CARNet

HRVATSKA AKADEMSKA I ISTRAŽIVAČKA MREŽA
CROATIAN ACADEMIC AND RESEARCH NETWORK

SNMP protokol

NCERT-PUBDOC-2010-09-313

Nacionalni CERT

u suradnji s

LSS

Sigurnosni problemi u računalnim programima i operativnim sustavima područje je na kojem Nacionalni CERT kontinuirano radi.

Rezultat toga rada je i ovaj dokument, koji je nastao suradnjom Nacionalnog CERT–a i LS&S–a, a za koji se nadamo se da će Vam koristiti u poboljšanju sigurnosti Vašeg sustava.

Nacionalni CERT, www.cert.hr

Nacionalno središte za **sigurnost računalnih mreža** i sustava.

LS&S, www.LSS.hr

Laboratorij za sustave i signale pri Zavodu za elektroničke sustave i obradbu informacija Fakulteta elektrotehnike i računarstva Sveučilišta u Zagrebu.

Sadržaj

1. Uvod

Upravljanje i nadzor moderne mrežne okoline izazovan je zadatak ponajprije zbog kompleksnosti i raznolikosti današnjih mrežnih arhitektura. Standardizacija strategije i tehnika upravljanja stoga je nužna kako bi se omogućilo uspješno nadziranje današnjih mreža. Većina arhitektura za upravljanje mrežom temeljena je na istim principima. Arhitektura tipičnog sustava za upravljanje mrežom (eng. NMS – *Network Management System*) sastoji od entiteta za upravljanje, entiteta kojima se upravlja i skupa veza između njih. Entiteti kojima se upravlja često se nazivaju i krajnje točke. Oni su obično računala, poslužitelji i drugi mrežni uređaji (usmjerivači, upravljivi preklopnici, vatrozidovi itd.) koji izvršavaju programe, tzv. agente, koji im omogućuju slanje obavijesti prilikom detekcije nekog problema (primjerice ako je zauzeće diskovnog prostora prešlo definiranu kritičnu razinu). Kada entitet za upravljanje ili više njih prime dojavu o problemu oni reagiraju tako da izvedu jednu ili više akcija ovisno o postavkama sustava za upravljanje mrežom. Entiteti za upravljanje pored primanja dojave mogu dohvaćati određene podatke s nadziranih sustava. Dohvaćanje može biti automatsko ili inicirano od strane korisnika. Programski agenti koji se nalaze na sustavima koji se nadziru predstavljaju poveznicu između fizičkog sustava i parametara koje je potrebno nadzirati te samog sustava za nadzor. Agenti prilikom prvog pokretanja stvaraju bazu podataka o parametrima koji se nadzirati na sustavu, pohranjuju je u specijaliziranom obliku i po potrebi šalju potrebne podatke sustavu za nadzor putem protokola za upravljanje mrežom.

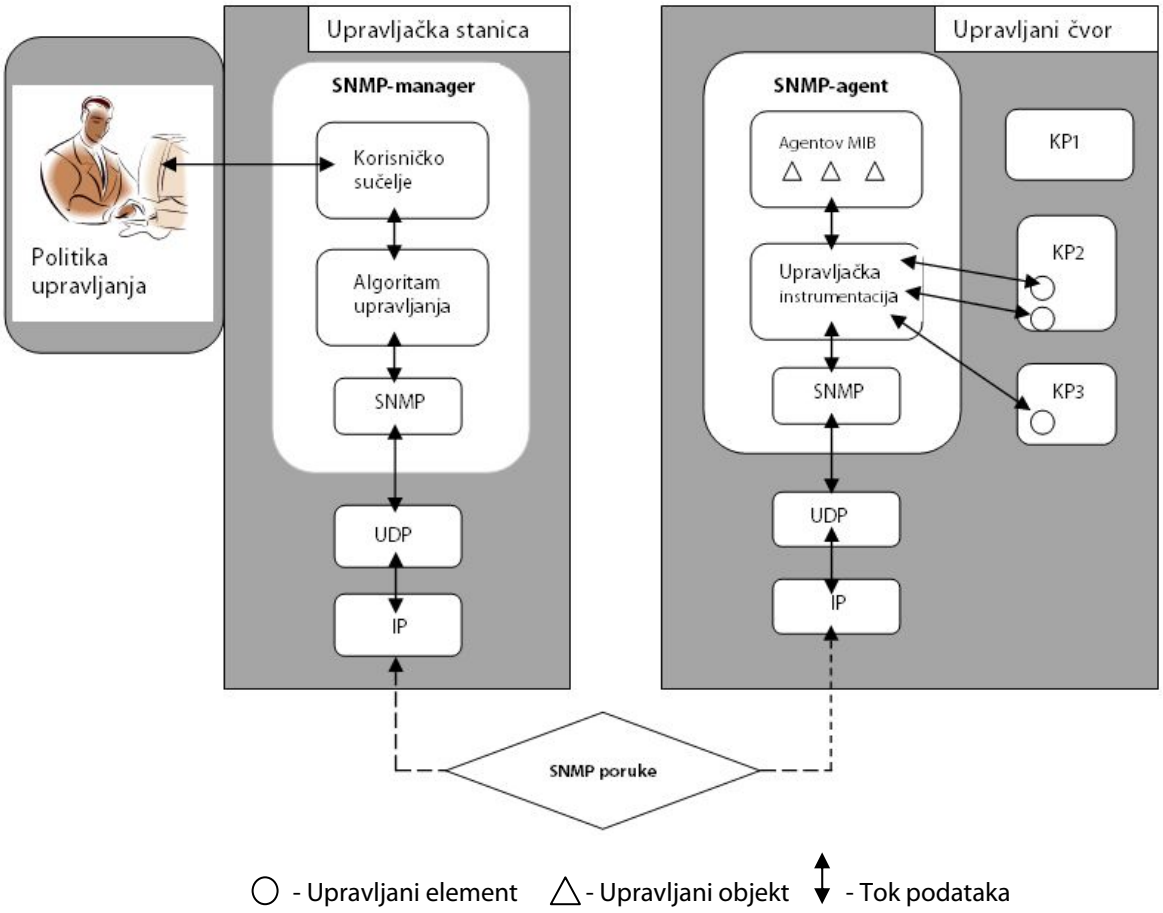
Jedan od najrasprostranjenijih protokola za upravljanje mrežom je SNMP (eng. *Simple Network Management Protocol*). SNMP je mrežni upravljački protokol dizajniran tako da olakša upravljanje i nadzor kompletne mreže te svih njenih entiteta. Funkcionalnost i implementacija SNMP protokola je relativno jednostavna no ipak dovoljno fleksibilna da pruži mogućnost kvalitetnog upravljanja velikim brojem različitih tipova uređaja u današnjoj distribuiranoj mrežnoj okolini.

U dokumentu će biti objašnjena arhitektura SNMP protokola, njegov povijesni razvoj i budućnost. Sagledat će se i neke od popularnih programskih implementacija SNMP upravljačke jedinice na tržištu te će biti dan pregled sigurnosti SNMP protokola.

2. Osnovni koncepti SNMP protokola

Za potrebe upravljanja aktivnim mrežnim uređajima u računalnim mrežama razvijen je aplikacijski protokol **SNMP (***Simple Network Management Protocol).* Njegova funkcija je prikupljanje i organiziranje primljenih informacija o stanju računalne mreže. SNMP protokol mrežnom administratoru omogućuje nadgledanje performansi te pronalaženje i rješavanje mrežnih problema. Protokol SNMP je dio **sustava za upravljanje mrežom – NMS** (eng. *network management system NMS*).

NMS je sastavljen od jedne ili više upravljačkih stanica na kojima se izvode upravljačke aplikacije te od nekolicine upravljanih čvorova na kojima se izvode upravljački agenti kao što je prikazano na slici 1.



Slika 1. Sustav za upravljanje mrežom u okviru SNMP protokola
Izvor: Douglas R. Mauro, Kevin James Schmidt – Essential SNMP

Cijeli sustav funkcionira pomoću niza upravljačkih naredbi. Upravljanje računalnom mrežom može se definirati kao usluga koja se dodaje u postojeću računalnu mrežu da bi se olakšalo upravljanje pojedinim dijelovima mrežnog sustava i mrežom kao cjelinom na jednom od slijedećih područja:

- upravljanje greškama, tj. otkrivanje i dojava grešaka u sustavu,
- upravljanje konfiguracijom,
- upravljanje performansama,
- upravljanje sigurnošću,
- upravljanje uslugama i
- upravljanje obračunavanjem troškova.

Upravljačka stanica je računalo s primarnom namjenom za komunikaciju putem računalne mreže. Računalna mreža može biti organizirana lokalno kao LAN mreža ili globalna Internet mreža. Procesna sposobnost

upravljačke stanice je dovoljna za izvođenje upravljačkih aplikacija. Upravljačka aplikacija **(**često se naziva i SNMP *manager)* je računalni program koji nadgleda ili upravlja elementima na upravljanim čvorovima mreže u skladu s **politikom upravljanja** koja je određena od strane upravitelja računalne mreže (najčešće mrežnog administratora). Upravljani čvor (eng. *Managed node*) je mrežni uređaj čijim se stanjima upravlja ili ih se samo nadgleda. Ovisno o stanju mrežnog uređaja upravljačka stanica može s njim izvesti neku akciju na mreži ili joj to može biti onemogućeno. Prema složenosti i funkciji, upravljani čvorovi mogu biti vrlo raznorodni. Na primjer, to mogu biti razne vrste računala na mreži, mrežni poslužitelji, usmjerivači, modemi ili mrežni pisači. Upravljački agent je procesni entitet (program ili dio programa) koji se izvodi na upravljanom čvoru i sadrži potrebnu instrumentaciju kojom upravlja funkcijama kontroliranih elemenata u čvoru. Upravljačka instrumentacija dirigira komunikacijom s upravljanim elementima (eng. *managed elements*), tj. njihovim podatkovnim strukturama. S druge strane, ona predstavlja te podatkovne strukture kao skup upravljanih objekata. U daljnjem tekstu, za upravljački agent koristit će se uobičajeni naziv - SNMP-agent. Upravljačke informacije govore o stanjima upravljanih elemenata u upravljanom čvoru. Dohvatom tih informacija SNMP *manager* nadgleda stanja upravljanih elemenata, dok postavljanjem njihovih vrijednosti mijenja ta stanja. Upravljačke informacije, koje su fizički smještene u SNMP agentima, SNMP *manageri* vide kao skup **upravljanih objekata** smještenih u jednom virtualnom spremištu informacija koje se naziva **baza upravljačkih informacija - MIB** (eng. *Management Information Base*).

MIB datoteka sadrži definiciju skupa objekata upravljanih SNMP-om. Detaljnije informacije o MIB-u bit će dane u poglavlju o arhitekturi SNMP NMS-a.

Prijenos upravljačkih informacija između SNMP-agenata i SNMP *managera* obavlja se SNMP protokolom. Navedeni odnosi između osnovnih dijelova upravljačkog sustava zasnovanog na protokolu SNMP prikazani su shemom na slici 1.

3. Povijesni razvoj SNMP protokola

U travnju 1988. objavljen je RFC 1052 (*Request For Comments*) - skup dokumenata koji sadrže Internet protokole i diskusije. Taj RFC je specifikacija za standardizirano mrežno upravljanje u kojoj su objašnjeni zahtjevi za mrežno upravljanje.

RFC dokumenti koji su inicijalno opisivali SNMP protokol (objavljeni 1988. godine) su:

1. RFC 1065 – Struktura i identifikacija upravljačkih informacija za TCP/IP,
2. RFC 1066 – Baza upravljačkih informacija za mrežno upravljanje i
3. RFC 1067 - Simple Network Management Protocol.

Sedamdesetih godina i u prvoj polovici osamdesetih godina prošlog stoljeća u mrežama koje koriste TCP/IP protokole nisu bili implementirani protokoli upravljanja mrežnom opremom. Za potrebe upravljanja bio je korišten protokol ICMP (eng. *Internet Control Message Protocol*). ICMP protokol omogućava prijenos upravljačkih poruka između računala i upravljanih mrežnih uređaja (druga računala, usmjerivači i dr.). Koristeći ICMP i različita zaglavlja IP paketa moguće je razviti jednostavne i moćne alate za upravljanje mrežom (PING, i sl.), no čak ni ti alati ne pružaju dovoljno učinkovitu funkcionalnost za upravljanje složenim mrežama. Stoga je 1987. godine razvijen protokol SGMP (eng. *Simple Gateway Monitoring protocol*), namijenjen nadzoru usmjerivača. Rastući zahtjevi i brz razvoj tada već složenih TCP/IP mreža, otežavali su mrežno upravljanje. To sve je uvjetovalo daljnji razvoj i poboljšanje protokola SGMP te je time nastao protokol SNMP. Kroz povijest od 1988. godine do danas razvijene su tri inačice protokola SNMP.

3.1. Inačice SNMP protokola

Verzije SNMP protokola koje su danas u upotrebi:

- **SNMPv1** - u upotrebi od 1988. godine -> RFC 1157
- **SNMPv2** - u upotrebi od 1995. godine -> RFC 1901 – 1908
- **SNMPv3** - u upotrebi od 1998. godine -> RFC 2271 – 2275

3.1.1. SNMPv1

SNMPv1 protokol je prihvaćen kao standard u TCP/IP mrežama od 1988. godine. Još i danas se dosta koristi unatoč poznatim sigurnosnim nedostacima. Sigurnost se kod SNMPv1 temelji na korištenju takozvanih zajedničkih znakovnih nizova (eng. *community string*). *Community string* je u stvari niz tekstualnih ASCII znakova i podsjeća na tradicionalne lozinke koje se koriste u operacijskim sustavima. Koristi se za autentikaciju SNMP poruka između upravljačke jedinice i upravljanog uređaja. Najveći problem je što se ne koristi nikakav oblik enkripcije pa neovlašteni korisnici mogu snimanjem IP paketa koji se prenose mrežom pročitati sadržaj SNMP poruka, a samim time i *community string*. Poznavajući taj podatak, zlonamjerni korisnici mogu pristupiti upravljačkim informacijama nekog mrežnog uređaja i promijeniti njegovu konfiguraciju.

Inačica 1.0, objavljena u svibnju 1991. godine, obuhvaćala je sljedeće RFC dokumente :

- RFC 1155 - struktura i identifikacija upravljačkih informacija za TCP/IP te struktura i identifikacija upravljačkih informacija za objekte,
- RFC 1212 - MIB definicije,
- RFC 1213 -baza upravljačkih informacija za mrežno upravljanje MIB-2 i
- RFC 1157 (*Simple Network Management Protocol*) - SNMP protokol, definira: poruke koje se mogu razmjenjivati između upravljačkih entiteta i upravljačkih stanica (poruke omogućavaju čitanje i obnavljanje vrijednosti), alarm poruke (trap), format poruka i komunikacijski protokol.

Druga inačica, SNMPv2, donijela je određena poboljšanja u odnosu na prvu, ali su problemi glede sigurnosti ostali i dalje prisutni.

3.1.2. SNMPv2

U travnju 1993. godine inačica 2.0 postala je standard. Ta inačica nudi dodatne mogućnosti kao što su sigurnost i autentikacija. SNMP inačica 2 je dokumentirana u nekoliko RFC dokumenata:

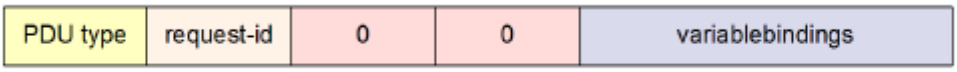
1. RFC 1902 - MIB struktura,
2. RFC 1903 - tekstualne konvencije (promjene i novosti u SNMP inačici 2),
3. RFC 1904 - izjave sukladnosti s SNMPv1 (kooperativnost SNMP inačica 1 i 2),
4. RFC 1905 - protokol operacija,
5. RFC 1906 - transport, mapiranje i
6. RFC 1907 - MIB.

SNMPv2 predstavlja proširenje protokola SNMPv1 te podržava tri načina pristupa upravljačkoj informaciji:

1. **upravljač-agent zahtjev-odgovor:** SNMPv2 upravljač šalje zahtjev agentu, a agent odgovara slanjem traženih upravljačkih informacija. Koristi se za dohvaćanje i modificiranje upravljačkih informacija;
2. **upravljač-upravljač zahtjev-odgovor:** jedan SNMPv2 upravljač šalje zahtjev drugom upravljaču, a drugi odgovara slanjem traženih upravljačkih informacija;
3. **agent-upravljač bez potvrde:** SNMPv2 agent šalje poruku „*Trap*“ upravljaču.

3.1.3. Razlike SNMPv1 i SNMPv2

SNMPv1 podržava prvi i treći način pristupa upravljačkim informacijama. Samo je drugi način specifičan za SNMPv2. Komunikaciju između upravljača omogućava mehanizam informiranja. Porukom *Inform* jedan upravljač obavještava drugog o upravljačkoj informaciji koju posjeduje. Operaciju *Inform* pokreće upravljač pošiljatelj slanjem *InformRequest* PDU-a (eng. *Packet Data Unit*) upravljaču primatelju. Na slici 2. prikazan je format poruke *InformRequest* PDU-a. Detaljnije o formatu i samom PDU-u kasnije u poglavlju **SNMP PDU**. Upravljač primatelj potvrđuje prijem PDU-a slanjem *Response* PDU-a upravljaču pošiljatelju. SNMPv2-Trap PDU ima drugačiji format od Trap PDU-a korištenog u SNMPv1. Format SNMPv2-Trap PDU je identičan formatu *GetRequest, GetNextRequest SetRequest i InformRequest* PDU-a korištenih u SNMPv2. Poruke *GetRequest, GetNextRequest i SetRequest* su zadržale isti format kao u prvoj inačici protokola SNMP, no SNMPv2 predviđa i uvođenje *Report* PDU-a u upotrebu, ali njegov točan način korištenja i semantika još nisu u potpunosti definirani. Suradnja i postojanje SNMPv1 i SNMPv2 unutar jednog NMS-a je moguća te za njihovu suradnju postoje jasna pravila unutar dvije kategorije: upravljačka informacija i protokol.

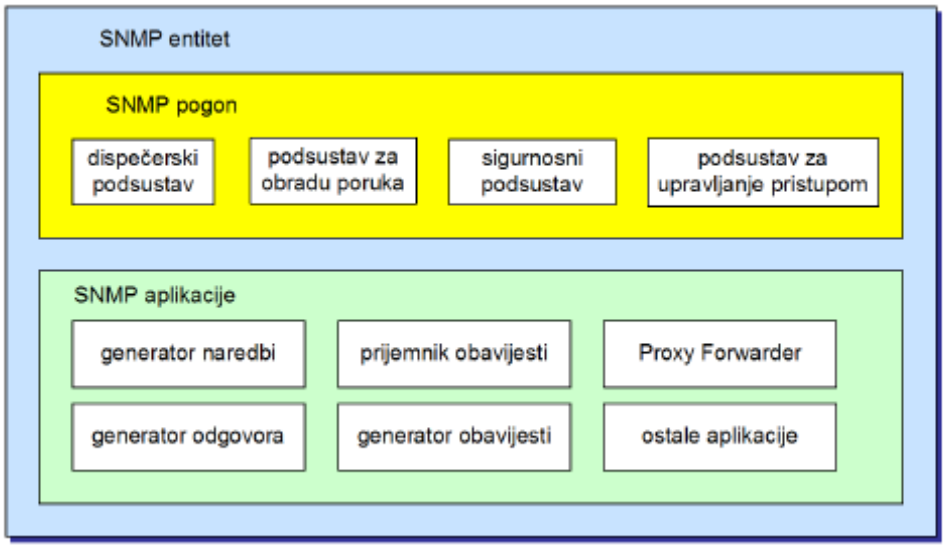


Slika2. SNMPv2 Trap PDU i InformRequest PDU
Izvor: SNMP protocol wiki

3.1.4. SNMPv3

SNMPv3 posjeduje bitno poboljšane sigurnosne mehanizme. Posebno treba izdvojiti mehanizme za autentikaciju, tj. provjeru vjerodostojnosti korisnika i zaštitno kodiranje SNMP poruka, odnosno enkripciju. SNMPv3 može koristiti takozvanu korisničku (*user-based*) autentikaciju (autentikaciju na temelju korisničkog imena i lozinke) ili se provjera vjerodostojnosti korisnika može obaviti bez slanja lozinki u čitljivom obliku. Takva provjera vjerodostojnosti se temelji na upotrebi algoritama HMAC-MD5 (eng. *Hash-based Message Authentication Code- Message-Digest algorithm 5*) ili HMAC-SHA (eng. *Hash-based Message Authentication Code -Secure Hash Algorithm*). MD5 i njegov nasljednik SHA su algoritmi za provjeru autentičnosti datoteka ili poruke prilikom prijenosa između pošiljaoca i primatelja. Zaštitna enkripcija koristi 56-bitni CBC-DES (eng. *Cipher Block Chaining-Data Encryption Standard*) algoritam za kodiranje i dekodiranje SNMP poruka. Najvažnija promjena u

SNMPv3 je napuštanje koncepta NMS-a koji se temelji na upravljačima i agentima. SNMPv3 NMS čine SNMP entiteti, prikazani na slici 3.



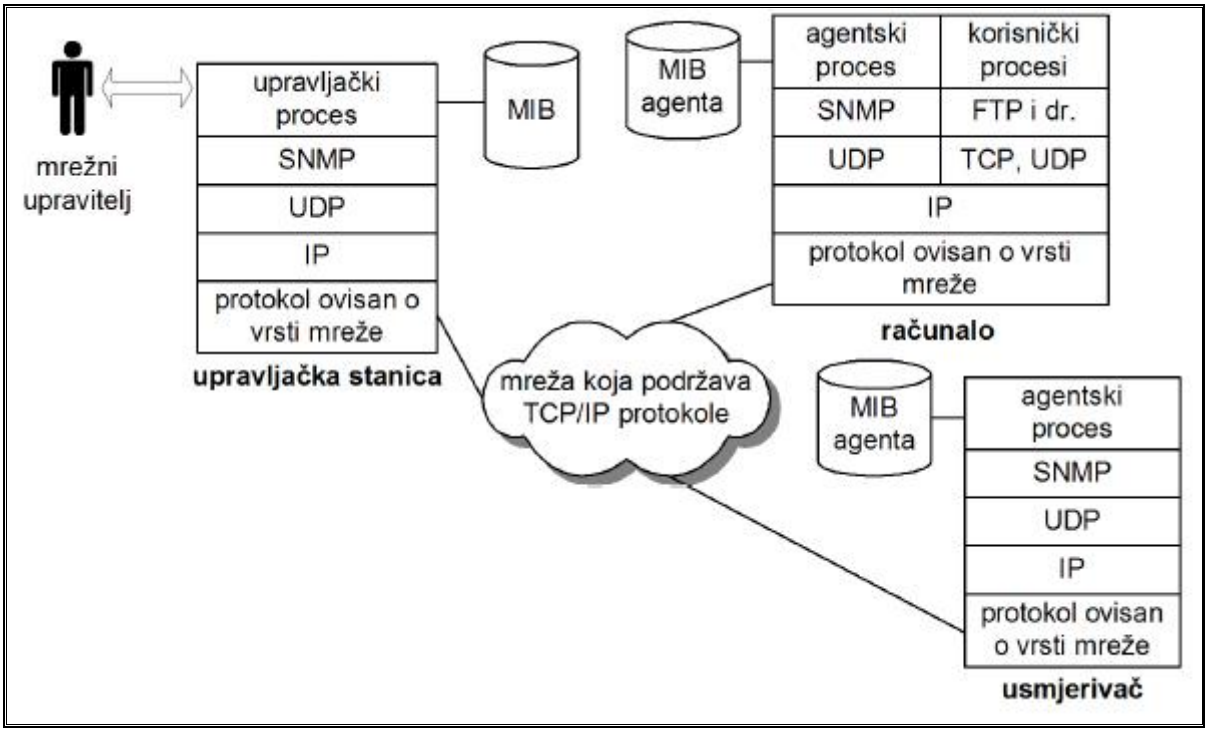
Slika 3. SNMP entitet
Izvor: Douglas R. Mauro, Kevin James Schmidt - Essential SNMP

Novi koncept definira arhitekturu NMS-a, a ne samo skup poruka kao ranije inačice. Svaki se entitet (slika 3) sastoji od SNMP pogona (eng. *SNMP engine*) i SNMP aplikacija. SNMP pogon se sastoji od 4 podsustava. Dispečerski podsustav (eng. *Dispatcher Subsystem*) šalje i prima SNMP poruke (u prijemu određuje inačicu svake primljene poruke). Podsustav za obradu poruka (eng. *Message Processing Susbsystem*) priprema SNMP poruke za slanje drugim entitetima i obrađuje podatke iz SNMP poruka primljenih od drugih entiteta. Sigurnosni podsustav (eng. *Security Subsystem*) pruža usluge autentikacije i zaštite privatnosti upravljačkih informacija (enkripcija). Autentikacija koristi mehanizam zajedničkih znakovnih nizova (eng. *community string*), ako se radi o SNMPv1 ili SNMPv2 porukama, odnosno SNMPv3 korisničku autentikaciju (mehanizmi autentikacije navedeni iznad slike 3). Podsustav za upravljanje pristupom (eng. *Access Control Subsystem*) je odgovoran za upravljanje pristupom objektima MIB-a. Pomoću tog podsustava moguće je upravljati pristupom korisnika pojedinim objektima (kojim objektima korisnik smije pristupiti i koje operacije smije nad pojedinim objektom izvoditi). Drugi dio entiteta su SNMP aplikacije:

- generator naredbi - generira **get, getnext, get-bulk** i **set** zahtjeve, (detaljnije o zahtjevima u poglavlju 4.3.) te obrađuje odgovore (implementira se u upravljačkoj stanici),
- generator odgovora - šalje odgovore na **get, get-next, getbulk i set** zahtjeve (implementira se u upravljanim mrežnim uređajima),
- generator obavijesti - generira SNMP **trapove** i obavijesti,
- prijemnik obavijesti - prima **trap** i **inform** poruke, a **proxy forwarder** olakšava prosljeđivanje SNMP poruka između entiteta.

4. Arhitektura SNMP NMS-a

SNMP je dizajniran kao protokol aplikacijskog sloja OSI modela [10]. Na transportnom sloju SNMP koristi transportnu uslugu protokola UDP (eng. *User Datagram Protocol*). UDP višim protokolnim slojevima pruža bespolnu (eng. *connectionless*) uslugu transporta informacija, jer uređaj koji primi UDP datagram ne potvrđuje prijem pošiljatelju. Na taj se način ubrzava prijenos i smanjuje količina prenesene informacije. U slučaju SNMP-a to je izuzetno važno jer je osnovni cilj NMS-a utemeljenog na protokolu SNMP (SNMP NMS) bio taj da upravljački protokol svojim porukama što manje opterećuje mrežu. Prikaz arhitekture SNMP NMS-a dan je na slici 4.



Slika 4. Arhitektura SNMP NMS-a
Izvor: William Stallings - Data and computer communications

Na upravljačkoj stanici pokrenut je proces koji upravlja pristupom središnjem MIB-u instaliranom u SNMP *manageru,* te pruža sučelje prema mrežnom upravitelju, osobi koja je zadužena za obavljanje poslova vezanih uz upravljanje mrežom (najčešće mrežnom administratoru). Agentski proces interpretira primljene SNMP poruke i upravlja pristupom MIB-u agenta. SNMP poruke su detaljno obrađene u poglavlju 4.3.

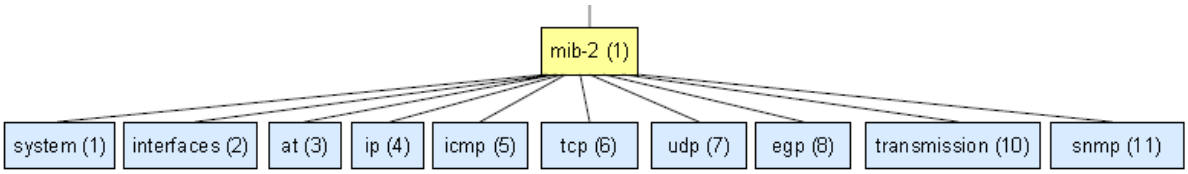
4.1. Baza upravljačkih informacija - MIB

Svaki SNMP agent sadrži popis svih svojih upravljanih objekata koji se naziva baza upravljačkih informacija. Baza sadrži sljedeće zapise:

- ime,
- OID,
- tip podatka,
- dozvole čitanja i pisanja te
- kratki opis za svaki objekt SNMP agenta.

Pomoću informacija o objektu i vrijednosti pojedine instance – varijable, SNMP *manager* može slati SNMP poruke za dohvat ili postavljanje pojedine varijable SNMP agenta.

Objekti namijenjeni upravljanju putem SNMP protokola grupirani su u deset skupina, od kojih svaka odgovara jednom čvoru u SMI stablu (detaljnije u poglavlju 4.6 „SMI standard“). Tih deset čvorova su podstabla čvora *mib-2* (slika 5).



Slika 5 : Čvor mib-2 SMI stabla

Opis podstabla čvora Mib-2

Mib-2 odgovara inačici SNMPv2 i stoga objekt *cmot* čiji je OID jednak 9 nije više prisutan u SMI stablu. CMOT (CMIP over TCP/IP) je protokol koji se 1989. godine razvijao zajedno sa SNMP protokolom, no zbog prevelike složenosti u implementaciji, IAB (*Internet Architecture Board*) je 1989. godine odlučio da se protokoli SNMP i CMOT nastave razvijati odvojeno s ciljem očuvanja jednostavne implementacije protokola SNMP. Spomenutih deset skupina upravljanih objekata predstavlja osnovni sadržaj koji bi svaka upravljačka stanica morala razumjeti. Skupina *system* omogućava upravljaču da odredi ime, lokaciju i opis mrežnog uređaja (naziv proizvođača, sklopovlje i programske pakete koje uređaj sadrži, namjena uređaja i dr.). Skupina *interfaces* odnosi se na mrežna sučelja i priključke na mrežnim uređajima (*ports)*. Skupina *at* pruža informacije o preslikavanju adresa (npr. Ethernet u IP adrese). Skupina *ip* je namijenjena prikupljanju informacija o IP prometu koji ulazi ili izlazi iz mrežnog čvora. Ova je skupina posebno važna za usmjerivače. Skupina *icmp* se odnosi na ICMP poruke o pogreškama u komunikaciji. Za svaku ICMP poruku definirana je posebna varijabla koja sadrži broj primljenih relevantnih ICMP poruka. Skupina *tcp* nadzire broj trenutno aktivnih TCP veza, kao i kumulativni broj TCP veza, broj primljenih i poslanih TCP segmenata te statistiku pogrešaka. Skupina *udp* je zadužena za praćenje broja primljenih i poslanih UDP datagrama i sličnih informacija. Skupina *egp* se primjenjuje u usmjerivačima koji podržavaju protokol EGP (*Exterior Gateway Protocol*). Skupina *transmission* je prazna skupina koja čuva mjesto u stablu za MIB-ove specifične za pojedinu vrstu mreže (npr. *Ethernet*). Skupina *snmp* je namijenjena prikupljanju statistike o djelovanju samog protokola SNMP (broj poslanih SNMP poruka, vrste poruka i sl.).

4.2. ASN.1

Glavna bit modela SNMP NMS-a predstavlja skup objekata kojima upravljaju agenti, a čitaju ih i zapisuju upravljačke stanice. Postoje dva problema koja se javljaju pri komunikaciji mrežne opreme različitih proizvođača. SNMP poruke moraju riješiti te probleme ako se želi postići interoperabilnost, tj. da svi SNMP uređaji (različitih proizvođača) razumiju i znaju interpretirati primljene SNMP poruke.

Prvi problem nastaje zbog toga što različiti programski jezici imaju različite tipove podataka (cjelobrojne, znakove, nizove znakova, oktete itd.). Ukoliko SNMP upravljač pošalje poruku punu vrsta podataka iz jednog programskog jezika (npr. Jave) a SNMP agent je napisan u drugom programskom jeziku (npr. programskom jeziku C), oni se neće razumjeti. Da bi se riješio ovaj problem SNMP koristi ASN.1 (*Abstract Syntax Notation One*) notaciju za definiranje tipova podataka korištenih za konstrukciju SNMP poruke. Budući da je ASN.1 sintaksa nezavisna od izbora programskog jezika, SNMP agenti i upravljači mogu biti pisani u bilo kojem programskom jeziku.

Drugi problem nastaje kada komuniciraju dva krajnja sustava (komunikacija koja se odvija logički izravno između dva sudionika, u ovom slučaju na svakom kraju je jedan sudionik komunikacije), od kojih jedan, na primjer, zapisuje cjelobrojne vrijednosti u obliku 32-bitnih binarnih brojeva i u tehnici dvojnog komplementa, a drugi u obliku 16-bitnih binarnih brojeva u tehnici jednostrukog komplementa. Ni C niti Java ne zadiru u tu problematiku. To je još jedan od razloga za neophodno korištenje jezika za standardiziranu definiciju upravljanih objekata.

Dakle, svi tipovi podataka u SNMP poruci moraju biti ispravni ASN.1 tipovi podataka i moraju biti kodirani u skladu s osnovnim pravilima kodiranja. ASN.1 sintaksa je vrlo moćna i kompleksna ali zato pati od nedostatka učinkovitosti. Glavna snaga ASN.1 sintakse je u definiranju jednoznačnih pravila kodiranja na razini bita. No, to je ujedno i slabost ASN.1 sintakse. Pravila kodiranja su takva da je cilj postići što manje bita na prijenosnom mediju, a to se plaća vrlo slabom učinkovitošću korištenja procesora na

komunikacijskim krajevima prilikom kodiranja i dekodiranja poruka. ASN.1 je definirana standardom ISO 8824, a pravila kodiranja standardom ISO 8825.

Apstraktna sintaksa definira strukturu podataka neovisnu o načinu kodiranja korištenom za prikaz podataka. Kroz apstraktnu sintaksu je moguće definirati tipove podataka i vrijednosti istih. Tip podataka uključuje veći broj vrijednosti, a osnovna podjela je na jednostavni i složeni tip.

Neki od osnovnih tipova podataka za ASN.1 jezik dani su u tablici 1.

Naziv tipa	Kod	Kratki opis
INTEGER	2	Cijeli broj proizvoljne duljine
BIT STRING	3	Niz koji sadrži 0 ili više bita
OCTET STRING	4	Niz koji sadrži 0 ili više okteta bez predznaka (*unsigned*)
NULL	5	*Place holder*
OBJECT IDENTIFIER	6	Tip za označavanje objekata

Tablica 1. Osnovni ASN.1 tipovi podataka

Kodiranjem se dobiva niz okteta koji se koristi za prikaz podatkovnih vrijednosti. Pravila kodiranja definiraju način preslikavanja iz jedne u drugu sintaksu, tj. iz apstraktne sintakse u sintaksu prijenosa (slika 6). Drugim riječima, pravilima kodiranja određen je način na koji će se skup podatkovnih vrijednosti iz apstraktne sintakse prikazati u sintaksi prijenosa.

Slika 6. Kodiranjem iz apstraktne sintakse u sintaksu prijelaza

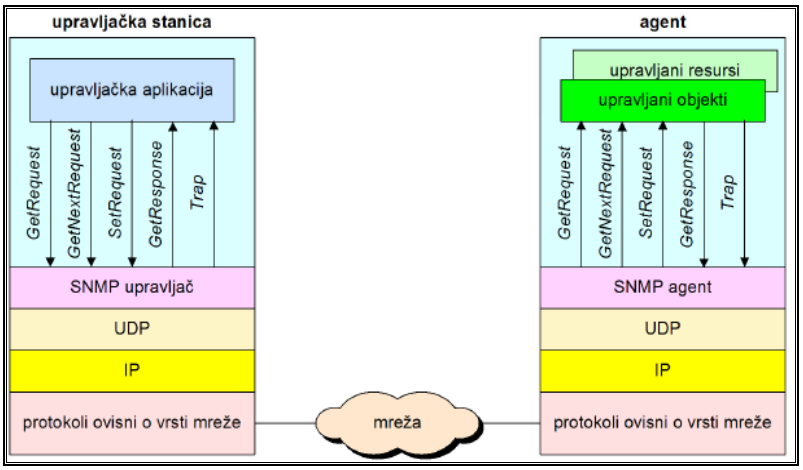
4.3. SNMP poruke

Tri osnovne funkcionalnosti koje protokol SNMP pruža sustavu upravljanja mrežom su mogućnost slanja *get, set* i *trap* poruka.

1. ***Get (dohvati)*** upravljačkoj stanici omogućava dohvaćanje vrijednosti upravljanih objekata sadržanih u MIB-ovima agenata. Slanje *get* poruka agentima naziva se prozivanje (*polling*). Upravljač agente najčešće proziva ciklički. Unutar nekog zadanog vremenskog intervala upravljač prozove redom sve agente i nakon toga započinje novi ciklus prozivanja. Frekvenciju prozivanja je moguće konfigurirati u SNMP *manageru*.
2. ***Set* (postavi)** upravljačkoj stanici omogućava postavljanje vrijednosti upravljanih objekata u MIB-ovima agenata. Aplikacija obično vrši operaciju *set* tako da upravljačkoj stanici preda naziv agenta i jedan ili više OID oznaka zajedno s pripadajućim inačicama te novu vrijednost. Agent prosljeđuje zahtjev i dodjeljuje nove vrijednosti MIB varijabli. Ako dođe do pogreške nova vrijednost neće biti dodijeljena.
3. ***Trap* (privuci pažnju)** omogućava agentu da obavijesti upravljačku stanicu o važnim događajima koji se zbivaju u komunikacijskoj okolini agenta.

Standardom nije određen broj upravljačkih stanica niti omjer broja upravljačkih stanica prema broju agenata u NMS-u. Praksa pokazuje da je u jednom NMS-u poželjno imati barem dvije upravljačke stanice (zbog pouzdanosti sustava), a broj agenata može iznositi i do nekoliko stotina. Na temelju osnovnih

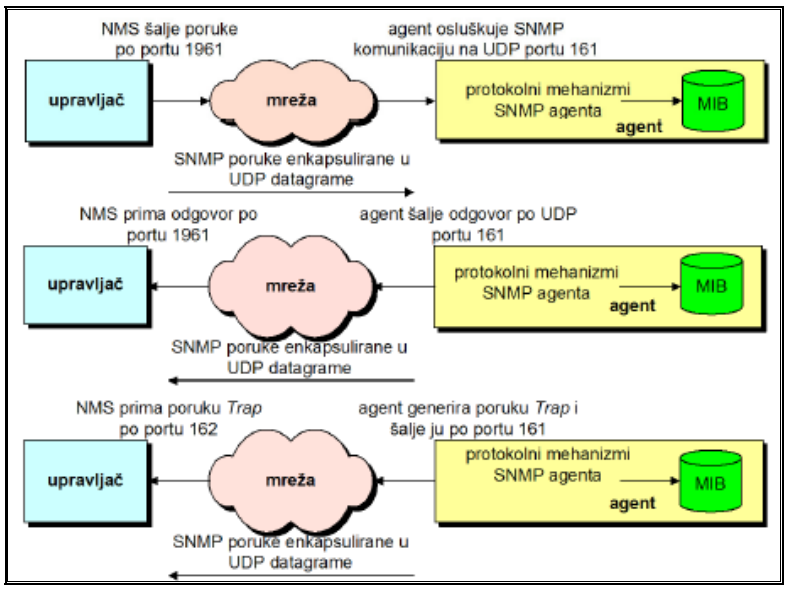
mogućnosti protokola SNMP definirane su SNMP poruke. Upravljačka aplikacija (NMA) šalje agentima poruke *GetRequest, GetNextRequest* i *SetRequest* (slika 7). Primitak bilo koje od tih poruka agent potvrđuje slanjem poruke *GetResponse* upravljačkoj stanici. Poruku *Trap* agent šalje upravljačkoj stanici kao reakciju na događaj koji utječe na sadržaj MIB-a agenta ili na njegove upravljane resurse u podlozi MIB-a. S obzirom da se SNMP oslanja na transportni protokol UDP, on je također bespojni protokol. Drugim riječima, prilikom komunikacije između upravljačke stanice i agenata ne uspostavljaju se veze. Svaka razmjena SNMP poruka predstavlja zasebnu transakciju.



Slika 7. Razmjena SNMP poruka

Izvor: William Stallings, Data and computer communications/ Network Management- SNMP

Sve se SNMP poruke enkapsuliraju se u UDP datagrame. Prilikom slanja SNMP poruke (*get ili set*), upravljač u zaglavlje UDP datagrama upisuje vrijednost izvorišnog porta. U primjeru na slici 8 odabrana je proizvoljna vrijednost 1961. To je tzv. privremeni port kojeg upravljaču dodjeljuje operacijski sustav stanice na kojoj se izvodi NMA. Odredišni port je port 161 na kojem agent osluškuje SNMP komunikaciju i prima UDP datagrame. Kad agent stvara odgovor na primljenu SNMP poruku, u zaglavlje UDP datagrama upisuje u polje izvorišni port vrijednost 161, a u polje odredišni port u ovom slučaju vrijednost 1961. Prilikom slanja poruke *Trap* agent u polje izvorišni port upisuje vrijednost 161 (to je osnovna vrijednost no moguće je korištenje i drugih vrijednosti), a u polje odredišni port vrijednost 162. Upravljač prima poruke *Trap* na portu 162. Maksimalna duljina SNMP poruke je ograničena dozvoljenom duljinom UDP datagrama, i iznosi 65.507 okteta.

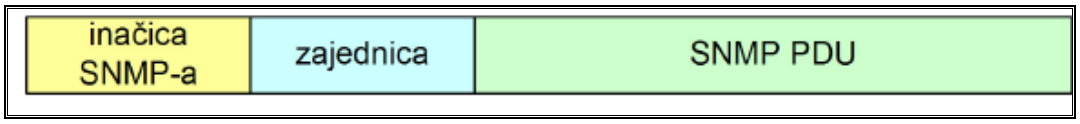


Slika 8. Razmjena SNMP poruka i asinkrono generiranje poruka Trap
Izvor: Douglas R. Mauro, Kevin James Schmidt – Essential SNMP

4.3.1. Format SNMP poruka

U SNMP NMS-u upravljačka informacija se između upravljača i agenata prenosi u obliku SNMP poruka. Svaka SNMP poruka sadrži tri polja:

- inačica protokola SNMP,
- naziv zajednice (zajednički znakovni niz) i
- SNMP PDU.



Slika 9. Format SNMP poruke

Izvor: William Stallings, Data and computer communications

Po nazivima polja vidi se odstupanje od OSI terminologije po kojoj bi se cijela SNMP poruka trebala zvati SNMP PDU (*Packet Data Unit*).

Polje **inačice** sadrži jednu od tri moguće vrijednosti:

- 1 za SNMPv1,
- 2 za SNMPv2, odnosno
- 3 za SNMPv3 poruku.

Zajednički znakovni niz (eng. *community string***)** se koristi za potrebe sigurnosti SNMP komunikacije u upravljačkim sustavima. Polje zajednice je u stvari niz tekstualnih znakova koji podsjeća na tradicionalne lozinke. Najveći problem je u tome što neovlašteni korisnik snimanjem IP paketa koji se prenosi mrežom može pročitati sadržaj SNMP poruke te tako saznati zajednički znakovni niz. Kobna posljedica navedenog je da bilo koji korisnik, ako poznaje *community string*, može pristupiti upravljačkim informacijama nekog mrežnog uređaja i promijeniti mu konfiguraciju.

U svakom agentu se konfiguriraju tri zajednička znakovna niza:

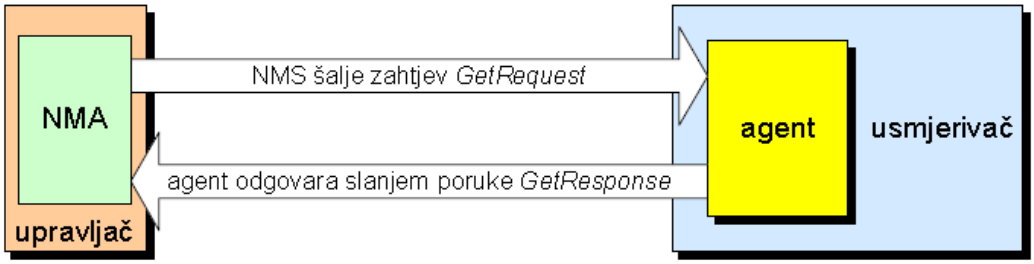
1. *read-only,*
2. *read-write i*
3. *trap.*

Read-only znakovni niz omogućava isključivo čitanje vrijednosti podataka iz MIB-ova. ***Read-write*** niz omogućava čitanje i upisivanje vrijednosti u MIB-ove. ***Trap*** niz omogućava upravljačima prijem poruka *Trap*. Većina proizvođača mrežne opreme isporučuje mrežne uređaje s unaprijed postavljenim *read-only i read-write* zajedničkim nizovima: *public,* odnosno *private*. Na primjer, u polju zajednice u SNMP poruci koja je namijenjena modificiranju određene vrijednosti u MIB-u agenta mora biti upisan tekst *private*. Naravno, administratori bi trebali svakako promijeniti zajedničke nizove prilikom početne instalacije mrežnih uređaja. U NMS-u koji se oslanja na SNMPv1 ili SNMPv2 poželjno je korištenje vatrozida koji će upravljanu mrežu zaštititi od ugrožavanja sigurnosti kroz mehanizme protokola SNMP. Druga mogućnost poboljšanja sigurnosti mreže prilikom korištenja zajedničkih znakovnih nizova su virtualne privatne mreže (*Virtual Private Network* - VPN).

4.4. Operacije SNMP protokola

4.4.1. GET

Upravljač inicira slanje zahtjeva za dohvaćanje vrijednosti varijabli iz baze upravljačkih informacija u agentu u obliku SNMP poruke *GetRequest* (slika 10). Odgovarajući SNMP PDU mora sadržavati OID oznake traženih objekata MIB-a. Svaka varijabla u polju *variablebindings* SNMP PDU-a mora se odnositi na jednu instancu objekta čija se vrijednost želi dohvatiti. Svaki objekt MIB-a označen je u obliku *x.y*, pri čemu je x OID dotičnog objekta, a y je oznaka instance unutar objekta. Agent odgovara slanjem SNMP poruke *GetResponse* i to tako da u *GetResponse* PDU polje upisuje odgovarajuće parove (OID, vrijednost).



Slika 10. Operacija GET protokola SNMP

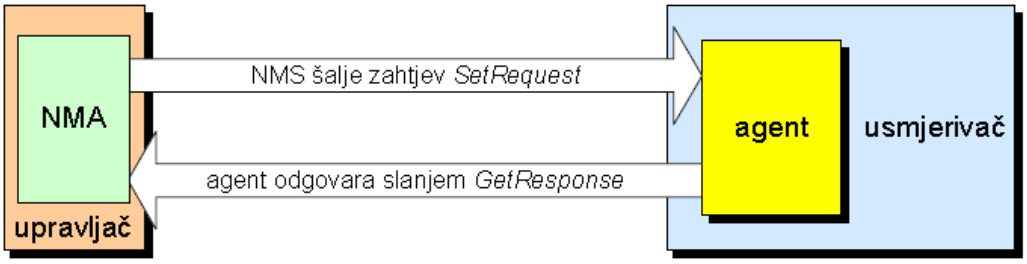
Izvor: William Stallings, Data and computer communications

4.4.2. GET-BULK

Prilikom slanja odgovora (u sklopu operacije *get*) agent može vratiti upravljačkoj aplikaciji vrijednosti više od jednog objekta MIB-a. Međutim, maksimalna duljina SNMP poruke *GetResponse* je ograničena i ovisi o izvedbi programske implementacije agenta. Agent inačice SNMPv1 rezultira porukom o grešci za sve slučajeve u kojima ne može u odgovoru poslati sve tražene vrijednosti. Stoga je u inačicu SNMPv2 ugrađena operacija *get-bulk* koja agentu daje uputu da pošalje onoliko vrijednosti koliko može. Drugim riječima, odgovor agenta smije biti nepotpun.

4.4.3. SET

Operacija *set* služi za postavljanje vrijednosti upravljanih objekata koji se nalaze u MIB-u agenta. Prilikom obavljanja operacije *set* NMA šalje agentu *SetRequest* PDU koji sadrži polje *variablebindings* s parovima *OID-vrijednost*. Jedan *SetRequest* PDU može postaviti vrijednosti jednog ili više objekata odjednom. Agent odgovara na poruku *SetRequest* slanjem poruke *GetResponse*. *GetResponse* PDU sadrži status pogreške (polje *error-status*). Ako je u to polje upisana vrijednost '0', znači da nije bilo problema prilikom provođenja zahtjeva u agentu (tražene vrijednosti su uspješno upisane u MIB). Ako je u polje *error-status* upisana vrijednost različita od nule, tada je u agentu nastupila pogreška prilikom pokušaja upisivanja traženih vrijednosti u MIB. Inačica SNMPv1 je definirala samo pet različitih tipova pogrešaka. Zbog protokolnih proširenja u odnosu na prvu inačicu, SNMPv2 dodaje još 13 tipova pogrešaka.



Slika 11. Operacija SET protokola SNMP
Izvor: William Stallings, Data and computer communications

4.4.4. TRAP

Koristeći *trap* mehanizam agent može obavijestiti upravljačku aplikaciju da je u njegovoj komunikacijskoj okolini nastupio neki neželjeni događaj. Nakon nastupa takvog događaja agent šalje *trap* PDU poruku prema odredištu čija je adresa unaprijed konfigurirana u agentu (to je najčešće IP adresa upravljačke stanice). Upravljačka stanica ne šalje agentu potvrdu o uspješnom primitku *trap* PDU-a, budući da se cijeli SNMP NMS oslanja na komunikaciju UDP protokolom. Unatoč tome što postoji realna mogućnosti njihova gubitka, *trap* poruke predstavljaju sastavni dio svakog ozbiljnijeg NMS-a. Jedna od posebno korisnih primjena je prozivanje agenata na temelju *trap* poruka (*trap-directed polling*). Uobičajeno je da upravljačka stanica po nekom definiranom rasporedu proziva redom sve agente. Prozivanje (*polling*) se ciklički ponavlja, međutim u NMS-u s velikim brojem agenata (pri čemu svaki agent održava veliki broj objekata) takvo prozivanje je nepraktično jer ciklus prozivanja predugo traje, a prikupljena upravljačka informacija ne prezentira dovoljno precizno zbivanja u mreži. U takvom je NMS-u bolje primijeniti sljedeću tehniku:

- pri inicijalizaciji sustava i povremeno (na primjer, jednom dnevno) upravljačka stanica proziva sve agente kako bi prikupila ključne informacije,
- ostalo vrijeme agenti obavještavaju upravljačku stanicu o eventualnom nastupu neželjenih događaja (npr. ispad mrežnog uređaja u kojem je instaliran agent iz rada, pad linka i sl.) pomoću *trap* mehanizma.

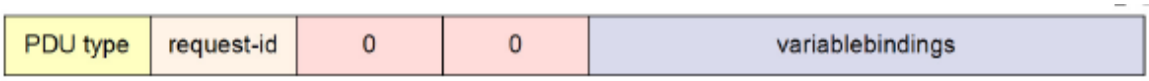
Prozivanje na temelju *trap* poruka štedi kapacitet mreže i procesorsko vrijeme obrade u agentima.

Slika 12. Operacija TRAP protokola SNMP
Izvor: William Stallings, Data and computer communications

4.5. SNMP PDU (Protocol data unit)

SNMP PDU (eng. *Protocol data unit*) je složena struktura podataka koja se sastoji od nekoliko manjih polja. U primjeni u SNMP protokolu postoje tri različite vrste PDU-a:

1. Vrsta SNMP PDU-a koju koriste SNMP poruke *GetRequest, GetNextRequest i SetRequest* (slika 13a).



***Slika 13a. PDU za poruke* GetRequest, GetNextRequest i SetRequest**
Izvor: William Stallings, Data and computer communications

2. Vrsta SNMP PDU-a koju koristi SNMP poruka *GetResponse* (slika 13b).

***Slika 13b. PDU za poruku* GetResponse**
Izvor: William Stallings, Data and computer communications

3. Vrsta SNMP PDU-a koju koristi SNMP poruka *Trap* (slika 13c).

***Slika 13c. PDU za poruku* Trap**

Izvor: William Stallings, Data and computer communications

Format SNMP PDU-a definiran je standardom ASN.1 (*Abstract Syntax Notation One*). Iako je polje *PDU type* dio SNMP PDU-a, ono nije definirano standardom ASN.1. Umjesto toga je za svaki od pet PDU-a definiran zaseban ASN.1 tip. Polje *request-id* (oznaka zahtjeva) se koristi za jednoznačno obilježavanje SNMP zahtjeva. Polja SNMP PDU-a su:

- *errorstatus* (oznaka ispravnosti zahtjeva) - označava da je za vrijeme obrade zahtjeva došlo do odstupanja od regularnosti. Ako je vrijednost *error-statusa* postavljena u 0, znači da je obrada zahtjeva protekla u redu. U protivnom je u polje *error-status* upisana vrijednost (različita od nule) koja pruža dodatnu informaciju o tome koja je varijabla prouzročila odstupanje od uobičajene obrade zahtjeva.
- *error-status i errorindex - u porukama GetRequest, GetNextRequest i SetRequest* su uvijek postavljeni na nulu.
- *variablebindings* (povezivanje varijabli) - predstavlja popis uređenih parova (ime varijable, vrijednost varijable), a vrijednosti svih varijabli u porukama *GetRequest i GetNextRequest* postavljene su na nulu.
- *enterprise* - označava vrstu objekta koji generira *trap* (temelji se na MIB varijabli *sysObjectID*).
- *agent-addr* - sadrži adresu objekta koji generira *trap*.
- *generic-trap* - određuje vrstu generičkog *trapa* (npr. *linkDown = 2*).
- *specifictrap* - sadrži informaciju specifičnu za određenu vrstu *trapa*.
- *time-stamp* - sadrži vrijeme koje je proteklo između posljednje inicijalizacije mrežnog entiteta i generiranja *trapa* (sadrži vrijednost MIB varijable *sysUpTime*).

4.6. SMI standard

SMI (eng. *Structure of Management Information*) predstavlja standard za izgradnju MIB baze. SMI je zapravo podskup od standarda ASN.1 uz neka proširenja. Ovim standardom određuju se vrste podataka koje se mogu koristiti u MIB-u, te način prikaza i imenovanja resursa MIB-a. Osnovni cilj je jednostavnost i proširivost MIB-a, pa se stoga dopuštaju samo jednostavni tipovi podataka (skalari i dvodimenzionalna polja skalara). SMI mora ispuniti sljedeće zahtjeve kako bi se postigao standardizirani način prikaza upravljanih informacija:

- pružanje standardizirane tehnike definiranja strukture pojedinog MIB-a,
- pružanje standardizirane tehnike definiranja pojedinih objekata MIB-a i
- pružanje standardizirane tehnike kodiranja vrijednosti objekata MIB-a.

Ti zahtjevi postižu se korištenjem identifikatora objekata, tipova objekata koji su definirani standardom ASN.1 i uporabom BER pravila kodiranja. Upravljani podaci prikazani su u obliku varijabli (npr. ime sustava, broj pokrenutih procesa) na agentima. Tim varijablama se onda mogu slati upiti za dohvat vrijednosti, ili ako je to predviđeno, postavljati vrijednosti od strane upravljača. SNMP koristi „dohvati-i-spremi" paradigmu (eng. *fetch and store*). SNMP poruka, dakle, služi postavljanju (*store*) ili nadzoru (*fetch*) varijabli SNMP agenta. Sama varijabla je instanca generičkog objekta. Definicija svake SNMP varijable počinje s oznakom OBJECT-TYPE ispred kojeg dolazi ime varijable. Time se definiraju svojstva varijable te postoje četiri obavezna parametra:

1. SYNTAX – definicija podatkovnog tipa varijable,
2. MAX-ACCESS – informacija o pravu pristupa varijabli (čitanje/pisanje),
3. STATUS – inačica SNMP protokola s kojom je varijabla u skladu i
4. DESCRIPTION – kratki opis varijable.

4.7. RMON – nadzor na udaljenoj lokaciji

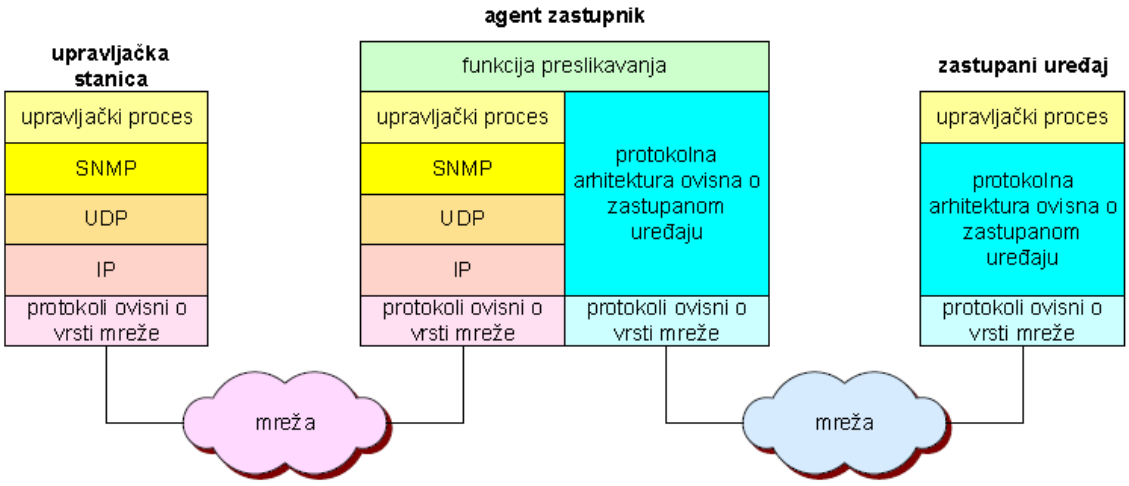
Standard RMON (eng. *Remote monitoring*) djeluje kao nadopuna SNMP standarda te definira mehanizme nadzora mreže u udaljenoj lokaciji bez trajnog sudjelovanja upravljačke stanice u tom procesu. Osnovna ideja je da se u upravljanom mrežnom uređaju implementira RMON sonda (eng. *probe*) koja prikuplja komunikacijsku statistiku u okolini mrežnog uređaja. Dakle, upravljačka stanica ne mora cijelo vrijeme prozivati mrežni uređaj kako bi prikupljala statističke podatke. Podaci se spremaju u bazu upravljačkih informacija u uređaju upravljanom od strane SNMP *managera* (u RMON MIB). SNMP *manager* ima mogućnost pristupa RMON MIB-u te dohvata svih željenih podataka. Takav se koncept osobito pokazuje učinkovitim u slučaju pada veze (eng. *link*) koja povezuje upravljačku stanicu (SNMP *manager*) i upravljani uređaj na mreži. RMON sonda može u bilo kojem trenutku obavijestiti stanicu o nastupu nekog važnog događaja u okolini upravljanog uređaja na mreži (tzv. *proactive monitoring*).

5. Primjena i sigurnost SNMP protokola

SNMP zahtijeva da svi agenti u NMS-u moraju podržavati jedinstveni OSI referentni model [10] zasnovan na protokolima UDP i IP. Takav pristup onemogućava primjenu SNMP upravljanja u uređajima kao što su npr. neki mostovi (eng. *Bridge*) i modemi, koji ne podržavaju TCP/IP OSI referentni model. Nadalje, postoje brojni manji sustavi (osobna računala, radne stanice, programibilni upravljači) u kojima nisu implementirani protokoli TCP/IP OSI modela. Zbog ograničenih procesnih mogućnosti nije poželjno u takve sustave ugrađivati dodatne programske implementacije koje podržavaju SNMP, agentsku logiku i održavanje MIB-ova. Neki su uređaji preopterećeni komunikacijskim poslovima pa također nije preporučljivo u njih instalirati podršku za SNMP. Kako bi se otklonili gore navedeni problemi te kako bi se spomenute uređaje uključilo u jedinstveni sustav upravljanja mrežom uveden je koncept *agenta zastupnika* (eng. *proxy agent*), odnosno posrednika u upravljanju mrežom.

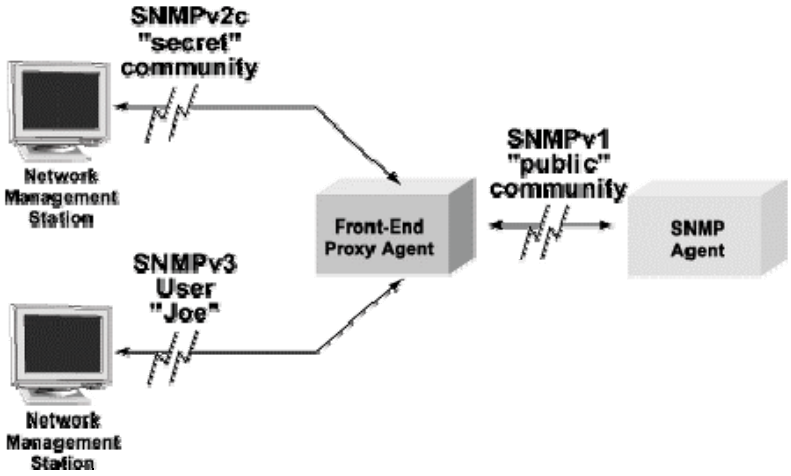
5.1. SNMP AGENT ZASTUPNIK

Jedan agent zastupnik može zastupati više mrežnih uređaja. S jedne strane agent zastupnik komunicira s aplikacijom mrežnog upravljanja, tj. s upravljačkom stanicom (SNMP *manager*) ili više njih, a s druge strane komunicira sa zastupanim uređajima (koji nemaju implementiran SNMP protokol, ili imaju drugu inačicu SNMP protokola). Kao što je vidljivo na slici 14, upravljačka stanica šalje SNMP upite agentu zastupniku koji ih pretvara u poruke upravljačkog protokola kojeg koristi zastupani uređaj, u ovom slučaju neki drugi korisnički uređaj. Nakon što primi odgovor od zastupanog uređaja agent zastupnik proslijedi odgovor u obliku SNMP poruke upravljačkoj stanici.



Slika 14. Primjer primjene agenta zastupnika (proxy agent)
Izvor: Douglas R. Mauro, Kevin James Schmidt - Essential SNMP

Jedna od mogućih primjena koncepta agenta zastupnika je povezivanje upravljačke stanice koja koristi SNMPv3 i mrežnih uređaja koji koriste SNMPv1 ili SNMPv2 (slika 15). U većini današnjih mrežnih uređaja nije još implementirana treća inačica protokola SNMP što svakako predstavlja prijetnju sigurnosti takvog sustava. Primjena agenta zastupnika može bitno reducirati probleme vezane uz sigurnost.



Slika 15. Povezivanje inačica SNMPv1 sa SNMPv2 i SNMPv3

Izvor: snmp.com

5.2. Sigurnost SNMP protokola

5.2.1. Sigurnost SNMPv1 protokola

SNMPv1 protokol se i danas dosta koristi unatoč poznatim sigurnosnim nedostacima. Sigurnost se kod SNMPv1 temelji na korištenju takozvanih zajedničkih znakovnih nizova (eng. *community string)* Najveći problem (prethodno već objašnjenog principa rada SNMPv1 u poglavlju 3.1.1) je u tome što neovlašteni korisnici mogu snimanjem IP paketa koji se prenose mrežom pročitati sadržaj SNMP poruka i saznati *community string* jer se on prenosi u čitljivom (nekriptiranom) obliku. Znajući *community string*, napadači mogu pristupiti upravljačkim informacijama nekog mrežnog uređaja i, što je još opasnije, promijeniti njegovu konfiguraciju. Neki od prijedloga za sigurniju uporabu SNMPv1 inačice su :

- Obratiti pozornost na nesigurne tvorničke (eng. *default*) postavke. Većina proizvoda sa SNMPv1 inačicom inicijalno dođu sa *community stringom* postavljenim na vrijednost „PUBLIC“. To je vjerojatno prva opcija koju će neovlašteni korisnik provjeriti. Zato je važno promijeniti nazive svim *community stringovima.*
- Često mijenjati *community string* imena, naravno ukoliko je to moguće i praktično. Ako se mreža sastoji od nekoliko stotina uređaja kojima se upravlja SNMP protokolom to nije praktična niti učinkovita akcija.
- Odabrati što složenija *community string* imena.
- Podesiti vatrozid tako da se omogući pristup samo nadziranim uređajima ili SNMP agentima s točno određene lokacije.
- Namjestiti *Trap* poruke tako da izvještavaju o svakom pokušaju komunikacije s krivim *community string* nazivom.

5.2.2. Sigurnost SNMPv2 protokola

SNMPv2 protokol je uveden u TCP/IP mrežama 1993. godine. On je nudio neka proširenja kao što su dodatne operacije i korištenje *party-based* sigurnosti koja se pokazala veoma složenom u svakodnevnoj upotrebi. Zbog toga SNMPv2 nije nikada zaživio u upravljanim mrežama. Umjesto njega je 1995. godine prihvaćen kao standard u TCP/IP mrežama *Community-Based Simple Network Management Protocol version 2*- SNMPv2c. Kao što mu i samo ime kaže, SNMPv2c protokol svoju sigurnost zasniva također na zajedničkim znakovnim nizovima tako da su sigurnosni problemi ostali isti kao i kod SNMPv1 protokola.

5.2.3. Sigurnost SNMPv3 protokola

SNMPv3 protokol je uveden u TCP/IP mrežama 1998. godine. Tek u trećoj verziji, SNMPv3, definirani su bitno poboljšani sigurnosni mehanizmi. SNMPv3 posjeduje mehanizme za sigurnu kontrolu pristupa, autentikaciju, odnosno provjeru vjerodostojnosti korisnika, te enkripciju, odnosno zaštitno kodiranje SNMP poruka. SNMPv3 može koristiti takozvanu korisničku (*user-based*) autentikaciju ili se provjera vjerodostojnosti korisnika može obaviti bez slanja lozinki u čitljivom obliku a temelji se na upotrebi algoritama HMAC-MD5 ili HMAC-SHA. Zaštitna enkripcija koristi 56-bitni CBC-DES algoritam za kodiranje i dekodiranje SNMP poruka.

SNMPv3 protokol osigurava sljedeće sigurnosne značajke:

- **Integritet poruka** - osigurava da prijenos SNMP poruka bude neometan, odnosno da SNMP paket na odredište stigne nepromijenjen,
- **Autentikacija** - osigurava da SNMP poruka dolazi od valjanog izvora i
- **Enkripcija** - kodira sadržaj SNMP paketa da bi se spriječila zloupotreba od neovlaštenog korisnika.

SNMPv3 protokol osigurava, osim sigurnosnih razina, i sigurnosne modele. Sigurnosni model je autentikacijski proces koji postoji kako za korisnika tako i za grupu u kojoj se korisnik nalazi. Sigurnosna razina je dozvoljena razina sigurnosti unutar sigurnosnog modela, odnosno tip sigurnosnog algoritma koji je dozvoljen u SNMP paketu. Kombinacija sigurnosnog modela i sigurnosne razine određuje koji se sigurnosni proces koristi kod manipulacije sa SNMP paketom.

Postoje tri sigurnosna modela: SNMPv1, SNMPv2c i SNMPv3.

Također, postoje tri sigurnosne razine:

- **NoAuthNoPriv** - autenticira SNMP paket koristeći *username* ili *community string* znakovni niz,
- **AuthNoPriv** - autenticira SNMP paket koristeći ili HMAC-MD5 ili HMAC-SHA algoritam i
- **AuthPriv** - autenticira SNMP paket koristeći ili HMAC-MD5 ili HMAC-SHA algoritam i enkriptira paket koristeći 56-bitni DES algoritam.

Sljedeća tablica prikazuje sve kombinacije sigurnosnih modela i sigurnosnih razina kod SNMP protokola:

Verzija SNMP-a	**Sigurnosna razina**	**Autentikacija**	**Enkripcija**
SNMPv1	NoAuthNoPriv	Community	-
SNMPv2c	NoAuthNoPriv	Community	-
SNMPv3	NoAuthNoPriv	Username	-
SNMPv3	AuthNoPriv	MD5 ili SHA	-
SNMPv3	AuthPriv	MD5 ili SHA	DES

Tablica 2. Sigurnosni modeli i razina zaštite za inačice SNMP

Izvor: Sigurnost SNMP protokola [10]

Radi sigurnosti SNMP komunikacije svakako se preporuča korištenje SNMPv3 protokola ako to upravljačke aplikacije i agenti omogućavaju. No, unatoč sigurnosnim slabostima SNMPv1 i SNMPv2c protokola, još uvijek se veliki broj upravljanih mreža zasniva na njima. To se događa jer neke upravljačke aplikacije još uvijek ne podržavaju SNMPv3. U tom slučaju, u upravljanim mrežama sa SNMPv1 ili SNMPv2c protokolima, radi povećanja sigurnosti, poželjno je korištenje filtara, tj. pristupnih listi. To su liste u kojima se izravno daje pravo pristupa SNMP agentu korisnicima određenima s tri parametra:

- lozinka – je naziv *community stringa* koji može pristupiti agentu,

- adresa – je lokacija s koje se može pristupiti agentu i
- OID – ograničava pristup samo na one čvorove koji su podređeni u stablu kojem je korijen OID.

Lista je zapravo naziv konfiguracijske datoteke u kojoj su pohranjeni navedeni podaci.

Nužni minimum kojeg je potrebno imati za praktičnu realizaciju upravljanja mrežom je instalacija agentske programske podrške u mrežne uređaje i programska implementacija koja može dohvaćati vrijednosti iz MIB-ova u upravljanim uređajima pomoću operacije SNMP *get*. U mrežnim uređajima kao što su usmjerivači (*router*) i komutatori (*switch)* agentsku programsku podršku instalira proizvođač opreme.

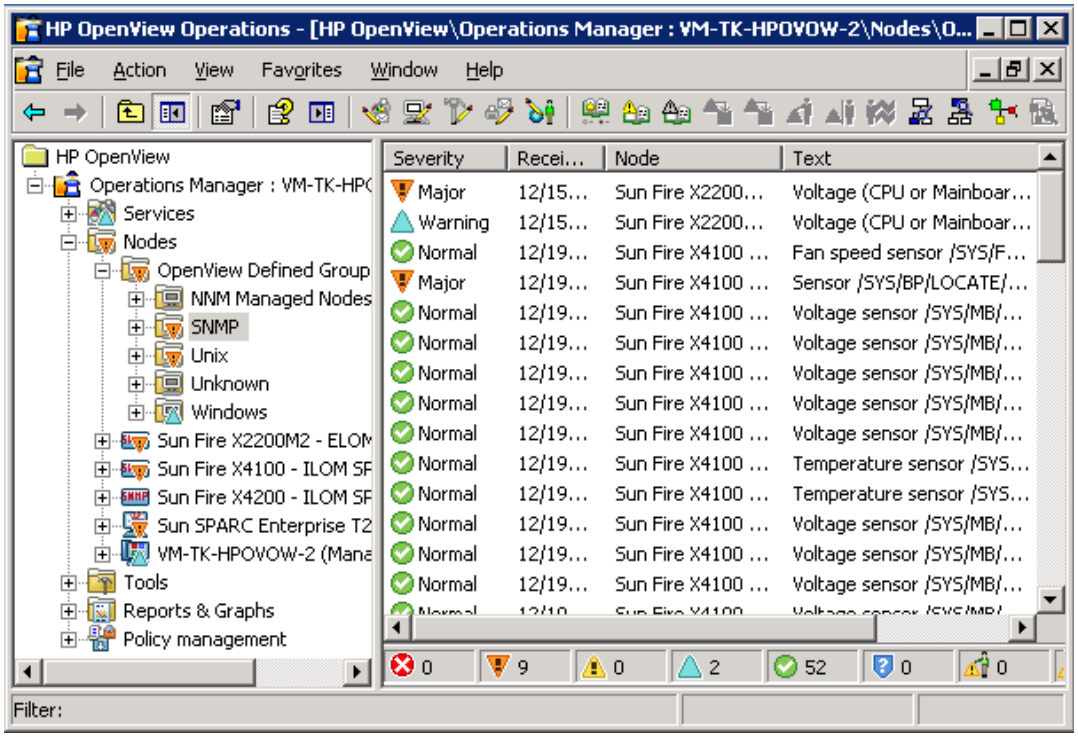
5.3. Programska implementacija SNMP upravljačke jedinice

Upravljanje mrežnim sustavima danas je sveprisutno i neophodno u gotovo svim svjetskim organizacijama. Vrsta programske implementacije koja prati upravljanje mrežnim sustavom u pojedinoj organizaciji ovisna je o veličini i složenosti mrežne infrastrukture. U skladu s tim, organizacije se odlučuju za besplatna ili komercijalna programska rješenja.

5.3.1. Komercijalne programske implementacije

Hp OpenView

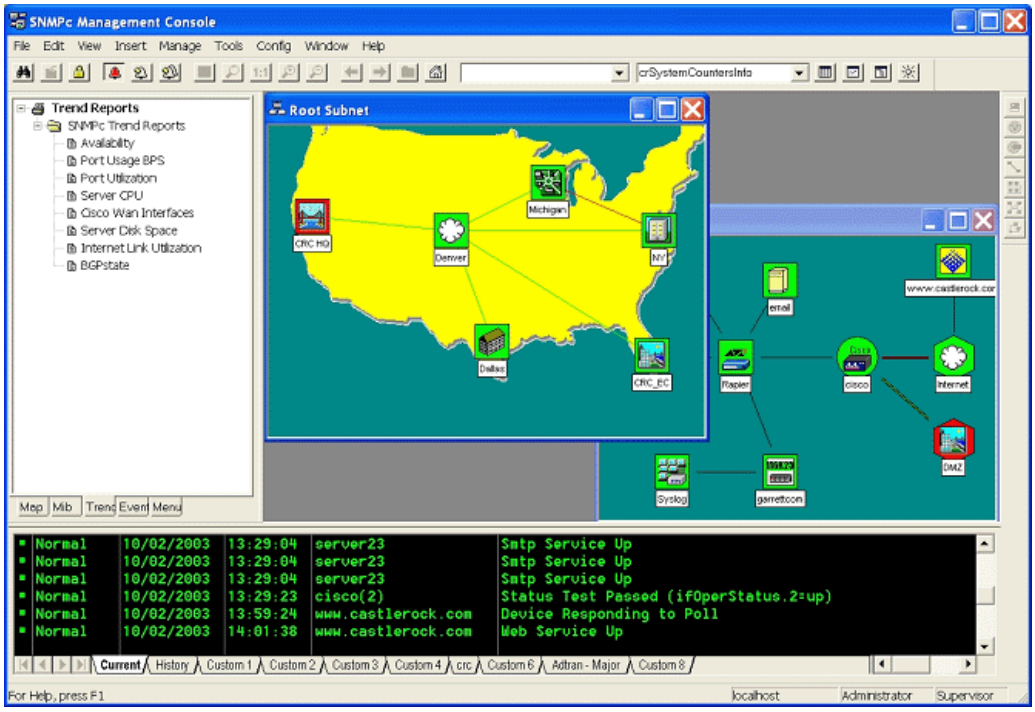
Ako postoji potreba za realizacijom složenijeg i funkcionalnijeg NMS-a tada je potrebno koristiti alate kao što je primjerice. ***HP OpenView***, proizvod tvrtke *Hewlett Packard*. Takav alat, između ostalog, omogućava automatsko otkrivanje mrežne topologije, administriranje korisnika, konfiguriranje intervala prozivanja i druge napredne funkcionalnosti. Neke od operacija i funkcionalnosti za operacijski sustav Windows prikazane su na slici 16. lako je izdan i održavan od strane HP-a, namijenjen je i podržan i za sklopovlja drugih proizvođača. Složenost i cijena priklanjaju uporabu HP OpenViewa samo velikim organizacijama.



Slika 16. Hp OpenView operacije za Windows operacijski sustav
Izvor:Hp OpenView [12]

SNMPc Castle Rock Computing

SNMPc Castle Rock Computing je programska implementacija namijenjena operacijskom sustavu Windows. Prikladna je za organizacije koje traže provjerenu i moćnu programsku podršku NMS-u, koja je, s druge strane, jednostavna za korištenje. Jednostavnost se izražava kroz intuitivno korisničko sučelje (slika 17) i grafički prikaz trenutnog stanja NMS-a. Podržava automatsku dojavu i izvješća svih neželjenih događaja na praćenoj mreži. Za razliku od HP OpenViewa, radi se o usko specijaliziranoj programskoj podršci usmjerenoj mehanizmima SNMP protokola.



Slika 17. SNMPc Castle Rock Computing
Izvor: www.hw-group.com

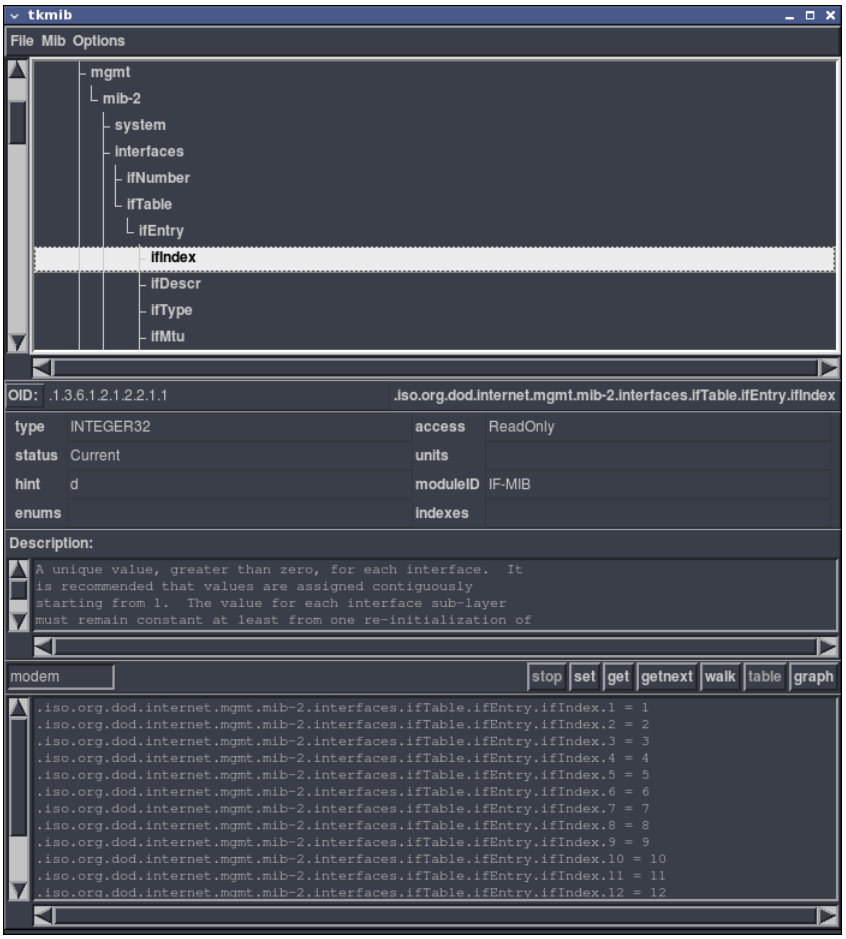
MSC Operations Manager 2007 & MSC essentials 2007

Programsko rješenje koje se temelji na System Center Operation Manager 2007 i Essentials 2007. Pripada među bolje alate za operacije praćenja i nadzora NMS-a, te cjelokupne IT infrastrukture. Microsoftova paleta proizvoda pod nazivom System Center obuhvaća nekoliko proizvoda koji pružaju funkcionalnost nadzora i upravljanja cjelokupnom informatičkom infrastrukturom. Standardne odlike Microsofta su jednostavno korisničko sučelje i pomoćne aplikacije (*wizards*) koje uvelike olakšavaju korištenje alata. Podržane su funkcionalnosti poput grafičkog prikaza analize NMS-a i zvučne dojave neželjenih događaja.

5.3.2. Besplatne programske implementacije

Net-SNMP

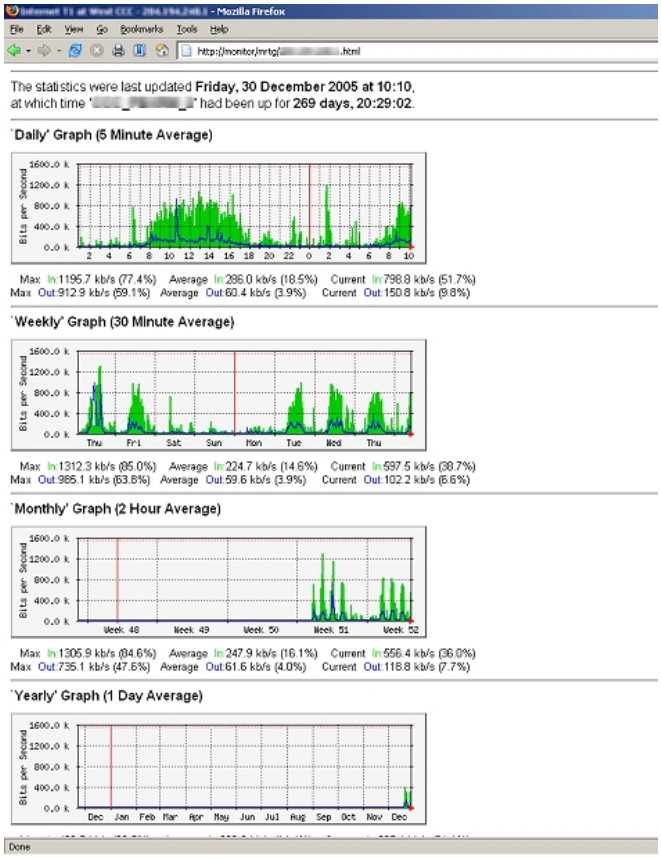
Programski paket Net-SNMP se koristi za implementaciju protokola SNMP v1, SNMP v2C i SNMP v3. Omogućava proširivanje agenata pomoću konfiguracijskih datoteka te pruža ostvarenje jednostavne upravljačke aplikacije. Grafičko sučelje prema korisniku moguće je realizirati pomoću skriptnog jezika *Perl i* alata *Td/Tk*. Primjer jednog takvog sučelja za NET-SNMP prikazan je na slici 18. Net-SNMP je besplatan program dostupan svakome te ga je moguće jednostavno skinuti s Interneta.



Slika 18. NET-SNMP GUI
Izvor: net-snmp.sourceforge.net

MRTG

MRTG (eng. *Multi Router Traffic Grapher*) je besplatan programski alat kojeg je moguće naći na stranici www.mrtg.org. MRTG je posebno koristan za utvrđivanje vršnih opterećenja mrežnih sučelja i uređaja unutar duljih vremenskih intervala. Izlaz kojeg generira moguće je pregledavati pomoću bilo kojeg web preglednika (slika 19).

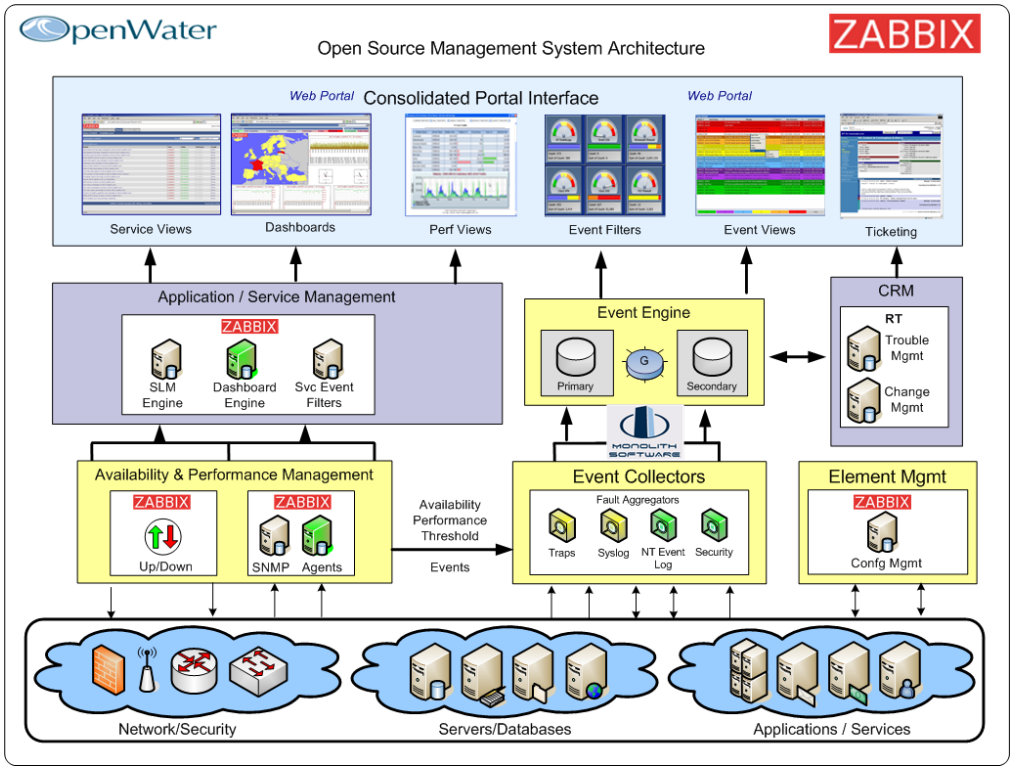


Slika 19. MRTG graf

MRTG se sastoji od Perl skripti koje koriste SNMP za analizu prometa usmjerivača (eng. *router*) te od brzog i jednostavnog C programa koji na temelju tih podataka o prometu kreira grafove. Grafovi predstavljaju stanje prometa na mreži. Uz detaljan dnevni pregled, MRTG također kreira i prikaz prometa u posljednjih sedam dana, proteklih pet tjedana i posljednjih dvanaest mjeseci (slika 19). To je moguće jer MRTG čuva zapisnik svih podataka koje je izvukao iz usmjerivača.

ZABBIX

ZABBIX je programska implementacija za praćenje korisničkih aplikacija, mreže i poslužitelja. Praćenje uređaja koji su upravljani od strane SNMP *managera* u ZABBIX-u je moguće koristeći oba mehanizma SNMP protokola (*trap* poruke i prozivanje). Dinamičan mehanizam obavijesti pruža lak i učinkovit način uzbune u slučaju neželjenih događaja. ZABBIX je moguće lako i jednostavno skinuti s Interneta. Arhitektura ZABBIX programske podrške predočena je na slici 20.



Slika 20. Arhitektura programske podrške ZABBIX
Izvor: zabbix.com

6. Budućnost SNMP protokola

Organizacije danas koriste IT sektor kao jedan od temeljnih stupova svoga poslovanja. Poslovni sustavi su oduvijek težili poboljšanju postojeće infrastrukture, jeftinijoj i učinkovitijoj implementaciji svojih poslovnih rješenja i, naravno, većoj zaradi. Sve to rezultira i razvojem informacijskih tehnologija u tom smjeru, konstantnom inovativnošću, neprestanim razvitkom novih tehnologija i okretanju ka globalnoj povezanosti putem Internet mreže. Pošto je svaka ozbiljnija organizacija danas interno povezana računalnom mrežom, nužan faktor dobrog poslovanja je upravljanje i nadzor mrežne okoline. SNMP je danas jedan od najrasprostranjenijih i najčešće korištenih protokola za upravljanje i nadzor mrežne okoline. Razvoj SNMP protokola zasada je stao na trećoj inačici - SNMPv3, no daljnji razvoj alata i protokola za upravljanje mrežnim sustavima u neprestanom je usavršavanju. IT sektor usmjeren je na prebacivanje većine vlastitih resursa na Internet. Budućnost upravljanja mrežom je usmjerena na sustav koji se temelji na Internet (web) tehnologijama.

*6.1. Internet sustav upravljanja mrežom (*WEB-NMS*)*

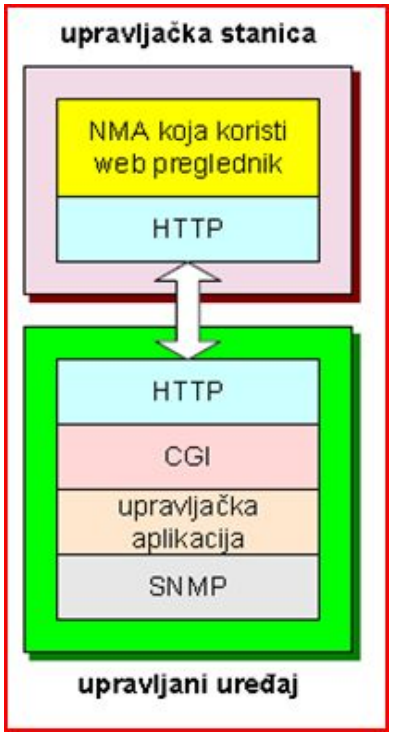
Sustav upravljanja mrežom koji se temelji na korištenju web tehnologija oslanja se na protokol HTTP i općenito na aplikacijski sloj OSI modela. Jedan od razvijenih mrežnih standarda kojem je cilj objediniti upravljanje računalnim mrežama je WBEM (eng. *Web-based Enterprise Management*), razvijen od strane organizacije DMTF (eng. *Distributed Management Task Force*).

DMTF je osnovan 1992. godine, tada kao *Desktop Management Task Force*. DMTF je neprofitna organizacija, a njezino djelovanje bilo je usmjereno ka normama za upravljanje osobnim računalima. S razvojem novih tehnologija i računalne industrije postalo je sve kompliciranije upravljati brojnim entitetima na korporativnoj razini. Ovakva evolucija odrazila se i na DMTF koja je zadržala svoj prvobitni akronim, ali se njegovo značenje promijenilo u *Distributed Management Task Force*. Današnja misija DMTF-a je koordiniranje razvoja norme za upravljanje distribuiranim računalnim sustavima te *enterprise* mrežnim i Internet okruženjima. DMTF čini nekoliko vodećih kompanija među kojima su 3Com, Hewlett-Packard, Cisco, Microsoft, SUN, Intel, IBM/Tivoli Systems, Novell i mnogi drugi.

DMTF je pokrenuo inicijativu WBEM (eng. *Web-based Enterprise Management).* WBEM je tehnologija koja omogućuje jednostavnu razmjenu podataka o upravljivim računalnim sustavima. DTMF je razvio temeljni skup standarda koji sadrže podatkovni model, kodnu specifikaciju te transportni model. WBEM se na najjednostavniji način može opisati kao skup standardiziranih apstrakcijskih slojeva koji skrivaju kompleksnost pristupu informacijama za upravljanje. Primjer implementacije WBEM-a u praksi je WMI (eng. *Windows Management Instrumentation*) sustav namijenjen nadzoru i upravljanju računala temeljenih na operacijskom sustavu Microsoft Windows. WMI intenzivno koristi CIM (eng. *Common Information Model*), jedan od standarda WBEM-a. CIM je općeniti model za opis podataka o upravljivim komponentama sustava (servisi, aplikacije, računala i sl.), neovisan o platformi i tehnologiji koja se koristi. CIM sadrži specifikaciju i podatkovnu shemu. Specifikacija definira detalje integracije s drugim modelima za upravljanje (npr. SNMP, MIB i CMIP), dok podatkovna shema opisuje potrebne podatkovne modele.

Vrlo važna mrežna specifikacija, dizajnirana tako da omogući ostvarivanje inteligentnijih računalnih mreža logičkim povezivanjem mrežnih servisa sa korisnicima i poslovnim kriterijima, je DEN (eng. *Direcory Enabled Networks*). Današnji sustavi teže ka raspodijeljenoj arhitekturi. DEN logičkim povezivanjem mrežnih komponenti raspodijeljenog sustava omogućuje i upravljanje mrežom. On specificira općeniti podatkovni model koji može koristiti brojne tehnologije od CIM-a do X.500 (serija mrežnih standarda). Ovakva širina tehnologija za upravljanje mrežom stvara temelje za kreiranje predložaka za razmjenu informacija i omogućuje dobavljačima sklopovlja i programske podrške međusobnu razmjenu definicija uređaja, aplikacija i servisa. DEN, također, dopušta kooperabilnost s rješenjima temeljenim na WBEM-u.

SNMP protokol je u arhitekturi WBEM temeljenog NMS-a samo jedan od alata za komunikaciju unutar računalnog sustava. WBEM je tehnologija koja se oslanja na protokol HTTP i programsko sučelje CGI (eng. *Common Gateway interface*), kao što je i predočeno na slici 21. U agenta se ugrađuje web poslužitelj zajedno s CGI pogonom (*engine*) koji pretvara upravljačke zahtjeve primljene od NMA (eng. *Network Management Aplication*) u stvarne SNMP operacije, i obratno. CGI predstavlja vezu između NMA i SNMP pogona. NMA može biti realizirana kao skup Java apleta (*applets*) koji se prikupljaju pomoću web preglednika (*web browser*) i izvršavaju na uređaju.

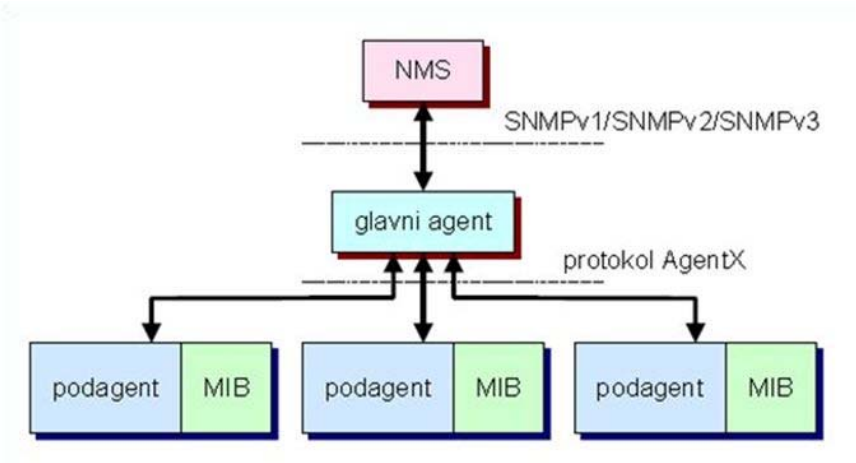


Slika 21. Web based-NMS
Izvor: Douglas R. Mauro, Kevin James Schmidt – Essential SNMP

Ovo su samo neke od važnijih inicijativa u budućem razvoju upravljanja mrežnim sustavima. Upravljanje mrežom pomoću web alata otklanja ili barem reducira upotrebu tradicionalne NMA programske implementacije.

6.2. SNMP AgentX

Potreba za arhitekturom NMS-a nazvanom *AgentX* (*Agent eXtensibility*) nastala je na temelju činjenice da je u standardnom SNMP NMS-u nemoguće dodavati i uklanjati objekte MIB-a dok je agent aktivan. Stoga je definiran glavni agent (*master agent*) i podagenti (*subagents*). Glavni agent i podagenti mogu biti instalirani na istom uređaju ili komunicirati pomoću posrednika *(proxy device*). Podagenti mogu izravno pristupati MIB-u, dok glavni agent ne posjeduje tu mogućnost. Glavni agent i podagenti međusobno komuniciraju protokolom *AgentX* kao što je vidljivo na slici 22.



Slika 22. AgentX protokol

Izvor: Douglas R. Mauro, Kevin James Schmidt – Essential SNMP

7. Zaključak

SNMP protokol u svom nazivu ima pridjev „jednostavan“, jer na poprilično jednostavan način (čitanjem i pisanjem u varijable te jednostavnom komunikacijom porukama između nadziranih uređaja) razdvaja upravljačku arhitekturu od arhitekture sklopovlja.

Dizajniran je da minimizira složenost i broj upravljačkih funkcija realiziranih od agenata, a da opet bude fleksibilan kako bi se mogao prilagoditi nepredvidljivim aspektima mrežnih operacija nadzora i upravljanja. Glavni aduti SNMP-a su jednostavnost i interoperabilnost. Njegova važna odlika SNMP je da mora efektivno raditi i kada mreža nije potpuno operabilna. To se odražava u izboru nespojnog transportnog protokola (UDP) koji dopušta upravljačkim aplikacijama potpunu kontrolu nad mehanizmom retransmisije.

Inženjerima koji su razvijali SNMP protokol drugi dizajnerski cilj je bio držati SNMP što je moguće nezavisnijim od drugih mrežnih servisa. To je jedan od glavnih razloga zašto su u SNMPv3 sigurnosni mehanizmi samostalni (sigurnosni algoritmi HMAC-MD5, HMAC-SHA, CBC-DES) i ne ovise o vanjskim sigurnosnim mehanizmima. Okolina u kojoj se odvijaju upravljačke operacije značajno se promijenila od vremena kad je SNMP osmišljen. Gledajući današnje mrežne tehnologije i stvarnu upotrebu SNMP modela, očito je kako bi uređaji mogli obavljati još kompleksnije upravljačke operacije uz nisko opterećenje.

Razumno je očekivati kako će uređaji, pogotovo novi usmjerivači i preklopnici, postati sve više programibilni i kako će postati moguće pokretanje sve snažnije kontrolne programske podrške na tim uređajima. Budućnost se nazire u sve većem okretanju upravljanju računalnim sustavom temeljenom na webu.

SNMP protokol uvelike olakšava kompletan proces upravljanja i nadzora nad računalnim sustavima, ali radi veće učinkovitosti bitno je i posložiti ostale faktore u cjelokupnoj politici upravljanja koja ovisi od pojedine organizacije te nikako nije univerzalna. Politika upravljanja računalnom mrežom uključuje i potrebnu edukaciju korisnika kako bi se smanjili sigurnosni rizici te ispravna uporaba uređaja na mreži.

8. Reference

[1] A.Bažant: Protokoli upravljanja mrežom
[2] D.Bruey: SNMP: Simple Network Management Protocol, 2005.
[3] Wikipedia: SNMP
http://en.wikipedia.org/wiki/Snmp
[4] CERT: SNMP FAQ
http://www.cert.org/tech_tips/snmp_faq.html
[5] SNMP - The Simple Network Management Protocol
http://www.henrys.de/daniel/index.php?cmd=texte_winsock_snmp.htm
[6] SNMP Implementation
http://www.dpstele.com/white-papers/snmp-implementation/introduction1.php
[7] SNMP - simple management tool for hackers?
http://www.networkworld.com/newsletters/sec/1004sec1.html
[8] William Stallings „Data and computer communications“
[9] Essential SNMP, O'Reilly, 2005.
[10] Wikipedia: OSI model,
http://en.wikipedia.org/wiki/OSI_model
[11] Sigurnost SNMP protokola (SRCE Sistemac)
[12] Hp OpenView
http://www.sun.com/bigadmin/features/articles/3pmi_mgmt.full.jsp