen unbekannten Fehler, der nicht zugeordnet werden konnte.

Oft handelt es sich dann um einen tooBig oder genErr-Fehler.

Request ID

Die meisten High-Level Kommunikations- Protokolle definieren ein header field. Es befinden sich Handshake Informationen, mit deren Hilfe Response-Nachrichten die passende Request-Nachricht finden. Bei dieser

Information handelt es sich um ein Synchronisations - Wert oder eine "Beziehungsetikette" (correlation tag). Bei der SNMP Nachricht ist dieser Wert im "Request ID"-Feld der Request- und GetResponse-Nachricht, gespeichert.

Die Request ID wird von vielen SNMP-Autoren etwas "stiefmütterlich" behandelt. Dies kommt daher, da das "Request ID"-Feld keine wichtigen Management Daten speichert. Dennoch ist es ein wichtiger Teil des Handshake-Mechanismus. Der Agent nutzt dieses Feld nicht selbst, sondern kopiert dessen Wert in das *"Request ID*"-Feld der Response-Nachricht, die er dann an den anfragenden Node sendet.

Das Management System nutzt die *Request ID* um die Response-Nachricht der richtigen Request-Nachricht zuzuordnen

Jede dieser Nachrichten werden von dem Agent, der sie empfangen hat, bearbeitet. Bevor er dann die Antwort an das Management System sendet, kopiert er die ID in die GetResponse-Nachricht. Danach wird sie abgesendet. Das Management System ordnet dann die Response-Nachricht dem richtigen Timer zu, speichert den Wert und löscht den Timer. Ist der Timer vor dem Empfangen der Nachricht abgelaufen, wird diese als verloren deklariert und eine erneute Anfrage wird gesendet. Wird eine festgelegte Anzahl von Anfragen nicht beantwortet, schließt das System daraus, daß der Agent nicht bereit ist, beendet daraufhin das Senden von Anfragen und meldet dies dem Netzbetreuer.

Die Request ID der Request-Nachricht wird auch zur Identifizierung des Prozesses, der sie erstellt hat genutzt.

Variable bindings

Die *variable bindings* (oder kurz *varbinds*) sind die eigentlichen/wirklichen Daten-Payloads einer jeden SNMP Request- oder Response-Nachricht. Alle Management-Daten, die eine SNMP-Nachricht transportiert, werden in der *varbinds list* gespeichert. Jede SNMP Nachricht, mit Ausnahme der Trap-Nachricht, beinhalten mindestens eine *varbind*. Die meisten SNMP-Nachrichten wären nutzlos, wenn sie keine Management-Daten transportieren würden.

Die *varbinds* erhielten ihren Namen, da sie eine OID (oder eine Identität) mit einem Wert binden

SNMP garantiert, daß die Reihenfolge der *varbinds* in einer GetResponse-Nachricht die selbe bleibt, wie die der vorher eingegangenen Request-Nachricht. In welcher Ordnung sie die *varbinds* in einer Request-Nachricht anordnen, hat keinerlei Bedeutung für den Agent

SNMPv2

SNMPv2 wurde fünf Jahre nach der Einführung von SNMPv1 im Jahre 1993 standardisiert.

SNMPv2 ist eine Weiterentwicklung von SNMPv1.

Es enthält Definitionen zu einer neuen MIB (MIB II),sowie Verbesserungen auf den Gebieten

-der Performance

-der Manager-zu-Manager- Kommunikation

-der Sicherheit

- der Verschlüsselung

Zwei neue Nachrichtentypen

Die wichtigsten Neuerungen von SNMPv2 sind zwei neue Nachrichtentypen:

getBulk und inform

getBulk wird für das Abfragen von großen Datenmengen eines Agenten verwendet,während inform verwendet wird,um eingegangene traps an weitere Netzwerkmanagementsysteme weiterzusenden.

Der getBulk ist ähnlich dem getNext-Message.Er dient zur Minimierung des Netzwerktransfers.Im Wesentlichen dient er dem effizienten Auslesen von Tabellen auf einen Schlag.

Der inform-Message dient der Manager-zu-manager-Kommunikation.

Auf einen inform-Request wird mit einem Response geantwortet.

Dies ist bei einem Trap-message nicht der Fall.Werden zu viele inform-Request gesendet,steigt die Netzlast,so dass die Anzahl der Inform-Requests klein gehalten werden sollte.

Manager-zu-Manager-Beziehungen

SNMPv2 erlaubt einem Prozess das Auftreten sowohl als Manager als auch als Agent.

Wenn ein Manager mittels SNMPv2-Protokoll eine Anfrage an einen Agenten startet,kann dieser die Operation entweder selbst durchführen oder den Request mit dem SNMPv2-Protokoll an weitere Agenten weiterreichen.

4.Praxis

Hier wird ein Überblick über die Verwendung der SNMP-Befehle in einem Netzwerk gemacht.

Für die praktische Umsetzung der SNMP-Befehle sind Kenntnisse über die MIB-Objekte der Standard-MIB notwendig.Hier werden einige wichtige Grundbefehle vorgestellt

Die Gruppe System

Diese Gruppe bietet Informationen über im Netzwerk verwalteten Netzwerkknoten.Diese Gruppe muss von jedem Agenten unterstützt werden.

Für die Benennung der MIB-Objekte wird der Anfangszeichner des Gruppennames verwendet.z.B. steht das sys des Objektes sysDescr für system.Dies wird durchgehend bei der Namensgebung der Objekte in den Gruppen angewendet.

Die Gruppe interfaces

Die interfaces-Gruppe liefert Informationen zur Anzahl der Schnittstellen,die namen der Schnittstellen,die Interface-Typen,die MAC-Adressen,Anzahl der ein/ausgegangenen Pakete,Anzahl der verworfenen Pakete und Anzahl der Paketfehler.Mit dem Kürzel if

Werden alle zu interfaces-Gruppe gehörenden MIB-Objekte benannt.So bedeutet z.B. ifTabelle-InterfaceTabelle.

Abfragen von Informationen aus der MIB

Jedem Object-Identifier,der nicht mit einem „. “ beginnt,setzen die snmp-Befehle automatisch den String „.iso.org.dod.internet.mgmt.mib “ voran.

Befehl	Zweck
snmpget	Abfragen von einzelnen Variablen aus der MIB
snmpgetnext	Abfragen der nachfolgenden Variablen aus der MIB
Snmpset	Setzen von einzelnen Variablen in der MIB
Snmpwalk	Abgragen von ganzen Gruppen auf einmal

snmpget
<versionnummer(v1|v2)><c><hostname><community><objektidentifier>

snmpgetnext
<versionnummer(v1|v2)><c><hostname><community><objektidentifier>

snmpwalk<versionnummer(v1|v2)><c><hostname><community><objektidenti
fier>

snmpset
<versionnumber(v1|v2)><c><hostname><community><object-ID><type>

<value>

Mit dem Parameter <type> wird der Typ von <value> angegeben.mögliche Einträge für <type> sind:

i: Integer; s:String; x:Hex String; d:decimal String; n:NullOBJ

o:ObjID; t:TimeTicks; a:ipAddress

5.Literatur

Memet Edemen „Simple Network Management Protokoll “(Diplomarbeit)

Marshall T.Rose,Keith McCloghrie
„How to Manage your Network Using SNMP“

“Projekt von Jürgen Klein SNMP “
unter http://www.jklein.de/technikerseite.html