

# Werkzeuge des NetSNMP-Paketes

Stephan Knabe

Hochschule Harz, Wernigerode

[stephan.knabe@desy.de](mailto:stephan.knabe@desy.de)

6. März 2004

# Projektüberblick

- Ehemaliges Projekt der University of California, Davis
- Weiterentwicklung als Open Source Projekt  
(ucdsnmp -> NetSNMP)
- Projekthomepage [www.net-snmp.org](http://www.net-snmp.org)
- Starke Verbreitung auch im kommerziellen Umfeld
- Enthält komplette SNMP Programm Suite

# Installation (1)

- Versionen vor 4.9 haben Sicherheitsprobleme
- Viele Distributionen liefern ältere Versionen
- Teilweise fehlen Features (Crypto, dlmod)
- Aktuelle Version ist 5.1
- Quellen und Binärpakete über Projekthomepage erhältlich

## Installation (2)

- Bei Installation zu Entwicklungszwecken ist Installation aus Sourcen empfehlenswert
- Unter Linux ist diese meist problemlos
- Requirements hängen von geplantem Umfang ab  
z.B. OpenSSL, TCP-Wrapper, Perl

# Konfigurationsdateien

- `snmp.conf` und `snmpd.conf`
- Mehrere Speicherorte möglich
- Abhängig von Distribution und Konfiguration
- Überladen der Default-Einstellungen ist möglich
- Persönliche Konfiguration  
    `~/.snmp/snmp.conf`

# snmptranslate

- Übersetzt zwischen OID-Notationen
- Arbeitet lokal
- Gut geeignet für erste Tests

# Einfache Agentenkonfiguration

- Eintrag in `/etc/hosts.allow` für `tcpwrapper`
- Konfiguration über `snmpd.conf`:  
`rocommunity`, `rwcommunity` für  
Einfache Access Kontrolle
- Wichtige Parameter:  
`-L`, `-s`, `-f`, `-D`, `-c`, `-C`, `-u`

# Basic Tools (1)

- Ähnliche Parameter für alle Tools.
- -v - Version (1,2c,3)
- -c - Community
- Hostname
- OID



## Basic Tools (2)

- `snmpget`
- `snmpgetnext`
- `snmpwalk`
- `snmptable`

## Basic Tools (3)

### snmpset

- Nur bei Schreibfreigabe
- Erfordert Parameter zum Variablentyp

# Einbindung fremder MIB's (1)

- NetSNMP bringt bereits wichtige MIB's
- Download von Hersteller-MIB's (z.B. CISCO) möglich
- Directory muss bekannt sein (ggf. nach Default-Verzeichnis kopieren)

## Einbindung fremder MIB's (2)

- Tools müssen zusätzlich MIB laden
- Laden über Umgebungsvariable, Parameter oder Configfile:

`$MIBDIRS, -M oder mibdirs`

`$MIBS, -m, mibs oder mibfile`

## Traps (1)

- `snmpd` Kann Traps versenden
- `snmptrap` - Kommandozeilen-Tool zum Trapversand
- `snmptrapd` - Demon zum Traphandling
- `snmpinform` - Tool zum Versand von Notifications

## Traps (2)

### Versand durch SNMP-Agent:

- `trapcommunity` - Community String für Trapversand
- `trapsink Host [Community [Port]]`  
für v1-Traps
- `trap2sink Host [Community [Port]]`  
für v2-Traps
- `informsink Host [Community [Port]]`  
für Notifications

## Traps (3)

### snmptrapd

- Logging eingehender (default an UDP:162)  
Traps
- Anbindung an syslogd möglich
- Unterstützung verschiedener  
Ausgabeformate

## Traps (4)

### snmptrapd.conf

- `traphandle` - Installation von Traphandlern (Ausführung externer Programme), Trap-Identifikation über OID oder default
- `forward` - Weiterleitung von Traps



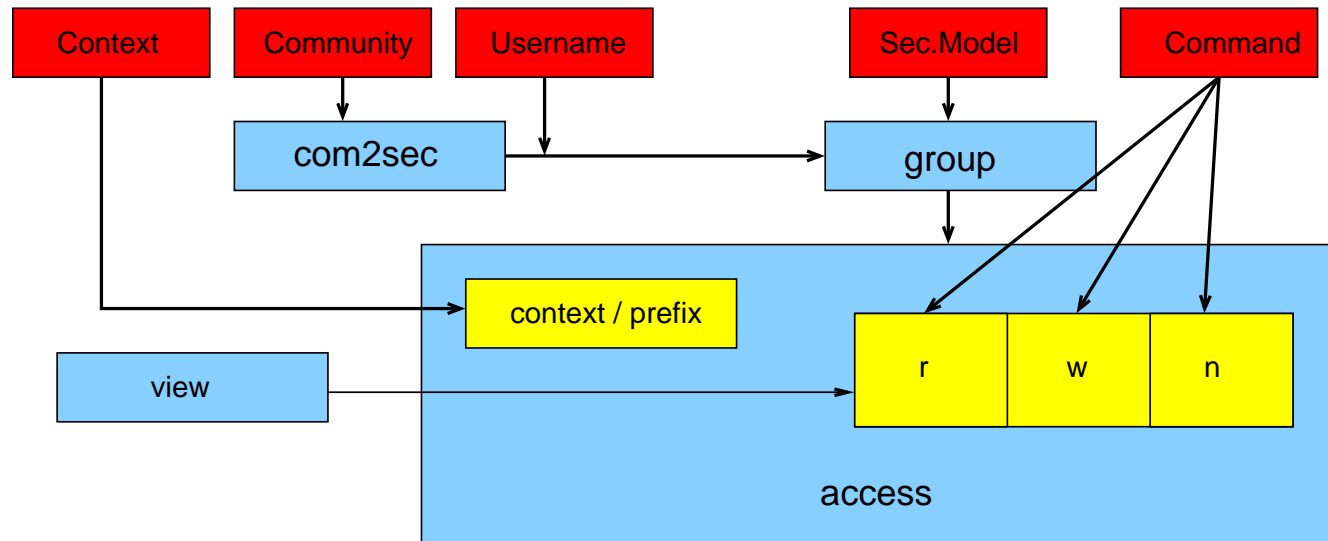
## Traps (5)

### snmptrap

- Versand von Traps
- Übermittlung von varbinds möglich

# VACM (1)

## Detaillierte Rechtezuweisung im MIB-Tree



## VACM (2)

com2sec

Mapping von Community Strings auf  
Security-Namen.

Direktive:

```
com2sec SecName Source Community
```

## VACM (3)

### group

Assoziation von Security-Name und Security-Modell (v1, v2c, usm)

Direktive:

```
group Gruppenname Sec.Modell Security-Name
```

## VACM (4)

### view

#### Definition von OID-Bereichen

- Direktive:

`view Name Mapping-Typ OID [Maske]`

- Mapping-Typ: `included`, `excluded`

- Maske: Hex-Oktett, maskiert festgelegte Knoten

## VACM (5)

### access

#### Definition von Zugriffsrechten

- Direktive:

```
access Name Context Sec.Modell \  
Sec.Level Prefix R W Notify
```

- Sec.Modell: v1, v2c, usm, any

## VACM (6)

### access

- Sec.Level: noAuth (v1,2), auth, priv, AuthPriv
- R,W,Notify: View-Name, none, all
- Context, Prefix, v.a. bei Subagenten

# USM (1)

## Sicherheitsziele

- Authentikation - Keine Modifikation der Nachricht und Identität des Senders
- Privacy Verschlüsselung
- Timeliness - Replay-Protection

Parameter sind Username, Schlüssel und ggf. EngineID, EngineBoot und EngineTime



## USM (2)

### Agentenkonfiguration in Konfigurationsdatei

```
createUser [-e ENGINEID] username \  
AUTHALG AUTHPHRASE [PRIVALG] [PRIVPHRASE]
```

- ENGINEID - wird per Default automatisch generiert
- AUTHALG - Authentisierungsalgorithmus (MD5, SHA)
- PRIVALG - Verschlüsselungsalgorithmus (DES)
- AUTHPHRASE, PRIVPHRASE - Passphrasen

Klartexteintrag in persistenter Datei wird beim Start durch verschlüsselten Eintrag ersetzt.

# USM Kommando-Parameter (v3)

- -u Username
- -l Security-Level
- -a Authentisierungsprotokoll (MD5, SHA)
- -A Passphrase Authentisierung
- -x Verschlüsselungsprotokoll (DES, AES)
- -X Passphrase Verschlüsselung

# Management Tools

- `snmpset` für USM- und VACM-Einträge in Tabellen (z.B. `usmUserTable`)
- `snmpusm` zum Anlegen, Klonen und Manipulieren von Usereinträgen
- `snmpvacm` zum Anlegen und Bearbeiten von VACM-Einträgen

# NetSNMP/UCD MIB (1)

- Ergänzung zu Standard MIB's (z.B. MIBII)
- Erweitert Agenten für verschiedene Managementaufgaben
- Variablen in verschiedenen Tabellen
- Konfiguration vor allem über `snmpd.conf`

# NetSNMP/UCD MIB (2)

- Basis-OID .1.3.6.1.4.1.2021  
(UCD-SNMP-MIB::ucdavis)
- Prozesstabelle (.2)
- Speicher-Auslastung (.4)
- Externe Programme (.8)
- Laufwerke (.9)
- Systemstatistiken (.11)
- Dateigrößen (.15)
- ...

# NetSNMP/UCD MIB (3)

## Konfigurationsdirektiven

- `proc NAME [MAX [MIN]]`  
Überwacht Anzahl von Einträgen in  
Prozesstabelle
- `procfix NAME PROG ARGS`  
Registrierung von „Notfallprogrammen“

# NetSNMP/UCD MIB (3)

## Konfigurationsdirektiven

- `disk PATH [MINSPACE | MINPERCENT%]`

Überwachung des Speicherplatzes von Laufwerken (kB oder %)

- `includeAllDisks MINPERCENT`

Automatische Einbeziehung aller Laufwerke

# NetSNMP/UCD MIB (4)

## Konfigurationsdirektiven

- `load [MAX1 [MAX5 [MAX15]]]`

Monitoring der Systemauslastung

(loadaverage) und Festsetzung von Limits

- `file FILE [MAXSIZE]`

Überwachung von Dateigrößen in kB



# NetSNMP/UCD MIB (5)

## Konfigurationsdirektiven

- `exec NAME PROG ARGS`  
Führt externes Programm aus und gibt einzeliligen Output aus
- `exec MIBNUM NAME PROG ARGS`  
gibt mehrzeiligen Output in Tabellenzeilen aus
- `execfix NAME PROG ARGS`  
Registrierung von „Notfallprogrammen“

# Zusätzliche Tools

- `snmpconfig` - Dialoggesteuerte Konfiguration
- `snmpdelta` - Kontinuierliches Abfragen von Variablen
- `snmpdf` - SNMP-Version von `df`
- `snmpnetstat` - SNMP-Abfragen ähnlich `netstat`
- `snmpstatus` - Polling ausgewählter Netzwerkinformationen

# Anbindung externer Datenquellen

- SNMP-Proxy - Modifikation und Weiterleitung von Requests an andere SNMP-Agenten
- Pass-Through - Delegation an externe Programme
- smux/AgentX - Netzwerkprotokolle zur Kommunikation mit Subagenten

# Prinzip von Subagenten

