TAMPEREEN
AMMATTIKORKEAKOULU

OPINNÄYTETYÖ

Nagios – Verkon palveluiden valvonta

Joni Rauta

Tietojenkäsittelyn koulutusohjelma
Marraskuu 2006
Työn ohjaaja: Harri Hakonen

Tampere 2006

TAMPEREEN
AMMATTIKORKEAKOULU

Tekijä Joni Rauta

Koulutusohjelma(t) Tietojenkäsittely

Opinnäytetyön nimi Nagios – Verkon palveluiden valvonta

Työn valmistumis-kuukausi ja -vuosi Marraskuu 2006

Työn ohjaaja Harri Hakonen **Sivumäärä:** 53

TIIVISTELMÄ

Tässä opinnäytetyössä käsitellään verkonvalvontasovellus Nagiosta, ja sen eri tarkastusmenetelmiä. Tarkastuksissa keskitytään lähinnä SNMP -protokollaa hyödyntäviin keinoihin, mikä oli myös opinnäytetyön ensisijainen tavoite.

Nagioksella voidaan valvoa verkossa olevia palvelimia, ja niihin kuuluvia palveluita. Lisäksi työssä esitellään Nagioksen käyttöliittymä ja sen toiminnot, sekä selvitellään siitä, miten tarkastuksia voidaan tehdä jaetusti monella Nagios-palvelimella.

Tämän työn toimeksiannon sain Netum Oy:ltä. Käsittelen opinnäytetyön teoriaosuudessa verkonhallinnan eri osa-alueita sekä SNMP:n toimintaa. Käytännön osuudessa asennettiin testiympäristössä oleviin Linux ja Windows-palvelimiin SNMP-agentit, joiden tehtävä oli kerätä tietoa palvelimien resursseista. Nagios-palvelimelle asennettavien pluginien avulla agenttien keräämä tieto oli nähtävissä Nagioksessa.

Tätä opinnäytetyötä voidaan käyttää apuna Nagioksen uudelleenasennuksessa ja päivittämisessä, sekä konfiguroinnissa SNMP:tä hyväksi hyödyntäen. Nagios-verkonvalvonta- sovelluksesta kiinnostuneille tämä työ sopii myös tietolähteeksi ja ohjeeksi.

Avainsanat Nagios SNMP Verkonvalvonta

TAMPERE
POLYTECHNIC

Author	Joni Rauta
Degree Programme(s)	Business Information Systems
Title	Nagios – Monitoring of network services
Month and year	November 2006
Supervisor	Harri Hakonen

Pages: 53

ABSTRACT

This thesis is handling network monitor program Nagios and it's different checking methods. In these checkings are concentrated mainly on the methods using SNMP –protocol. This also was the primary mission for this thesis.

With Nagios you can monitor the servers in the network and the services included. This thesis also presents Nagios user interface and its functions, and it is told how theses checkings can be made dividedly using many Nagios servers.

I got the commission for this thesis from Netum LTD. In the theory passage of this thesis I handle the different parts of network monitoring and the function of SNMP. In the practical parts the SNMP –agents were installed to the Linux and Windows servers in the test environment. The SNMP –Agent's task was to collect information of the resources of servers. With the help of the plugins installed to Nagios server the information collected by the agents was to be seen to Nagios.

This thesis can be used as a help on Nagios´s re-installation and datings and also in the configuration using SNMP. For those interested in Nagios network monitor program this thesis is suitable as a guidebook.

Keywords Nagios SNMP Verkonvalvonta

Sisällysluettelo

1 JOHDANTO

Tietojärjestelmien palvelukatkokset ja niiden tuomat negatiiviset tapahtumaketjut voidaan välttää verkon palveluiden ja palvelinten valvomisella siihen tarkoitetulla valvontasovelluksella. Opinnäytetyössäni keskityn Nagios verkonvalvontasovellukseen, ja sen tuomiin mahdollisuuksiin paikallistaa potentiaalinen vikatilalle lähiverkon palveluissa sekä palvelimissa.

Suoritin opintoihini kuuluvan viiden kuukauden työharjoittelun Netum Oy:ssä vuonna 2006. Netum Oy on tietotekniikan palveluyritys, joka vastaa mm. käytettävyyspalveluista, sovelluskehityksestä sekä konsultoinnista. Sain kyseisestä yrityksestä myös toimeksiannon opinnäytetyötä varten. Toimeksiantaja toivoi selvitystä siitä, miten yrityksessä jo käytössä olevan verkonhallintasovellus Nagioksen tarkastuksia voidaan tehdä muun muassa eri tarkastusmenetelmillä, ja kuinka nopeasti ja mitä eri tarkastuksia voidaan ottaa käyttöön. Käytössäni oli pieni testiympäristö, johon kuului Linux ja Windows – palvelimia sekä tulostimia. Työssäni keskityn enimmäkseen SNMP (Simple Network Management Protocol) - protokollan käyttämiseen Nagioksen tekemissä tarkastuksissa, mutta käyn läpi myös NRPE:n (Nagios Remote Plugin Executor) sekä NSCA:n (Nagios Service Check Acceptor) käyttämisen mahdollisuudet.

Opinnäytetyöni jakaantuu neljään eri osaan. Ensimmäisessä osassa käydään läpi verkonhallinnan eri osa-alueet sekä niiden merkitykset. Toinen osa keskittyy Nagios – sovelluksen esittelyyn sekä sen peruskonfiguraatioon. Kolmas osa-alue kattaa Nagiosen tekemät tarkastukset sekä niissä käytettävät työkalut. Neljännessä ja viimeisessä osassa käydään läpi mahdollisia kehittämisratkaisuja, jotka lisäävät järjestelmien vikasietoisuutta.

2 VERKONHALLINTA

Kun verkonvalvontaa tarkastellaan kokonaisuutena, siinä on tunnistettavissa eri osa-alueita. Alueiden jako on tehty osana ISO:n (International Organization for Standardization) OSI (Open Systems Interconnection) järjestelmähallintaa. Tämä jako on saavuttanut laajan hyväksynnän, ja se on käytössä myös muiden verkonhallintajärjestelmien vaatimusten kuvauksessa. Avainalueet ISO:n määrittelemässä verkonhallinnassa ovat: vikojen hallinta, käytön hallinta, kokoonpanon hallinta, suorituskyvyn hallinta ja turvallisuuden hallinta. (Hautamäki 1994)

2.1 Vikojen hallinta

Vikojenhallinta sisältää verkon vikojen havaitsemisen, eristämisen ja korjaamisen. Vikojenhallintaan kuluu: virhelokien ylläpito, toimenpiteiden suorittaminen vikahavaintojen perusteella, diagnostiikkatestien tekeminen vikojen seuraamiseksi ja yksilöimiseksi ja vikojen korjaaminen. Vian eristäminen ja laitteen ominaisuuksien hyväksikäyttö vikatilanteessa liittyy myös vianhallintaa. (Jaakohuhta 2005: 309)

2.2 Käytön hallinta

Verkon ylläpitäjälle on tärkeää pystyä seuraamaan verkon resurssien käyttöä käyttäjä- tai käyttäjäryhmätasolla. Tätä tietoa tarvitaan esimerkiksi laskutukseen, verkon käytön tehokkuuden varmistamiseen sekä verkon laajennusten ja parannusten suunnitteluun. Ylläpitäjän on pystyttävä määrittelemään, mitä tietoa kerätään, mistä sitä kerätään ja kuinka usein tieto kootaan yhteen. Samoin on pystyttävä määrittelemään, miten koottu tieto analysoidaan ja mahdollinen laskutus suoritetaan. Käytön hallinnan ensisijainen etu on se, että se mahdollistaa seurata verkon resurssien todellista käyttöä. Siitä saatava informaatiota tarvitaan verkkoon suunnattavien investointien kohdistamisessa oikeisiin paikkoihin. (Stallings 1993: 625)

2.3 Kokoonpanon hallinta

Kokoonpanon hallintaa käytetään fyysisten (lisäkortit, sovittimet jne) ja loogisten (pääsylistat, reititysasetukset) olioiden käsittelyyn ja yksi-

löimiseen. Se sisältää toimintoja hallittavien olioiden luontiin, alustamiseen ja poistamiseen. Siihen kuuluu myös oleellisena osana nimien liittäminen hallittaviin olioihin. Tämän lisäksi se antaa mahdollisuuden muuttaa ja lukea olioiden attribuutteja.
(Jaakohuhta 2005: 310)

Kokoonpanon hallinnan ensisijainen etu on mahdollisuus muuttaa verkon rakennetta. Esimerkiksi ongelmatilanteessa muutetaan reititystä niin, ettei vikaantunut laite aiheuta häiriötä verkon toimiviin osiin. (Stallings 1993: 625)

2.4 Suorituskyvyn hallinta

Suorituskyvyn hallinta kerää ja analysoi tietoa verkon suorituskyvystä. Useimmat verkkojen laitteet käyttävät hyväkseen verkosta saatuja jaettuja resursseja esimerkiksi levytilaa, ja nimenomaan resurssien jakamisen tarve on melkein aina ollut verkon rakentamisen syynä. Joillekin sovelluksille, jotka kommunikoivat verkon välityksellä, on kriittistä että verkon suoristuskyky on riittävällä tasolla. Tietokoneverkon suorituskyvyn hallinta koostuu valvonnasta ja hallinnasta. Valvonta tarkoittaa verkon liikenteen tarkkailua, kun taas hallinta mahdollistaa suorituskyvyn parantamisen tarjoamalla ratkaisut verkon asetusten säätämiseksi. (Jaakohuhta 2005: 310)

2.5 Turvallisuuden hallinta

Turvallisuuden hallinta on verkkoon ja siihen liitettyihin laitteisiin pääsyn seurantaa ja kontrollointia, sekä pääsyä siihen tietoon, jota on kerätty verkon laitteista osana verkonhallintaa. Esimerkiksi lokeihin kerätyt tiedot ovat hyvin tärkeä osa turvallisuuden hallintaa. Turvallisuuden hallinta keskittyy siihen, kenellä on oikeus päästä eri laitteisiin ja niiden palveluihin. Yritysten tietoturvallisuuspolitiikassa on määritelty turvallisuuden hallinta. (Jaakohuhta 2005: 310-311)

2.6 Verkonhallinnan protokollat

1970 -luvun lopulla verkot kasvoivat pienistä ja hajanaisista lähiverkoista suuriksi ja massiivisiksi verkoiksi. Näitä suuria verkkoja kutsuttiin interneteiksi ja ne kasvoivat edelleen kovaa vauhtia. Verkkojen kova kasvuvauhti aiheutti kuitenkin ongelman hallittavuudessa, jolloin uusi protokolla kehitettiin verkonhallintaan.

2.6.1 SNMP

SNMP (Simple Network Management Protocol) on nykyinen de facto -standardi TCP/IP (Transmission Control Protocol / Internet Protocol) -pohjaisten verkkojen verkonhallintaan. Sen ensimmäinen versio hyväksyttiin vuonna 1989, jolloin sitä pidettiin yleisesti tilapäisenä verkonhallintaprotokollana, joka otettiin käyttöön vain väliaikaisesti, kunnes suuremmat ja paremmat verkonhallintaprotokollat otettaisiin käyttöön. SNMP on valmistajariippumaton ja se kehitetty Yhdysvalloissa yhteistyössä USA:n armeijan, teollisuuden ja akateemisen yhteisön kanssa. (Jaakohuhta 2005:312)

SNMP määritettiin vuonna 1989 TCP/IP -verkonhallintaan ja siitä tuli nopeasti standardi. Se oli kuitenkin liian rajoittunut. 1991 SNMP:tä laajennettiin sisäänrakennetulla RMON (Remote Network Monitoring) 1991 määrittelyillä. RMON määrittelee algoritmit ja tietokannat että (ip-verkkojen) hallintaan sekä verkkoon liitettyjen laitteiden hallintaan. RMONia laajennettiin vuonna 1995 ja 1997 tuli RMON2. SNMPv2 julkistettiin 1993 ja sitä laajennettiin 1995. SNMPv3 julkaistiin 1998 (Stallings 1999:604)

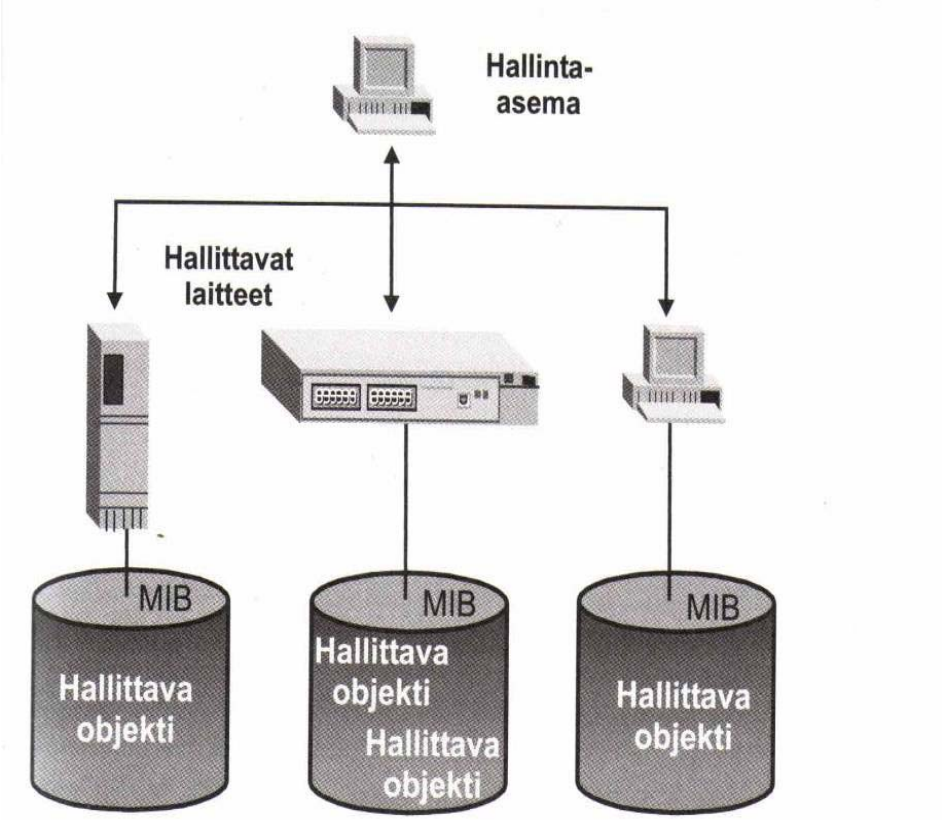
SNMPv1:ssä käytetään salasanaa ja tarkistetaan lähettäjän IP-osoite. Salasana kulkee kuitenkin selväkielisenä, ja lähettäjän IP-osoitteen väärentäminen ei ole erityisen vaikeaa. Siksi versio 1:n Set-viestit ovat harvoin käytössä, ja SNMPv1:tä käytetään enemmänkin verkon valvontaan kuin varsinaiseen hallintaan.

SNMPv2:ta suunnitellessa haluttiin parantaa turvallisuutta ja siinä suunniteltiin käytettävän salausta ja autentikointia. Toteutuksesta tuli kuitenkin raskas ja huono ja sen tilalle kehiteltiin erilaisia viritelmiä. Näin saatiin erilaisia SNMPv2:n toteutuksia, esim. SNMPv2*, SNMPv2c ja SNMPv2u. SNMPv2 ei ole täysin yhteensopiva SNMPv1:n kanssa, joskin yrityksiä niiden yhteiskäyttöön on tehty. (Haikonen & kumppanit 2000)

SNMPv3:sen suunnittelussa on kiinnitetty huomiota turvallisuuteen. SNMPv3 sisältää useita suojauksiin liittyviä ominaisuuksia, joiden asetukset voidaan määrittää itsenäisesti. Se tukee esimerkiksi sanomien todennusta, yksityisyyttä, sekä valtuuksien tarkastusta, jolla varmistetaan, että verkonhallitsijalla on tarvittavat käyttöoikeudet. (Comer 2002:572)

Kuvassa 1 on esitelty SNMP -hallintaympäristön rakenne. SNMP on asiakas-/palvelinprotokolla. SNMP – protokollan omalla terminologialla asiakkaalla tarkoitetaan manageriohjelmistoa verkkoa hallitsevassa asemassa NMS (Network Management Station). Palvelin on niin sanottu agenttiohjelmisto, joka sijaitsee verkossa olevassa hallittavassa laitteessa MNE (Managed Network Entity). Agenttiohjelmiston teh-

tävä on kerätä tietoa hallittavasta objektista ja välittää tieto verkon manageriohjelmistolle, joka taas kerää tiedot kaikista hallittavista laitteista, joihin agenttiohjelmisto on asennettu ja esittää tiedot raporttimuotoisena. (Pöllönen 2006)



Kuva 1. SNMP-hallintaympäristön rakenne (Jaakonhuhta 2005: 313)

SNMP protokollan asiakas ja palvelin ohjelmistot käyttävät tiedonsiirtoon UDP -protokollaa ja sen porttia 161. Poikkeuksena mainittakoon trap – viestit, jotka kulkevat portin 162 kautta. Toiminnaltaan SNMP on kysely-/vastausprotokolla. Tiedonsiirto toimi niin, että manageriohjelma kysyy agentilta tarvittavaa tietoa, johon agentti vastaa. Poikkeuksena ovat agentin lähettämät trap – viestit, jotka ilmaisevat joitakin ennalta määritellyn tapahtuman tapahtumista valvottavassa laitteessa. Trap – viestit määritellään asettamalla seurattavan verkkolaitteen SNMP – agenttiohjelmistoon seurattavalle resurssille halutut rajaarvot. Jos nämä arvot ylittyvät, agenttiohjelmisto lähettää manageriohjelmistolle tiedon trap – viestillä. Viestejä kutsutaan nimellä PDU (Protocol Data Unit). (Pöllönen 2006)

Hallintaohjelmistot ja agentit kommunikoivat rajoitetun operaatiojoukon kautta, joihin viitataan nimellä primitiivit. Näitä primitiivejä käytetään pyyntöjen tekemiseen ja informaation lähettämiseen kahden koneen välillä (kuva 2). Hallintaohjelmisto luo seuraavat primitiivit:

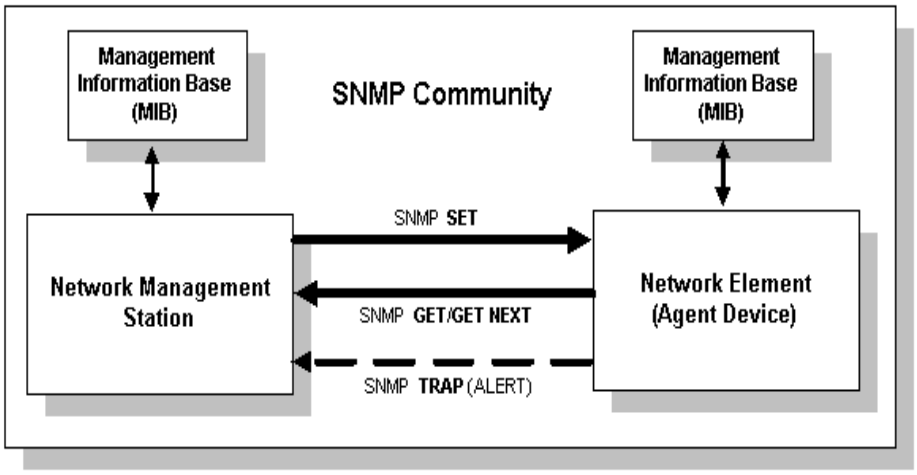
get – Hallintaohjelmisto käyttää tätä primitiiviä noutaakseen agentilta yhden tiedonpalan.

get-next – Kun tieto, jonka hallintaohjelmisto haluaa noutaa agentilta, koostuu useammasta kuin yhdestä osasta, esimerkiksi taulukot, tätä primitiiviä käytetään noutamaan tietosarja.

Set – Hallintaohjelmiston voi tällä primitiivillä pyytää etälaitteelta ajettavaa agenttia asettamaan tietyn muuttujan tietylle arvolle.

get response – Tätä primitiiviä käytetään vastaamaan hallintaohjelmiston get -tai get-next -pyyntöön.

trap- Vaikka SNMP -vaihtoliikenteen aloittaa yleensä hallintaohjelmisto, agentin täytyy toisinaan informoida hallintakomponenttia jonkin tärkeän tapahtuman vuoksi. Siinä tapauksessa käytetään tätä primitiiviä.(Ogletree 2001:47)



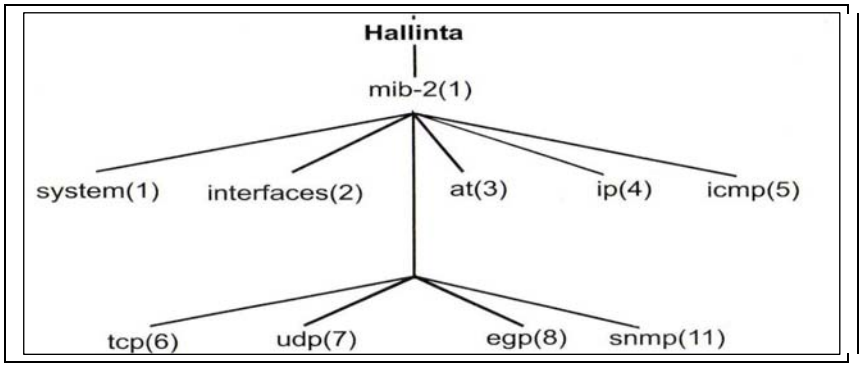
KUVA 2. SNMP-primitiivien kommunikointi (wtcs.org)

2.6.2 MIB

MIB (Management Information base) on SNMP:n määrittämä hallintatietokanta joukolle objekteja (olioita). MIB:in hierarkkinen osoiteavaruus sisältää yksittäisen osoitteen jokaiselle objektille. Objektilla tarkoitetaan mitä tahansa tarkkailtavaa/hallittavaa kohdetta (esimerkiksi porttia, protokollatilastoja, vastaanotettujen ja lähetettyjen pakettien määrää). MIB on määritelty SMI:n (Structure of Management Information) standardin mukaisesti. Agentti ylläpitää hallintatietokantaa, josta se kerää tiedot hallintatyöaseman kyselyihin. MIB mahdollistaa valvontaohjelmiston sekä agenttien välisen täsmällisen kommunikoinnin, mutta tämä edellyttää, että molemmilla on samanlaiset MIB-rakenteet.(Puska 2000: 309)

MIB-rakenteen objektit ovat usein laskureita, joilla on numeerinen arvo. Esimerkkinä ipInReceives, joka laskee sisään tulleiden ippakettien määrän siitä lähtien, kun valvontaohjelmisto käynnistetty. Valvontaohjelmisto pyytää ipInReceives-tietoja agentilta MIB-osoittella .iso.org.dot. internet.mgnt.mib.ip.ip- InReceives ja käytännössä viittaus tapahtuu sitä vastaavalla numeerisella osoitteella.1.3.6.1.2.4.3. MIB:in puumaista rakennetta luetaan ylhäältä alaspäin. (Casad & Wilsey 1999: 294)

Kuvassa 3 on esiteltynä puumaisen MIB:in rakenne. Puun yläosan ominaisuudet ovat ISO:n (International Standard Organization) määrittelemiä. Puun alemmat tasot määrittelevät muut organisaatiot ja laitetoimittajat. Puun ylin taso sisältää objektiryhmiä kuten System (kuvaa laitteet), Interfaces (kuvaa verkkoliitäntöjen liikenteen), IP (IP-pakettien tilastot), ICMP (ICMP-viestit), TCP (TCP-liikennetilastot, UDP (UDP-liikennetilastot, EGP (EGP-tilastot) ja SNMP (SNMP-liikennetilastot). RMON on MIB:in laajennus, joka kerää hallintatietoa yksittäisestä verkon segmentistä. (Jaakonhuhta 2000: 274.)



KUVA 3. Puumaisen MIB:in rakenne (Jaakohuhta 2005: 315)

2.6.3 RMON

RMON on kehitetty vuonna 1995 korvaamaan SNMP:n puutteita ja sitä käytetäänkin yhdessä SNMP:n kanssa. Molemmat, sekä SNMP että RMON käyttävät agentteja, joita kutsutaan yleisesti nimellä tiedonkeruuyksikkö (probe). Se on ohjelmisto tai erillinen laite, joka on sijoitettu verkon laitteeseen keräämään tietoa verkkoliikenteestä ja tallentaa keräämänsä tiedot tiedostoon (MIB). SNMP-hallinta-aseman on koko ajan käytävä tutkimassa SNMP-agentteja toisin kuin RMON-tapauksessa. Siinä kerättyä tietoa voi myöhemmin tarkistella historiaan perustuen. Tiedon kerääminen ei ruuhkauta verkkoa, vaan kerätty informaatio voidaan välittää kokonaisuutena hallinta-asemille.

RMON voidaan sijoittaa lähes mihin tahansa verkon laitteeseen ja yleisemmin sen löytää kytkimestä, reitittimestä tai moniporttitoistimesta. Verkon hallinta-aseman ei jatkuvasti tarvitse tutkia keruuyksiköitä, vaan vasta sen perusteella onko keruuyksiköllä jotakin kerrotta-

vaa. Tämän mukaan keruuyksiköt ovat palvelimia ja hallinta-asemat asiakkaita asiakas-palvelin suhteessa. SNMP:n rooli tässä mallissa on välittää tietoa RMON-keruuyksikön ja hallinta-aseman välillä. (Jaakohuhta 2005: 316-317)

2.6.4 CMIP

CMIP (Common Management Information Protocol) kehitettiin 1980-luvulla korvaamaan SNMP – protokolla. Protokollan toteuttaminen osoittautui kuitenkin niin vaativaksi, että sen kehittäjä pystyy tarjoamaan sitä vain rajoitetussa muodossa. CMIP suunniteltiin niin, että se pystyisi paikkaamaan SNMP:n puutteita sekä rajoituksia. CMIP perustuu PDU:n kuten SNMP – protokollakin, mutta CMIP:llä PDU:ita on 11 kun taas SNMP:llä niitä on viisi. (Turunen & Leppälahti 2000)

CMIP:tä käytetään enimmäkseen telekommunikaatiopuolella, ja se on ITU:n (The International Telecommunication Union) hyväksymä protokolla. Lähiverkkojen pohjautuessa suurimmassa osin TCP/IP – protokollapinoon, siinä olevat laitteet tukevat lähinnä SNMP:tä. Tämä on suurin syy siihen, ettei CMIP:stä tullut kovinkaan suosittua protokollaa lähiverkkoihin. Toinen tekijä CMIP:n harvinaisuuteen lähiverkoissa on sen suuri resurssien käyttö, sekä protokollan monimutkaisuus. (Leinwald & Conroy 1996: 190)

3 NAGIOKSEN ESITTELY JA KONFIGUROINTI

Nagios on avoimeen lähdekoodiin perustuva verkonvalvontaohjelmisto, joka on suunniteltu toimimaan Linux ja Unix-käyttöjärjestelmissä. Nagios perustuu GPL-lisenssiin (General Public License), joka tarkoittaa sitä, että lisenssin alainen ohjelmisto on kenen tahansa levitettävissä sekä muokattavissa.

Nagios oli alun perin nimeltään NetSaint, mutta se muutettiin myöhemmin Nagiokseksi. Nagioksen kehitti henkilö nimeltä Ethan Galdstad. Karl DeBisschop, Subhendu Ghosh, Ton Voon, ja Stanley Hopcroft ovat kehittäneet taas Nagioksessa käytettävät pluginit. Plugineista kerrotaan myöhemmin tässä työssä. Nagios on hyvin suosittu verkonhallintaohjelma, jota yritykset käyttävät ympäri maailman.

Nagioksen avulla pystytään valvomaan laitteita jotka ovat liitetty verkkoon, kuten työasemat, palvelimet, kytkimet, reitittimet ja tulostimet. Nagios jakaa valvottavat kohteet palvelimiin (Hosts) ja palveluihin (Services). Palvelimet ovat fyysisiä laitteita, kuten työasemat, kun taas palvelut ovat palvelimiin liitettyjä palveluita, kuten FTP. Nagios verkonvalvontaohjelma on apuväline verkon valvonnassa, virheiden huomaamisessa sekä niiden ennalta ehkäisyssä. Lisäksi Nagiokseen voidaan määrittää toiminto, jolla se lähettää viestin verkonvalvojan kännykkään tai sähköpostiosoitteeseen virheen havainnoidessaan. (Nagios: About Nagios.)

Nagios valvoo muun muassa seuraavia asioita:

- HTTP, SMTP, PING, SSH, NNTP, FTP, DNS, POP, Telnet.
- Palvelimen resurssit, kuten levyn ja muistin kulutus, prosessit, varmistukset, lokitiedostot, toiminnanohjaus, kulunvalvonta.
- Ympäristötekijät, kuten sensorit.

Nagios mahdollistaa mm. seuraavia asioita:

- Nagioksessa on täysin avoin plugin-tekniikka, joka sallii omien palvelin- ja palvelutarkastusten teon sekä kehityksen.
- Luo Real-time-ilmoituksen virheistä (sähköposti, tekstiviesti) nimetyille vastuuhenkilöille tai -ryhmille.

- Voidaan määritellä komentoja, jotka ajetaan vikatilanteiden ilmettyä (uudelleenkäynnistykset).

- Web-seurantaan voidaan määrittää oikeuksia niille, ketkä pääsevät seuraamaan palvelujen tiloja.

3.1 Ohjelmistovaatimukset

Nagioksen alustana täytyy olla Linux tai jokin muu Unix-variantti. Käytettäessä CGI:tä, täytyy alustalle asentaa ja konfiguroida Web-palvelin. Suositelluin on Apache, mikä on myös yleisin HTTP-palvelinohjelma. Linux-asennuksessa on toivottavaa asentaa täydellinen asennus, jotta Nagioksen toimivuus olisi hyvä.

3.2 Asennus

Asennus aloitetaan kirjautumalla root- käyttäjätunnuksella serverille. Tämän jälkeen voi luoda kansion, johon Nagios asennetaan esimerkiksi **mkdir -p /usr/src/backs/nagios**. Tämän jälkeen asennuspaketti haetaan osoitteesta www.nagios.com/download. Komennolla **wget** http:// keihanna.dl.sourceforge.net/sourceforge/nagios/nagios-2.3.1.tar.gz asennuspaketti ohjautuu koneelle. Seuraavaksi mennään siihen hakemistoon, mihin asennuspaketti on ladattu ja puretaan paketit komennolla **tar -zxvf /usr/src/backs/nagios/nagios-2.3.1.tar.gz**.

Kun paketti on purettu ajetaan konfigurointiskripti ***./configure --prefix=prefix--with-cgiurl=cgiurl --with-htmurl=htmurl --with-nagios-user=someuser--with-nagios-grp=somegroup./configure*.** Nagios käännös tehdään komennolla **make all** sekä asennetaan binäärit ja html-tiedostot komenolla **make install**. Lopuksi muodostetaan inid-skripti joka toteutetaan komenolla **make-inid**.

3.3 Konfigurointi

Kun Nagios-paketti on asennettu oikeaan hakemistoon, on seuraavaksi vuorossa Nagioksen peruskonfigurointi. Nagios ei pysty valvomaan mitään, ellei sille määritä, mitä sen pitäisi valvoa. Tarkastettavat kohteet tarvitsevat tarkastustiedoston tarkastusten tekemiseksi. Nagioksessa on mahdollista käyttää mallipohjaista tiedostopohjaa, jolloin tarvitsee vain muuttaa tiedostoon muuttujat. Tässä muutamia tarkastuksia selityksineen:

/etc/nagios/services.cfg

```
define service{

host_name                       localhost, kone
service_description             PING
is_volatile                       0
check_period                    24x7
max_check_attempts              3
normal_check_interval           10
retry_check_interval            1
contact_groups                  linux-admin
notification_interval           240
notification_period             24x7
notification_options            w,u,c,r
check_command                   check_ping
    }
```

Edellä olevaan tarkastustiedostoon on määritelty palvelu jota nagios-palvelin käyttää. Host_name-kohtaan lisätään ne koneet, joihin kyseinen palvelin palvelun kohdistaa. Service_description kohtaan lisätään palvelun kuvaus, joka on tässä tapauksessa PING. Is_volative-kohtaan voidaan määrittää, onko palvelu epävakaa. Check_period-kohtaan lisätään tarkastuksen aikajaksot. Max_check_attemps-kohtaan lisätään uudelleen tarkastusten lukumäärää ennen kuin siitä lähetetään ilmoitus. Normal_check_interval-kohtaan lisätään tarkastusten suoritusväli. Retry_check_interval-kohtaan voidaan määrittään tarkastusten suoritusväli, jos tila on OK (Tilatyypit käsitellään myöhemin). Contact_groups-kohtaan lisätään kontaktiryhmät. Notification_interval-kohtaan voidaan määrittää aika, jonka kuluttua lähetetään uusi ilmoitus. Notification_period-kohtaan määritetään ilmoitusten lähettämisen ajanjakso. Notification_options-kohtaan lisätään tilat, joissa ilmoitukset lähetetään. Check_command on tarkastuskomento (määritetään plugineissä).

etc/nagios/hosts.cfg

```
define host{
host_name                       localhost
alias                           nagios server
address                         127.0.0.1
check_command                   check-host-alive
max_check_attempts              10
notification_interval           60
notification_period             24x7
notification_options            d,u,r
contact_groups          l       inux-admin
    }
```

```
define host{
 host_name                      kone
 alias                          workstation1
 address                        192.168.0.200
 check_command                  check-host-alive
 max_check_attempts             10
 notification_interval          120
 notification_period            24x7
notification_options             d,u,r
contact_groups                  linux-admin
                        }
```

Edellä olevaan tarkastustiedostoon on määritelty hostit, joita Nagios tarkastaa.

/etc/nagios/contactgroups.cfg

```
define contactgroup{
contactgroup_name               linux-admin
alias                           testi
members                         admin
                                }
```

Edellä olevaan tarkastustiedostoon on määritelty kontaktiryhmä. Tämä on ryhmä, joka saa tarkastustulokset Nagiokselta. Ryhmän nimi on linux-admin ja ryhmän jäsenenä on admin.

/etc/nagios/contact.cfg

```
define contact{
contact_name                    nagiosadmin
alias                           linux-guru
service_notification_period     24x7
host_notification_period        24x7
service_notification_options    c,r
host_notification_options       d,r
service_notification_commands   notify-by-email
host_notification_commands      host-notify-by-email
email                           nagiosadmin@netum.fi
                                }
```

Edellä olevaan tarkastustiedostoon on määritelty kontaktihenkilöt. Tämä henkilö saa tarkastustulokset Nagiokselta. Tarkastukset on määritelty tulemaan sähköpostiviestinä kontaktihenkilön sähköpostiin.

/etc/nagios/timeperiod.cfg

```
define timeperiod{
timeperiod_name                 24x7
alias                           24 Hours A Day, 7 Days A Week
sunday                          00:00-24:00
monday                          00:00-24:00
tuesday                         00:00-24:00
wednesday                       00:00-24:00
thursday                        00:00-24:00
friday                          00:00-24:00
saturday                        00:00-24:00
                                }
```

Tässä määritellään, millä periodilla tarkastukset suoritetaan. Yllä olevassa tarkastustiedostossa tarkistukset tehdään joka päivä 24 tunnin ajan. Vaihtoehtona tälle periodille olisi työaikaperiodi johon määritellään aika maanantaista perjantaihin klo 8- 16.

/etc/nagios/checkcommands.cfg

```
define command{
command_name                    check_ping
command_line                    /usr/lib/nagios/plugins/check_ping
                                -H
                                $HOSTADDRESS$ -w $ARG1$ -
                                c $ARG2$ -p 5
                                }
```

Tässä määritellään tiedosto joka käyttää ping_check-komentoa tarkistaakseen, onko yhteys etäkoneeseen kunnossa.

3.4 Tilatyypit

Tarkastettavien kohteiden tilat (tilatyypit) muodostuvat tiloista joita ovat mm. OK, WARNING, CRITICAL sekä näiden tilojen eri tyypeistä. Itse tilatyypit ovat nimeltään HARD ja SOFT. Nämä tilatyypit ovat Nagioksen peruspilarit, koska niiden perusteella Nagios lähettää ilmoituksia. Esimerkkinä voisi mainita define host-konfiguraatiotiedoston, jossa on kohta max_check_attempts. Tähän määritellään uusintatarkastusten määrä, jos tarkastusten tuloksena on jokin muu kuin OK. Uusintatarkastusten ansiosta voidaan välttää virheellisten ilmoitusten lähettäminen, jos ilmoitus johtuu jostain muusta kuin tarkastuksen kohteesta.

Kuten edellä mainittiin, tilat ovat HARD ja SOFT. SOFT -vikatila on tila, jossa tarkastuksen tuloksena on saatu jokin muu kuin OK, eikä uusintatarkistuksia ole vielä tehty. SOFT -tila on elpymistila, jossa tarkastuksen kohde on palannut toimimaan vikatilasta, mutta ei vielä tehnyt kaikkia uusintatarkastuksia. HARD -tila on tila, jossa uusintatarkastukset on tehty, mutta muutoksia ei ole tapahtunut. (Nagios: State Types.)

3.5 Apachen asennus

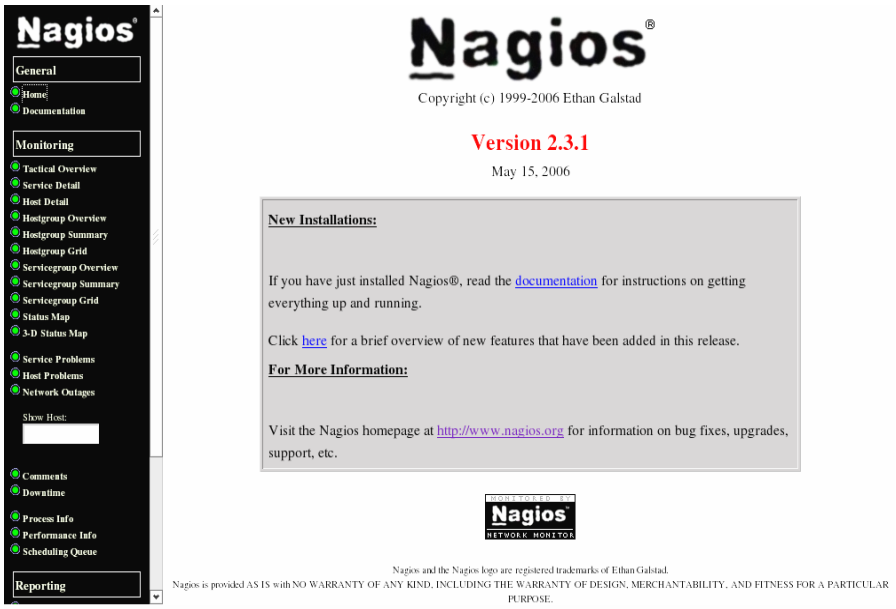
Jotta Nagioksen saa toimimaan oikein ja näyttämään tarkistukset selaimen kautta, on asennettava web-palvelin. Helpoin ja luultavasti vaivattomin ratkaisu on valita Apache. Apache tulee nykyisin melkein kaikkien Linux-jakelujen mukana.

Apachen konfigurointitiedostoon **etc/http/conf/httpd.conf** täytyy tehdä lisäyksiä ja muutoksia. Cgi:hin tarvitsee lisätä ScriptAlias, jotta se toimii oikein.

```
ScriptAlias /nagios/cgi-bin "/usr/lib/nagios/cgi"
<Directory "/usr/local/nagios/sbin">
   Options ExecCGI
  AllowOverride None
  Order allow,deny
  Allow from all
</Directory>

Alias /nagios "/usr/share/nagios"
<Directory "/usr/local/nagios/share">
# SSLRequireSSL
   Options None
   AllowOverride None
   Order allow,deny
   Allow from all
</Directory>
```

Aliaksen lisäyksen jälkeen käynnistetään Apache uudestaan komennolla
service httpd restart. Kun Apache on käynnistetty, selaimen osoiteriville kirjoitetaan **http://localhost/nagios**. Tämän jälkeen selaimen pitäisi näyttää Nagiosen pääsivu (kuva 4).



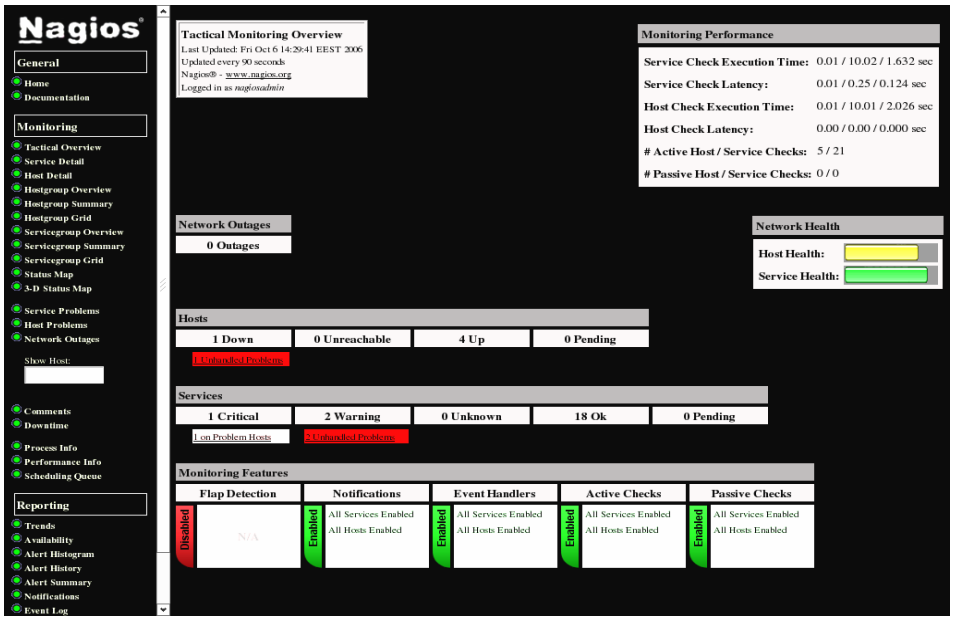
KUVA 4. Pääsivu

3.6 Nagios käyttöliittymä

Nagios käyttöliittymä on selainpohjainen. Käyttöliittymän avaamiseksi tarvitsee avata selain ja kirjoittaa osoiteriville http**://nimi/nagios/ tai http://ip-osoite/nagios/**. Nagioksen etusivun pitäisi näin avautua näytölle. Etusivulla käy ilmi, mikä Nagioksen versio on asennettu sekä sivun vasemmalla puolella oleva dokumentaatiolinkki sekä linkit monitorointiin ja raportointeihin.

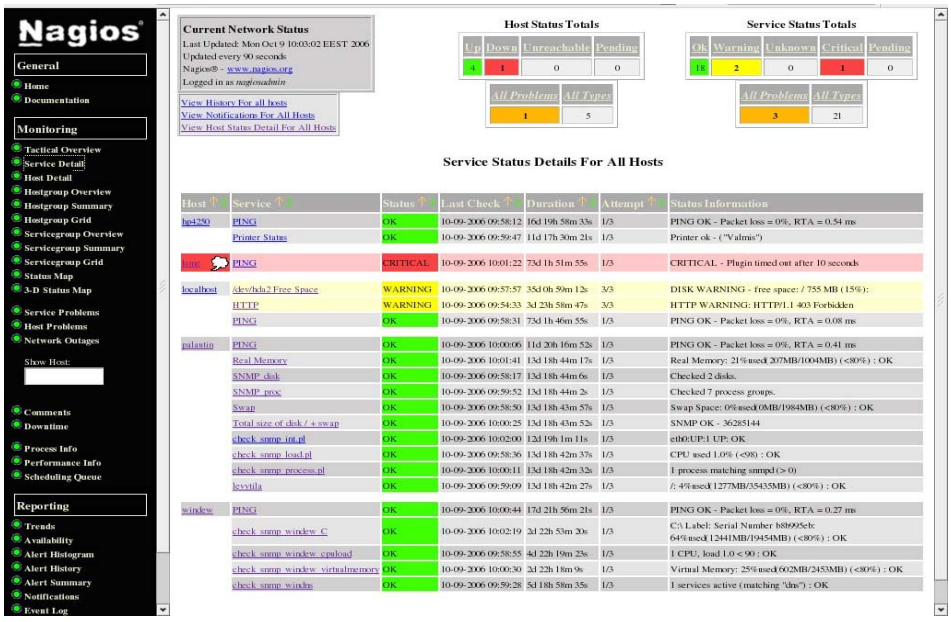
3.6.1 Monitoring

Tactical Overview näyttää kokonaisvaltaisesti verkon tilanteen (kuva 5). Tässä tilanteessa viisi palvelinta (hosts) on monitoroituna ja neljä niistä on UP -tilassa. Yksi palvelin on alhaalla (down), mutta tavoitteottamattomissa (unreachable) olevia palvelimia ei ole. Näytössä näkyy myös, että yksi palvelu (services) on kriittisessä tilassa (critical), kaksi on varoitustilassa (warning), ja 18 on OK.



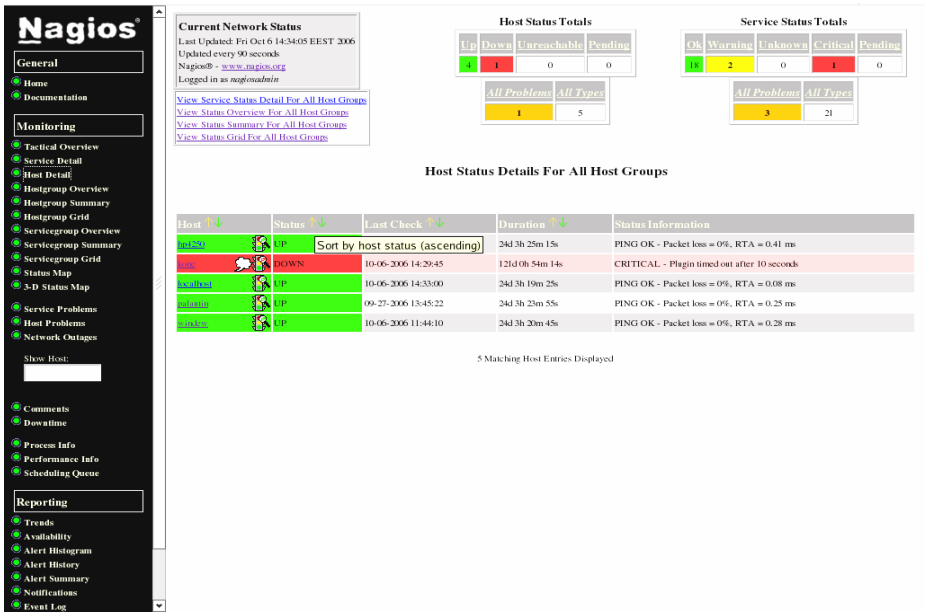
KUVA 5 Tactical Overview

Service details näyttää kaikki palvelut (services), mitä palvelimille (hosts) on määritelty sekä näyttää palvelun tilan (kuva 6).



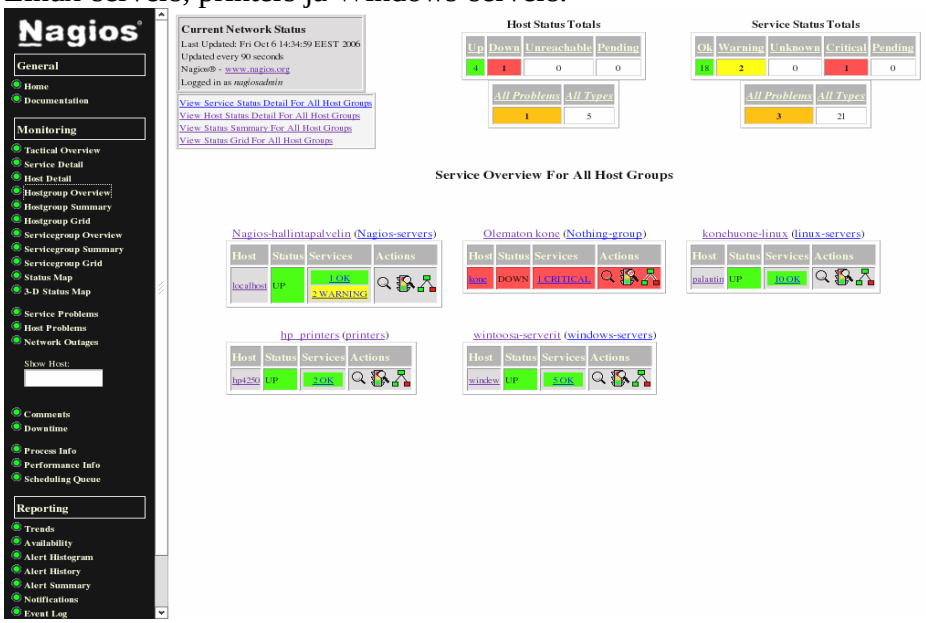
KUVA 6. Service Details.

Host detail näyttää tämänhetkisten Hostien (palvelinten) tilan. Tässä (kuva 7) esimerkissä Hostit ovat nimeltään hp420, kone, localhost, palautin ja windew. Kone nimisen palvelimen tila on Down ja loppujen tila on UP. Hosteille on määritelty PING-tarkastus, joka tarkastaa, että hostiin saadaan yhteys. Kone nimiselle hostille ei saada yhteyttä, joten sen tila on Down.



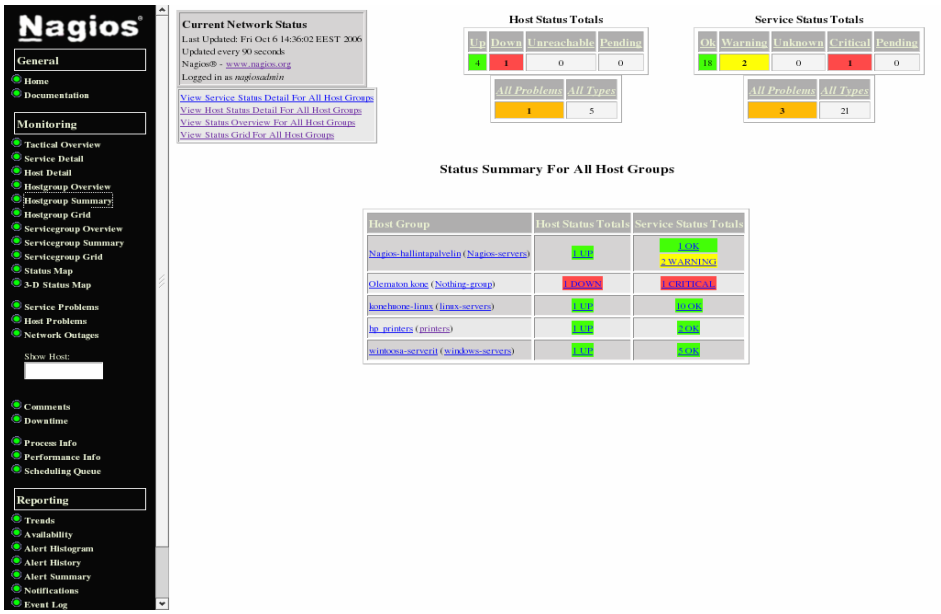
KUVA 7. Host Detail

Hostgroup overview näyttää palvelimet ryhmiteltynä. (Kuva 8) Esimerkissä hosteilla on viisi ryhmää, Nagios-servers, Nothing-group, Linux-servers, printers ja Windows-servers.



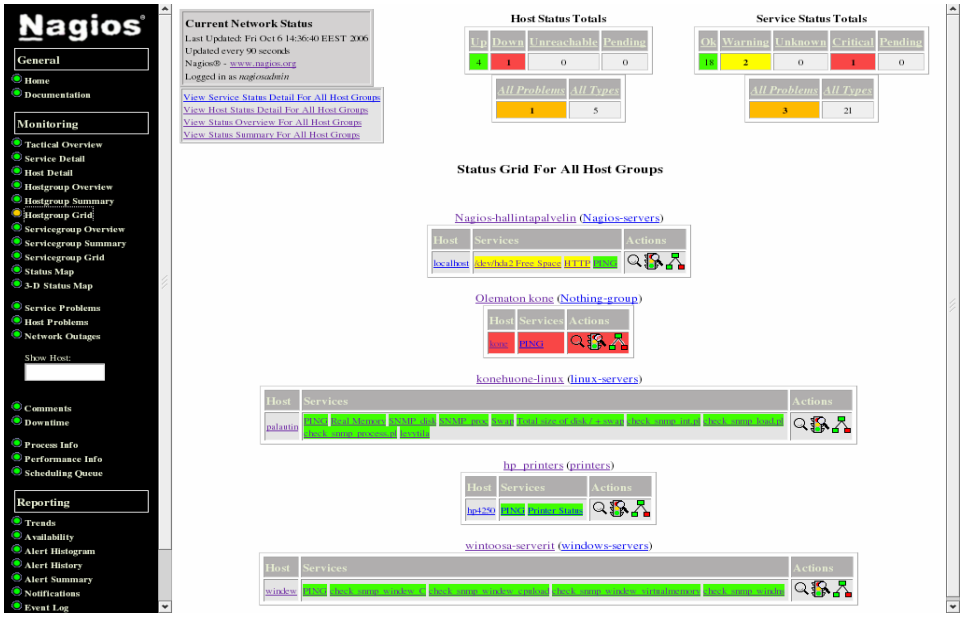
KUVA 8. Hostgroup Overview

Hostgroup Summary (kuva 9) näyttää yhteenvedon ryhmistä ja ryhmään kuuluvista palvelimista ja palveluista.



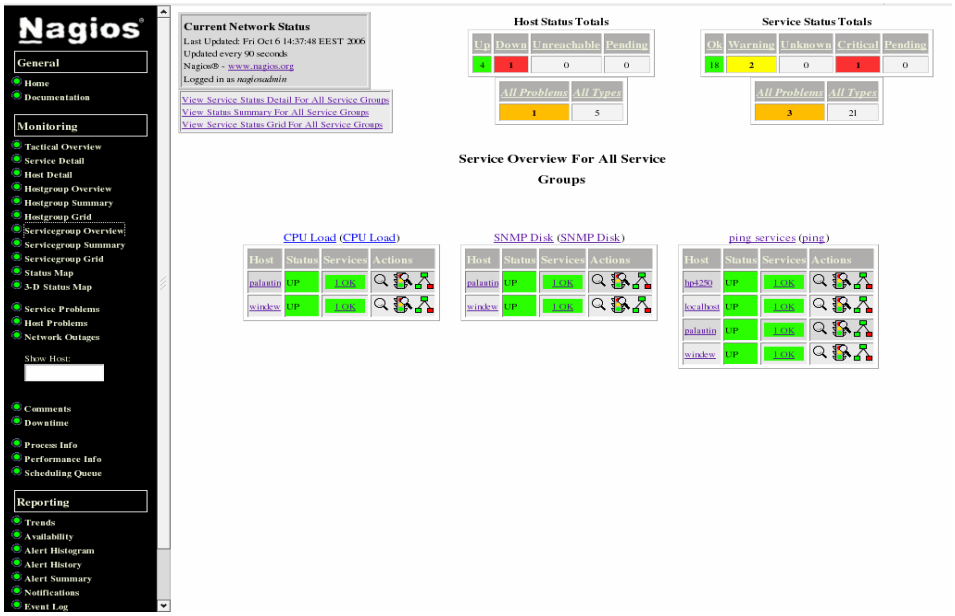
KUVA 9. Hostgroup Summary

Hostgroup Grid (kuva 10) näyttää muuten samanlaiselta kuin Hostgroup Summary, mutta Hostgroup Grid näyttää jokaiseen ryhmään kuuluvan koneen tiedot erikseen.



KUVA10. Hostgroup Grid

Servicegroup Overview (kuva 11), Servicegroup Summary ja Servicegroup Grid toimivat periaatteessa samoin kuin hostgroup-linkit. Servicegroup-linkit näyttävät palvelut ryhmiteltynä, joita voidaan tarkastella myös jokainen palvelu erikseen.



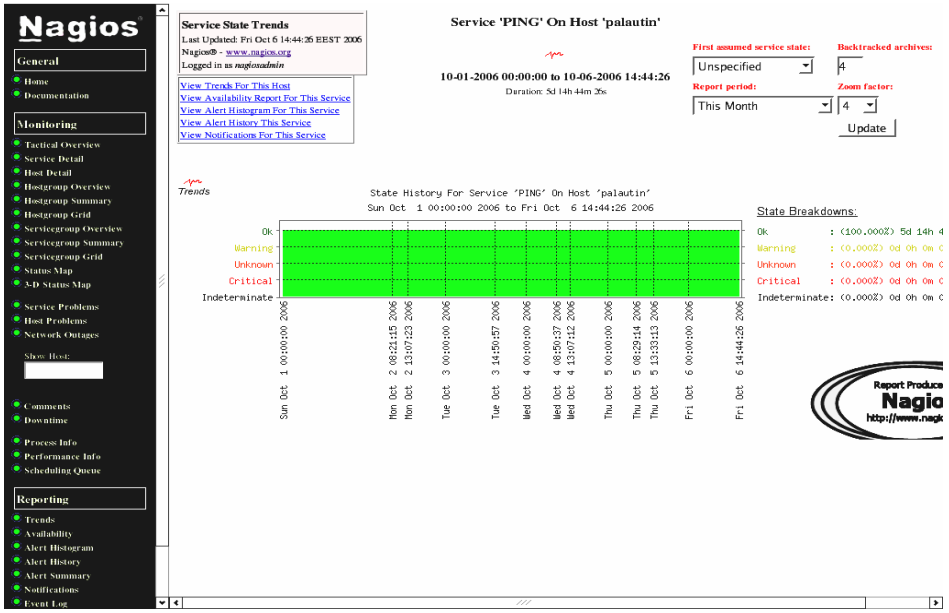
KUVA11. Servicegroup Overview

Statusmap esittää graafisin keinoin valvottavan verkon kuvan. Tämä auttaa hahmottamaan valvottavien kohteiden määrää. Väriä on käytetty havainnollistamaan laitteiden palvelut ja tilat. Service Problems ja Host Problems esittävät ongelmat koneilla ja palveluilla. Nämä mahdolliset ongelmat nähdään myös mm. Service Detail- ja Servise Host-linkeistä, joista mainittiin edellä. Comments-linkistä käyttäjä voi kirjoittaa kommentteja Palveluille ja Palvelimille. Downtime-linkistä voidaan ajastaa palvelin tai palvelu sammumaan haluttuun aikaan. Process Info antaa tiedon Nagios-prosessista, esim. kuinka kauan prosessi on ollut päällä, koska se kytkettiin päälle, viimeisin tarkastusaika ym. Performance Info antaa informaatiota palvelinten sekä palveluiden aktiivisista ja passiivisista tarkistuksista.

Monitoring-linkkiosion viimeinen kohta on Scheluding Queue, joka ilmoittaa ajan palveluiden ja palvelinten viime tarkistuksesta sekä seuraavasta tulevasta tarkistuksesta.

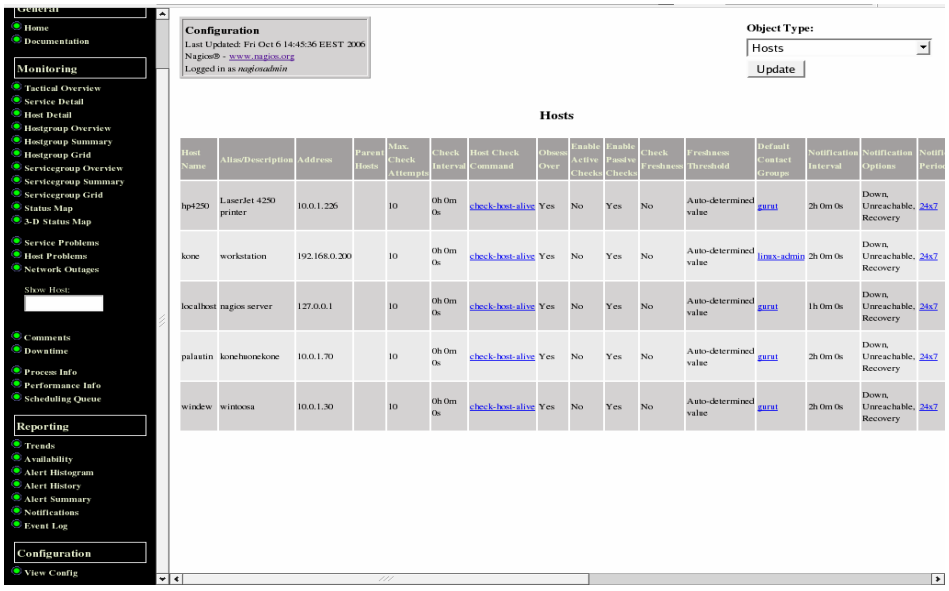
3.6.2 Reporting

Trends -työkalun avulla käyttäjä voi tutkia palveluita ja palvelimia, mahdollisia hälytyksiä, lokeja sekä luoda näistä raportteja (kuva 12).



KUVA12. Trends

Wiew Config-linkin kautta voi selata palvelinten ja palveluiden konfiguraatiotietoja (kuva 13). Tämä on helppo polku katsoa tiedostojen konfiguraatio. Toinen vaivalloisempi tapa on tutkia tiedostoja Nagios -serverin tiedostoista.



KUVA13. Wiew Config

4 NAGIOS TARKASTUKSET

Tarkastuksiin Nagios käyttää erilaisia plugineja. Itse Nagioksen asennuspaketissa ei kyseisiä lisäohjelmia ole, joten ne täytyy ladata erikseen. Kuten edellä on tullut selväksi, Nagios tarkastaa palveluita (services) ja palvelimia (hosts), jotka määritetään niiden konfiguraatiotiedostojen avulla. Tarkastuslisäohjelmat ovat Nagioksen tavoin ilmaisia, koska ne perustuvat GPL -lisenssiin. Tarkastusohjelmia kutsutaan plugineiksi.

4.1 Pluginit

4.1.1 Asennus

Nagios-pluginien asennus tehdään melko samalla tavalla kuin Nagios. Asennuspaketti haetaan osoitteesta www.nagios.com/ download. Valitaan kohta Official Nagios Plugins ja ladataan wget-komennolla oikea pluging-paketti koneeseen. Paketti puretaan ja asennetaan. Paketissa olevat pluginit asentuvat automaattisesti libexec -hakemistoon. Paketissa oleellisempia plugineja ovat mm. check_ping, check_http, check_disk, check_dns, check_load, check_users, check_tcp.

4.1.2 Ominaisuudet

Check_ping lähettää ICMP ECHO-reguest paketin, johon määritellyt laitteet vastaavat ECHO REPLY paketilla. Tällä komennolla testataan onko paketin vastaanottava osapuoli verkkoyhteyksien päässä, eli jos laite vastaa lähettäjän check_ping komentoon, laite on verkossa.

Check_http tarkistaa vastaanottavan palvelimen http-palvelun tilan. Kyseisellä toiminnolla voidaan tarkistaa http- sekä https-palvelimia.

Check_disk plugin palauttaa valvottavasta koneesta käytetyn levytilan koon prosentuaalisen arvon sekä varoittaa, jos arvo menee tietyn rajan yli.

Check_dns tarkistaa palvelimen dns-palvelun toimivuuden käyttämällä nslookup-kyselyä, jotta se hakee annetulle domain-nimelle vastaavan ip-osoitteen.

Check_load tarkistaa paikallisen palvelimen prosessorin kuormaa. Tarkistetaan palvelimen ns. keskimääräistä kuormaa ja jos kuorman ylin sallittu kuorma ylittyy, Nagios antaa ilmoituksen siitä.

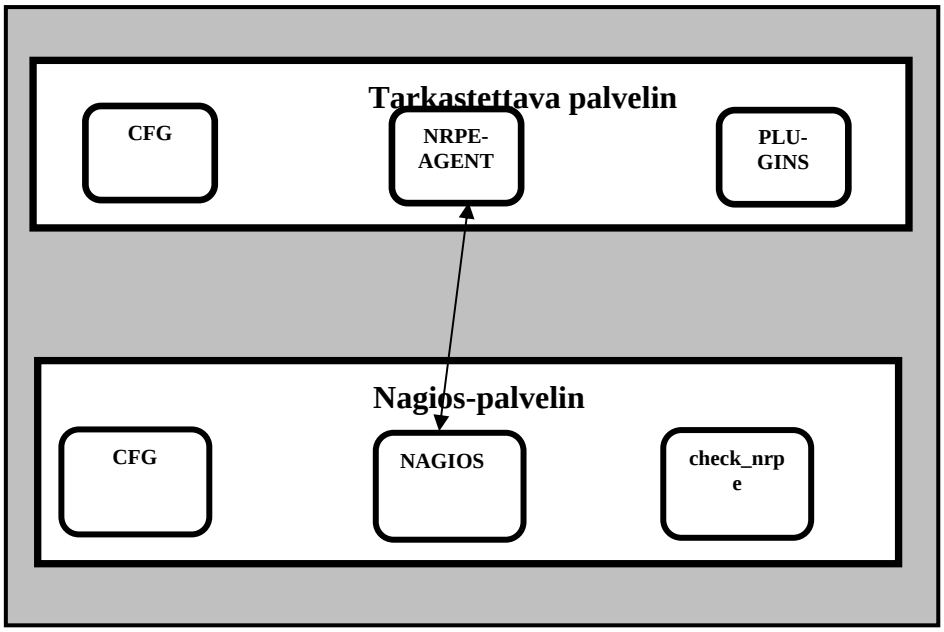
Check_users seuraa valvottavan koneen käyttäjämääriä. Jos esimerkiksi koneelle on kirjautumisia enemmän kuin on sallittu, Nagios antaa ilmoituksen siitä. Käyttäjämäärä on määriteltävä konfiguraatiotiedostoon.

Check_tcp tarkistaa tcp-yhteyksiä, eli tämä plugin tarkistaa tarkistettavan koneen osoitteen sekä porttinumeron.

4.2 NRPE

4.2.1 Ominaisuudet ja asennus

NRPE, joka on lyhenne sanoista Nagios Remote Plugin Executor, on Nagiokseen kuuluva aktiivinen lisäosa, joka mahdollistaa tietojen haun valvottavilta koneilta. NRPE:n avulla voidaan valvottavilta koneilta tarkkailla mm. prosessorin kulutusta, levynkäyttöä sekä muistinkulutusta. Nagios-palvelimeen on asennettava check nrpe-plugin tiedosto sekä koneisiin joita halutaan tarkkailla, asennetaan nrpe-agentti ja nrpe-plugin-tiedosto. NRPE toimintaprosessi on seuraava: Nagios antaa suorituskäskyn check nrpe-nimiselle tiedostolle, joka lähettää tiedot verkon yli valvottavan koneen NRPE-prosessiin. Kun NRPE -prosessi on suorittanut tarkastuksen, tarkastuksen tulos lähetetään takaisin Nagios palvelimelle (kuva 14).



KUVA14. NRPE -prosessi

4.2.2 NRPE Linux

NRPE agentti ei sisälly normaaliin Nagios -plugins pakettiin, joten se pitää ladata valvottavalle koneelle, jota Nagios -palvelin “kutsuu” omien NRPE_pluginien (check_nrpe) avulla. Linuxille tarkoitetun NRPE-agenttin voi ladata osoitteesta http://www.nagiosexchange.org /NRPE. 77.0.html ?&tx_netnagext_pi1[p_view]=126. Valvottavalle koneelle asennetaan vielä tarvittavat pluginit, jotka voivat olla Nagioksen perus -plugin paketti. Paketin voi ladata osoitteesta http://www.nagios.org /download/.

NRPE:n tietoturva on vielä tällä hetkellä kysymysmerkki, mutta käyttämällä niin sanottua TCP-Wrapper-toimintoa, voidaan tietoturvaa parantaa huomattavasti. Etäkoneelle voi määrittää sovelluskohtaisesti NRPE:lle yhteyden sallittuihin osoitteisiin sekä kieltää yhteydet. Sallitut yhteydet määritetään host-allow-listalla ja kielletyt yhteydet määritetään host-deny-listalla. Esimerkki kuinka tämä tehdään yhdellä tavalla:

Oletetaan että käyttöjärjestelmä on niin uusi, että siinä on xinetd-palvelu. Tällöin ei erillistä TCP wrapper:ia tarvitse asentaa, koska xinetd jo sisältää sen ominaisuuden. Luodaan nrpe- tiedosto etc/xinetd.d hakemistoon, joka sisältää seuraavia asioita:

```
# default: on
# description: NRPE
service nrpe
{
flags                   = REUSE
socket type             =stream
wait                    = no
user                    = <user>
server                  = <nrpebin>
server_args             = -c <nrpecfg> --inetd
log_on_failure          += USERID
disable                  = no
only_from                = <ipaddress1> <ipaddress2> ...
                        }
```

User kohtaan vaihdetaan NRPE käyttäjän nimi.
<nrpebin> kohtaan vaihdetaan polku, jossa sijaitsee nrpe-binääritiedostot.
<nrpecfg> kohtaan vaihdetaan polku, jossa sijaitsee nrpe-konfiguraatiotiedostot.
ipaddress1 ja ipaddress2 kohtaan vaihdetaan niiden koneiden ip-osoitteet, joilla on oikeus ottaa yhteyttä NRPE palveluun.

Tämän jälkeen käynnistetään xinetd-palvelu komennolla service xinetd restart, jotta lisätyt konfiguraatiot tulevat voimaan.

Jotta Nagios-kone osaa ottaa yhteyden NRPE-agenttiin, Nagios-koneeseen on lisättävä vielä checkcommands- sekä service-tiedostoon seuraavat tiedot, jotka vastaavat NRPE-palvelua.

```
define command{
command_name                    check_nrpe
command_line                    /usr/local/nagios/
                                plugins/check_nrpe-
                                H$HOSTADDRESS$ -c $ARG1$
                                }

define service{
host_name                       someremotehost
service_description             someremoteservice
check_command                   check_nrpe
                                }
```

4.2.3 NRPE Windows

Valvottaville Windows koneille on oma agenttinsa, jonka nimi NRPE_NT. Sen voi ladata osoitteesta http://www.miwidv .com /nrpent/. Myös Windowsille on tarkoitettu omat pluginit ja ne voi ladata osoitteesta http://www.nagiosexchange.org/NRPE_Plugins .66.0.html. Tämä nimenomainen paketti sisältää seuraavia tarkistuksia:

- cpuload_nrpe_nt.exe (prosessorin kuormitus)
- diskspace_nrpe_nt.exe (levytilan haluttu koko)
- eventlog_nrpe_nt.exe (logien tarkkailu)
- memload_nrpe_nt.exe (muistin kuormitus)
- service_nrpe_nt.exe (haluttujen palveluiden päällä olo)

Avataan agentin mukana tullut nrpe.conf- tiedosto ja lisätään sen allowed.host-kohtaan Nagios-koneen IP-osoite. Command definitions-kohtaan lisätään, mitä tarkastetaan esimerkiksi check_cpu. Tämän lisäksi lisätään nrpe.conf-tiedostoon komento:

command[check_cpu]=c:\WINDOWS\system32\cscript.exe //NoLogo //T:10 C:\NRPE\bin\check_cpu.wsf /w:20 /c:10

Tämän jälkeen siirrytään komentokehotteeseen käynnistämään nrpe-agentti palveluksi. Se tehdään komennolla: **net start nrpe_agent.**

4.2.4 NRPE Nagios -palvelin

Kuten aiemmin kävi ilmi, on Nagios-palvelimeen asennettava nrpe-plugin_paketti. Paketin voi ladata osoitteesta http://nagios.org / download/. Paketin asennuksessa nrpe- pluginit menevät automaattisesti samaan hakemistoon muiden pluginien kanssa. Seuraavassa on esimerkki, kuinka checkcommand.cfg, service.cfg ja escalations.cfg tiedostoja konfiguroimalla saadaan haettua levytilan tarkastus nrpe-lisäosaa käyttämällä.

```
define command{
command_name          check_disk
command_line          /usr/lib/nagios/plugins/check_nrpe -H
$HOSTADDRESS$ -c check_disk
                                        }
```

Tähän checkcommand-tiedostoon on lisätty check_disk, joka määrittelee nrpe:lle toiminnon, mitä tietoa haetaan. Tässä tapauksessa haetaan etäkoneen levytilatietoja. Check_nrpe -pluginilla otetaan yhteyttä etäkoneeseen ja sen nrpe –agenttiin, joka palauttaa haetut tiedot.

```
define service{

 host_name                      etakone
 service_description            NRPE_disk
 is_volatile                    0
 check_period                   24x7
 max_check_attempts             3
 normal_check_interval          10
 retry_check_interval           1
 contact_groups                 NRPE_koneet
 notification_interval          240
 notification_period            24x7
 notification_options            w,u,c,r
 check_command                  check_disk
     }
```

Service.cfg -tiedostoon lisätään, mitä palveluita ja mitkä koneet tai ryhmät palveluita käyttävät. Service_description kohdan pitää täsmätä escalations.cfg:n Service_description kohdan kanssa, jotta nimenomaiset haut toteutuvat nrpe:n avulla.

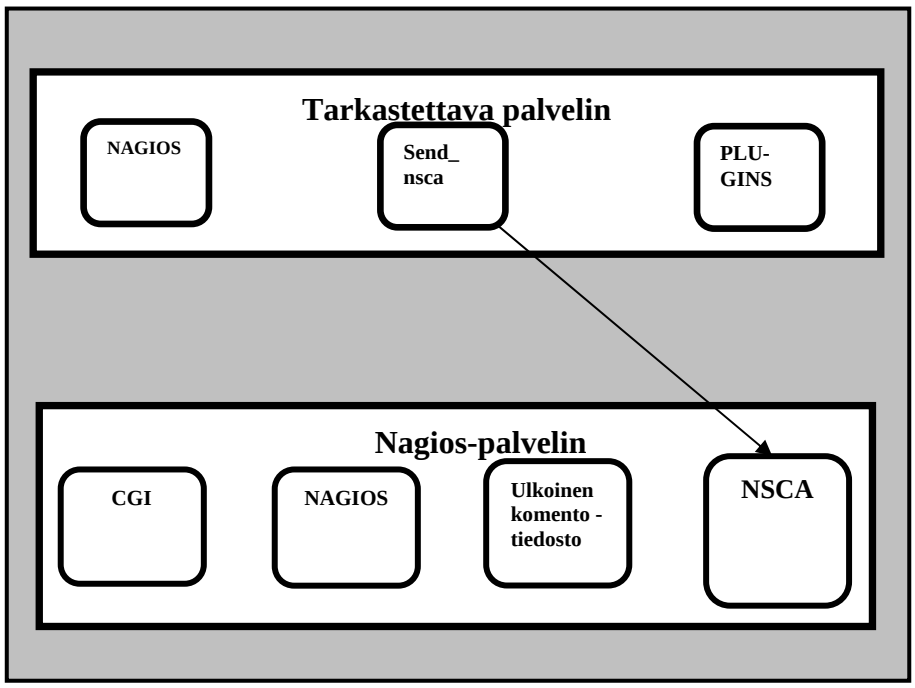
```
define serviceescalation {

host_name             etakone
service_description   NRPE_disk
}
```

4.3 NSCA

4.3.1 Ominaisuudet

NSCA on lyhenne, joka tulee sanoista Nagios Service Check Acceptor. Nimensä mukaan Nagios-palvelin pelkästään vastaanottaa tarkastusten tuloksia eikä kysele niitä etäpalvelimilta (kuva 15). Tätä kutsutaan myös passiiviseksi tarkistukseksi. NSCA:n avulla voidaan valvottavilta koneilta tarkkailla mm. prosessorin kulutusta, levynkäyttöä sekä muistinkulutusta. NSCA:ta käytetään eniten redundantissa valvonnassa sekä valvonnan jakamisessa, joita käsitellään enemmän kappaleessa 9. Valvonnan kehittämisratkaisuja.



KUVA15. NCSA -prosessi

4.3.2 NSCA Linux

NSCA:ta käytettäessä on valvottavalle etäkoneelle asennettava send_nsca -client, Nagios ja tarvittavat pluginit. Pluginit suorittavat etäpalvelimella tarkastuksia, joitten jälkeen send_nsca -client lähettää ne Nagios -palvelimelle. Checkcommand- tiedostoon on lisättävä submit_check_result -skripti, joka lähettää tarkastuksesta saadun tilatyypin Nagios -serverille. Tämä on nimeltään niin sanottu OCSP-käsky (Obsessive Compulsive Service Processor).

```
define command{
command_name                    submit_check_result
command_line                    usr/lib/plugins/eventhandlers/
```

submit_check_result $HOST-
NAME$
$SERVICEDESC$ $SER-
VICES-
STATE$ $OUTPUT$ |
$PERFDATA$

Tässä on esimerkki submit_check_result -tiedostosta, johon ei tarvitse muuttaa kuin Nagios -serverin osoite.

```
#!/bin/sh
# Arguments:
# $1 = host_name (Short name of host that the service is
# associated with)
# $2 = svc_description (Description of the service)
# $3 = state_string (A string representing the status of
# the given service - "OK", "WARNING", "CRITICAL"
# or "UNKNOWN")
# $4 = plugin_output (A text string that should be used
# as the plugin output for the service checks)
#

# Convert the state string to the corresponding return code
return_code=-1

case "$3" in
OK)
return_code=0
;;
WARNING)
return_code=1
;;
CRITICAL)
return_code=2
;;
UNKNOWN)
return_code=-1
;;
esac

# pipe the service check info into the send_nsca program, which
# in turn transmits the data to the nsca daemon on the central
# monitoring server

/bin/echo "$1\t$2\t$return_code\t$4\n" |
/export/home/nagios/bin/send_nsca 10.0.1.98 -c
/export/home/nagios/etc/send_nsca.cfg
```

4.3.3 NSCA Nagios

Tarkistukset otetaan vastaan Nagios-palvelimella olevalla nsca -lisällä, josta se kirjoittaa ne ulkoiseen komentotiedostoon. Nagios-prosessi konfiguroidaan tarkastamaan tätä komentotiedostoa tietyin väliajoin. Nagios -palvelimelle konfiguroidaan samat host - ja services -tiedot kuin etäkoneellekin. Nagios.cfg- tiedostoon lisätään accept_passive_host check, sekä check_external_commmands komennot.

Oletetaan, että käyttöjärjestelmä niin uusi, että siinä on xinedt-palvelu, jonka alla NSCA-palvelu toimii. Konfiguraatio on samanlainen kuin NRPE:ssä. Tämän jälkeen käynnistetään xinedt-palvelu service xinetd restart.

4.4 Itse tehdyt tarkastusohjelmat

Perl- kielellä voidaan toteuttaa räätälöityjä tarkastusohjelmia Nagiokseen. Perl-ohjelmointikielen käyttö on suotavaa tarkastusten teossa, koska sillä saadaan aikaan suorituskyvyltään nopeita tarkistuksia.

Itse tarkistukset perustuvat paluuarvoihin, joita ohjelma ehtolauseke tulostaa. Tässä on esimerkkinä taulukkomuodossa tilatyyppien ja paluukoodien suhde.

Paluukoodi	Status	Tila
0	Tarkistus on suoritettu ja kaikki on OK	OK
1	Tarkistus on suoritettu mutta kohde ei toiminut kunnolla	WARNING
2	Tarkistus on suoritettu mutta kohde ei toiminut ollenkaan	CRITICAL
3	Tarkistus ei pysty suorittamaan tehtävää, koska tarkistusohjelmalle on annettu vääriä parametreja	UNKNOW

Taulukko1. Paluukoodit

Alla on esimerkki, kuinka muodostetaan ohjelmakoodi tarkistuksia varten.

```
if (ehtolause1) {
print “$tila : status1” ;
                exit $ERRORS {'OK'};
            }
```

```
elseif (ehtolause2) {
print “$tila : status2” ;
        exit $ERRORS {'WARNING'};
        }

elseif (ehtolause3) {
print “$tila : status3” ;
        exit $ERRORS {'CRITICAL'};
        }
elseif (ehtolause4) {
print “$tila : status4” ;
        exit $ERRORS {'UNKNOWN'};
        }
```

4.5 SNMP -tarkastukset

4.5.1 Ominaisuudet

SNMP on TCP/IP-verkkojen hallinnassa käytetty protokolla, mutta sillä pystytään myös tekemään tarkistuksia Nagioksen avulla. SNMP on suosittu Nagioksen tekemissä tarkistuksissa, koska sen tietoturva-ominaisuudet ovat hyvät. Uudemmat SNMP:tä käyttävät tarkastuspluginit tukevat SNMPv3-ominaisuuksia.

SNMP mahdollistaa etäpalvelimen resursseja tarkastavan toiminnon. Sitä varten etäpalvelimelle on asennettava agentti, jolloin se toimiin rajapintana verkon ja tarkastettavien kohteiden välillä. SNMP:n avulla pystytään tarkkailemaan mm. prosessorin kulutusta, levynkäyttöä sekä muistinkulutusta ym. SNMP:llä on puumainen rakenne MIB, jossa jokaisella kohteella on yksilöllinen tieto. Tätä tieto kutsutaan myös OID:ksi. OID on numeerinen arvo MIB:stä. Kuten aikaisemmin on käynyt ilmi, SNMP käyttää tarkastuksissaan luku- ja keskeytysominaisuutta. Keskeytysomaisuuden (trap) avulla palvelin voi vikatilanteessa lähettää hälytyksen Nagiokselle. Tämä pitää tosin konfiguroida palvelimelle erikseen.

4.5.2 SNMP- agentti

Agentti on hallinnan kohteena oleva laite tai siihen asennettu ohjelma, joka vastaa hallintalaitteelta tuleviin kyselyihin. Agentin tehtävänä on kerätä tietoa sen verkkosegmentin alueelta, johon se on asennettu toimimaan. Kerätty tieto tallennetaan omaan hallintatietokantaan, josta hallintatyöasema voi noutaa kerätyn tietopaketin analysoitavaksi. Agentti on kuin mikä tahansa muu palvelu tietokoneen ollessa käynnissä. Kun agentti käynnistetään, se ryhtyy keräämään tietoa määritettyjen asetusten mukaisesti.

Agentti voi toimia myös tiedonvälittäjänä toisessa verkkosegmentissä olevalle agentille. Tämän toiminnon ideana on mahdollistaa eri standardien väliseen kommunikaatioon kykenemättömät ohjelman osat ymmärtämään toisiaan. Välittävä agentti muuntaa sisään tulevat viestit kohdeagentin ymmärtämään muotoon ja vastaukset taas hallintaaseman ymmärtämään muotoon. Etenkin suurissa verkoissa voidaan hyödyntää hierarkiaa, jossa hallintaohjelma kommunikoi vain muutaman agentin kanssa, jotka keräävät tietoa muutaman agentin alla sijaitsevalta olevan verkon solmulta. Näin verkonhallinta kuormittaa verkkoa tasaisemmin.

Agentti osaa vastata seuraaviin neljään komentoon, joilla voidaan käsitellä tietokantaa:

get lukee agentin MIB -tietokannasta yhden tiedon.
get-next lukee agentin MIB -tietokannasta järjestyksessä seuraavan tiedon.
walk lukee agentin MIB -tietokannasta kaikki tiedot tietystä alkukohdasta lähtien.
set käskee agentin antamaan arvon konfiguroitavalle parametrille, tai nollaamalla tietyn tiedon, esimerkiksi lähteneiden datapakettien laskurin.

Agentin toiminta riippuu verkonvalvojan tarpeista. Agentti voi toimia seuraavanlaisesti.

- Valvontaohjelmisto lähettää agentille pyyntöjä ja agentti vastaa niihin.
- Agentti palauttaa vastauksen tiettyyn osoitteeseen saatuaan get- tai getnext-komennon.
- Agentti lähettää walk-komennolla useita vastauksia selatessaan MIB-tietokantaa alaspäin annetusta alkukohdasta.
- Agentti vastaanottaa set-komentoja valvontaohjelmistolta ja asettaa raja-arvoja verkon valvontaa varten.
- Agentti voi ilman pyyntöä esimerkiksi seurata, ylittyikö jokin set-komennolla asetettu raja-arvo. Kun agentti on havainnut rajan ylityksen, se lähettää sanoman valvontaohjelmistolle.

4.5.3 SNMP Linux

SNMP-agentin voi ladata osoitteesta http://dag. wieers.com /packages /perl-Net-SNMP/. Kun kyseessä on RPM- paketti, asennus tapahtuu

komennolla **rpm -ivh ”paketin_nimi”**. Net –SNMP-agentti on ajattavana tiedostona hakemistossa /etc/rc.d/intt.d/snmpd. Avaamalla snmpd tiedoston, saadaan selville mihin kansioihin SNMP:n binäärit sekä konfiguraatiotiedostot kopioidaan. Tässä tapauksessa binääritiedostot sijaitsevat //usr/sbin/snmp -hakemistossa, joten sen voi määrittää vaihtamalla prog-riville oikean polun. Paketin mukana tulee EXAMPLE.conf tiedosto, jota voi käyttää agentin konfiguroinnissa. Tiedosto asentuu automaattisesti kansioon /usr/share/doc/net-snmp-5.3.1/ EXAMPLE.conf. Tiedoston voi nimetä snmp.conf -nimiseksi ja kopioida se kansioon /etc/snmp/ snmp.conf. Alla on esimerkki tiedostojen poluista.

```
#!/bin/sh
#
# snmpd This shell script takes care of starting and stopping
# the net-snmp SNMP daemon
#
# chkconfig: - 26 74
# description: snmpd is net-snmp SNMP daemon.

# Source function library.
. /etc/rc.d/init.d/functions

# Source networking configuration.
. /etc/sysconfig/network

# Check that networking is up.
[ ${NETWORKING} = "no" ] && exit 0

RETVAL=0
name="snmpd"
prog="//usr/sbin/snmpd"

[ -x $prog -a -f /etc/snmp/snmpd.conf ] || exit 0

start() {
        # Start daemons.
        echo -n $"Starting $name: "
        daemon $prog
        RETVAL=$?
        echo
        [ $RETVAL -eq 0 ] && touch /var/lock/subsys/$name
        return $RETVAL
}

stop() {
        # Stop daemons.
        echo -n $"Shutting down $name: "
```

```
        killproc $prog
        RETVAL=$?
        echo
        [ $RETVAL -eq 0 ] && rm -f /var/lock/subsys/$name
        return $RETVAL
}

# See how we were called.
case "$1" in
  start)
        start
        ;;
  stop)
        stop
        ;;
  status)
        status $name
        RETVAL=$?
        ;;
  restart|reload)
        stop
        start
        RETVAL=$?
        ;;
  condrestart)
        if [ -f /var/lock/subsys/$name ]; then
           stop
           start
           RETVAL=$?
        fi
        ;;
  *)
        echo $"Usage: $0 {start|stop|restart|condrestart|status}"
        exit 1
esac

exit $RETVAL
```

SNMP -agentin saa toimiminaan muutamalla peruskonfiguraatiolla, jotka lisätään SNMP.conf -tiedostoon. Lisäykset voi kirjoittaa joko valmiina oleviin string-kohtiin, tai lisätä ne vapaasti tiedostoon.

rocommunity public
syslocation konehuone
syscontact nagiosadmin@netum.fi

Rocommunity public-komennolla määritetään versioille SNMPv1 ja SNMPv2 read-only-oikeus tarkistuksiin. Syslocation-komennolla ker-

rotaan koneen fyysinen sijainti, johon agentti on asennettu. Syscontact-komennolla kerrotaan koneen kontakti henkilö. Tämän jälkeen SNMP-palvelun (agentin) voi käynnistää komennolla **service snmpd start**.

Palveluiden automaattista käynnistymistä voidaan säätää chkconfig -työkalulla. Eli jos ja kun halutaan SNMP -palvelun käynnistyvän automaattisesti silloin kun kone käynnistyy uudestaan, käytetään komentoa chkconfig –level 3 snmp on. Tarkistuksen siitä, että SNMP- palvelu on päällä koneen käynnistyessäkin, voi tehdä komennolla chkconfig –list.

4.5.4 Agentin testaus

Agentin toimivuutta voi testata eri SNMP -työkaluilla, kuten snmpwalk ja snmpget. Snmpwalk -työkalulla saadaan ruudulle kaikki mahdollinen tieto koneen MIB-groupista. Tässä on esimerkki snmpwalk -komennosta ja tulosteesta.

#snmpwalk -v2c -c public localhost | less
SNMPv2-MIB::sysDescr.0 = STRING: Linux palautin 2.6.9-34.EL #1 Wed Mar 8 00:07:35 CST 2006 i686
SNMPv2-MIB::sysObjectID.0 = OID: SNMPv2-SMI::dod.0.0.0.0.0.0.0
DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks: (1303017) 3:37:10.17
SNMPv2-MIB::sysContact.0 = STRING: Me <nagiosadmin@netum.fi>
SNMPv2-MIB::sysName.0 = STRING: palautin
SNMPv2-MIB::sysLocation.0 = STRING: konehuone
SNMPv2-MIB::sysORLastChange.0 = Timeticks: (0) 0:00:00.00
SNMPv2-MIB::sysORID.1 = OID: IF-MIB::ifMIB
SNMPv2-MIB::sysORID.2 = OID: SNMPv2-MIB::snmpMIB
SNMPv2-MIB::sysORID.3 = OID: TCP-MIB::tcpMIB
SNMPv2-MIB::sysORID.4 = OID: IP-MIB::ip
SNMPv2-MIB::sysORID.5 = OID: UDP-MIB::udpMIB
SNMPv2-MIB::sysORID.6 = OID: SNMP-VIEW-BASED-ACM-MIB::vacmBasicGroup
SNMPv2-MIB::sysORID.7 = OID: SNMP-FRAMEWORK-MIB::snmpFrameworkMIBCompliance
SNMPv2-MIB::sysORID.8 = OID: SNMP-MPD-MIB::snmpMPDCompliance
SNMPv2-MIB::sysORID.9 = OID: SNMP-USER-BASED-SM-MIB::usmMIBCompliance
SNMPv2-MIB::sysORDescr.1 = STRING: The MIB module to describe generic objects for network interface sub-layers
SNMPv2-MIB::sysORDescr.2 = STRING: The MIB module for SNMPv2 entities

SNMPv2-MIB::sysORDescr.3 = STRING: The MIB module for managing TCP implementations
SNMPv2-MIB::sysORDescr.4 = STRING: The MIB module for managing IP and ICMP imple

Tulosteesta käy selville mm. koneen nimi (palautin), käyttöjärjestelmä (Linux 2.6.9-34.EL), kontaktihenkilö (nagiosadmin@netum.fi) ja fyysinen sijainti (konehuone). Snmpget -työkalulla voidaan saada tietoja vaikkapa vain koneen fyysisestä sijainnista:

snmpget -v2c -c public localhost sysLocation.0
SNMPv2-MIB::sysLocation.0 = STRING: konehuone

4.5.5 SNMP Nagios-server

Hallintalaite on jokin työasema tai palvelin verkossa, johon valvontaohjelmisto halutaan asettaa käyttöön. Työasema valvontasoftineen mahdollistaa haettujen tietojen ja saapuneitten tiedotteiden perusteella reaaliaikaisten tilanäkymän esittämisen. Hallintalaite on tässä tauksessa Nagios -palvelin. Nagios -palvelimelta voi myös tehdä tarkistuksia manuaalisesti komentorivin kautta. Tällöin voi käyttää snmpget -työkalua. Snmpget -työkalulla voidaan tehdä tarkistuksia OID-osoitteiden perusteella. Kuten aikaisemmin kävi selväksi, jokaisella kohteella on yksilöllinen arvo ja OID on numeerinen arvo MIB:stä. Tässä on esimerkki snmpget-komennosta ja sen tulosteesta:

snmpget -v2c -c public 10.0.1.70 .1.3.6.1.4.1.2021.4.5.0
UCD-SNMP-MIB::memTotalReal.0 = INTEGER: 1027668

Snmpget -työkalulla kysytään koneelta, jonka ip-numero on 10.0.1.70, RAM-muistin määrää. V2c tarkoittaa SNMP versio kahta, -c public on community name, joka on salasana, jonka avulla päästään lukemaan tietoja. Numerosarja .1.3.6.1.4.1.2021.4.5.0 on OID-osoite, eli tarkastettavan kohteen yksilöivä tunnus. Tuloste näyttää RAM-muistin määräksi 1027688 kilotavua. Snmpget ja snmpwalk ovat nimenomaan tarkoitettu komentorivityökaluiksi, joita käytetään yleensä silloin kun tarkastuksiin tarkoitettua käyttöliittymää ei ole käytössä. Kun tarkastuksia suorittaa Nagios-sovellus, on sille asennettava tarvittavat SNMP -plugin-paketit. Snmp-plugin-paketit voi ladata yksitellen sivulta http://www. manubulon.com/nagios. Kun kyseessä on Perl-paketti, täytyy antaa suoritusoikeus, jotta pluginit voi ajaa. Suoritusoikeus annetaan komennolla chmod +x ”paketin nimi”. Snmp-pluginit ohjautuvat automaattisesti samaan hakemistoon muiden pluginien kanssa. Seuraavassa on kaksi esimerkkiä, kuinka pluginit voi testata ennen kuin tarkastukset lisätään checkcommand.cfg ja service.cfg -tiedostoihin.

./check_snmp_storage.pl -H 10.0.1.70 -C public -m / -r -w 80 -c 90
/: 4%used(1277MB/35435MB) (<80%) : OK

Tässä on kyseessä check_snmp_storage- plugin. Eli se tarkastaa koneesta, jonka nimi on 10.0.1.70, kuinka paljon kovalevyn tilavuudesta on käytetty. Tulosteesta käy ilmi, että kovalevystä on käytetty neljä prosenttia koko määrästä. Tulosteesta selviää myös kovalevyn käytetty ja kokonaistilavuus megatavuina. Tarkastukseen on lisätty määritykset, että jos kovalevyn käytetty tilavuus ylittää 80 prosenttia, tilatyypiksi tulostuu Warning, ja jos tilavuus ylittää 90 prosenttia, tulostuu näytölle Critical.

./check_snmp_process.pl -H 10.0.1.70 -C public -n snmpd
1 process matching snmpd (> 0)

Tämä plugin tarkastaa onko tarkastettavalla kohteella erikseen määritelty prosessi päällä. Tarkastettavaksi prosessiksi on määritetty SNMP. Tulosteesta selviää SNMP -prosessin olevan päällä.

Kun on tarkistettu että pluginit toimivat komentorivillä, ne voi lisätä Nagioksen tarkistettavaksi. Tämä tapahtuu periaatteessa samalla lailla kuin NRPE- pluginien lisäys.

```
#'check_snmp_storage.pl DISK' command definition
define command{
command_name                    check_snmp_storage_levy
command_line                    /usr/lib/nagios/plugins/
                                check_snmp_storage.pl
                                 -H 10.0.1.70 -C public -m / -
                                r -w 80 -c 90
                                }
```

Checkcommands.cfg-tiedostoon määritetään tarkastuskomento levyntilan tarkastamisesta. Nimenomainen check_snmp_storage -plugin on suunniteltu tarkastamaan eri kohteita ja tapahtumia kuten levytila, muistinkulutus ja Swap. Toisin sanoen kyseistä pluginia voi käyttää moneen eri tarkoituksen yhtä aikaa muuttamalla pluginin muuttujia ja parametreja.

```
#define service  {
use                             generic-service
host_name                       palautin
service_description             levytila
is_volatile                     0
check_period                    24x7
max_check_attempts              3
normal_check_interval           5
retry_check_interval             1
```

```
contact_groups                  gurut
notification_interval           240
notification_period             24x7
notification_options             w,u,c,r
check_command
check_snmp_storage_levy }
```

Service.cfg -tiedostoon lisätään check_snmp_storage_levy –komento, jotta se täsmää checkcommand..cfg:hen tehdyn tarkastuskomennon kanssa.

4.5.6 SNMPv3 tarkastukset

SNMPv3:n avulla tarkastuksia voi tehdä autentikoituna ja salattuna. Tätä varten on lisättävä käyttäjänimi sekä salasana agentin konfiguraatiotiedostoon, jotta pystytään hakemaan tietoa koneen toiminnoista käyttämällä SNMPv3:a. Käyttäjien lisäämisen ajaksi on SNMPD -palvelu pysäytettävä komennolla **service snmpd stop**. Tämän jälkeen lisätään käyttäjätunnukset ja salasanat. Käyttäjätunnus on netadmin ja salasana netadminpassword. Kirjoitetaan seuraava komento:

#net-snmp-config --create-snmpv3-user -ro -a "netadminpassword" netadmin

Tämän jälkeen lisätään hakemisto, johon luodaan käyttäjänimi sekä salasana. Kyseinen tieto salataan DES –salausalkoritmillä:

/var/net-snmp/snmpd.conf: createUser netadmin MD5 "netadminpassword" DES

Annetaan kyseiselle käyttäjälle read-only -oikeudet tietojen tarkastamiseen. Tämä tieto tulee eri kansioon:

/usr/share/snmp/snmpd.conf: rouser netadmin

Kun käyttäjätiedot ovat lisätty tiedostoihin, SNMPD- palvelun voi käynnistää komennolla **service snmpd start**. Katsomalla kansioon, johon luotiin käyttäjätunnus sekä salasana, huomataan että salaus toimii:

cat /var/net-snmp/snmpd.conf

```
.
usmUser 1 3 0x800007e580562c512f61f77443
0x6e657461646d696e00
0x6e657461646d696e00 NULL .1.3.6.1.6.3.10.1.1.2
0x1701cbd1feb64559cf18f81fecb60965 .1.3.6.1.6.3.10.1.2.2
0x1701cbd1feb64559cf18f81fecb60965 ""
engineBoots 1
```

oldEngineID 0x800007e580562c512f61f77443

Käyttäjätunnusta voi testata vaikka snmpget -työkalulla. Haetaan kyseisellä työkalulla käyttäjätunnusta ja salasanaa käyttäen tiedot koneen fyysisestä sijainnista.

snmpget -v 3 -u netadmin -l authNoPriv -a MD5 -A netadminpassword 10.0.1.70 sysLocation.0
SNMPv2-MIB::sysLocation.0 = STRING: konehuone

Snmpget -työkalu kysyy koneelta, jonka ip-numero on 10.0.1.70, koneen sijaintia, v3 tarkoittaa käytettyä SNMP-versiota, -u parametrille kirjaillaan käyttäjätunnus, -l parametriin kirjaillaan turvallisuustaso, -a parametrille lisätään autentikointiprotokolla, sekä -A parametriin lisätään salasana.

Seuraavassa on sama kysely, mutta salasana on kirjoitettu väärin.

snmpget -v 3 -u netadmin -l authNoPriv -a MD5 -A punaniska 10.0.1.70 sysLocation.0
snmpget: Authentication failure (incorrect password, community or key)

Nagios-tarkistuksissa, jossa käytetään SNMPv3:a, tarvitaan pluginit, jotka tukevat SNMPv3:a. Samaiset pluginit, jotka ladattiin sivustolta http://www.manubulon.com/nagios, ja joita käytettiin versioiden kaksi ja kolme tarkistuksissa, toimivat myös versio kolmen tarkistuksissa. Pluginin voi testata ennen kuin tarkastukset lisätään checkcommand.cfg- ja service.cfg -tiedostoihin.

./check_snmp_int.pl -H 10.0.1.70 -l netadmin -x netadminpassword -n eth0 -r
eth0:UP:1 UP: OK

Plugin tarkastaa onko kohdekoneessa verkkokortti ylhäällä. Parametrit -l ja -x edustavat käyttäjätunnusta sekä salasanaa. Plugin toimii moitteettomasti ja tulostaa oikeat arvot. Kun pluginin toimivuus on tarkistettu komentorivillä, sen voi lisätä tiedostoihin.

```
#'check_snmp_int.pl' command definition
define command{
  command_name                  check_snmp_int_v3
  command_line                  /usr/lib/nagios/plugins/
                                check_snmp_int.pl -H 10.0.1.70
                                $USER8$ -n eth0 -r
```

Tarkastuskonfiguraatiossa on käytetty USER-makroa, johon on määritelty käyttäjätunnus sekä salasana. Jotta makroa voidaan käyttää, lisä-

tään resource.cfg -tiedostoon seuraava lisäys: $USER8= -l netadmin -x netadminpassword.

```
define service  {
use                             generic-service

host_name                       palautin
service_description             check_snmp_int.pl
is_volatile                     0
check_period                    24x7
max_check_attempts              3
normal_check_interval           5
retry_check_interval            1
contact_groups                  gurut
notification_interval           240
notification_period             24x7
notification_options             w,u,c,r
check_command                   check_snmp_int_v3
```

4.5.7 SNMP Windows

Windows-koneille tarkoitetun SNMP -agentin voi ladata osoitteesta http://sourceforge.net/project/showfiles.php? group_id=12694& package _id=162885&release_id=431951. Klikkaamalla exe tiedoston auki, asennuksen voi suorittaa graafisesti. Asennuksessa on valittava, mitkä paketit asennukseen haluaa ja mihin kansioon tiedostot menevät. Asennetaan tiedostot C:\usr\bin hakemistoon ja valitaan asennettavaksi vain peruskomponentit ja SNMP -agentti. Kun asennus on saatu suoritettua loppuun, on testattava agentin toimivuus.

C:\usr\bin>snmpd -V
No log handling enabled - turning on stderr logging
NET-SNMP version 5.2.1.2

Tämä jälkeen ajetaan registeragent.bat.

c:\usr>registeragent.bat
Registering snmpd as a service using the following additional options:
. -Lf "C:/usr/log/snmpd.log"
For information on running snmpd.exe and snmptrapd.exe as a Windows
service, see 'How to Register the Net-SNMP Agent and Trap Daemon as
Windows services' in README.win32.
Press any key to continue . . .

Lopuksi käynnistetään palvelu.

C:\>net start "net-snmp agent"
The Net-SNMP Agent service is starting.
The Net-SNMP Agent service was started successfully

Toinen vaihtoehto on asentaa Windows -koneen oma SNMP -palvelu, jolloin erillistä agenttia ei tarvita. Ainakin Windows XP, 2000, ja 2003 -koneissa on tämä lisäominaisuus.

- Klikataan Start -> Settings -> Control Panel ja tuplaklikataan Add/Remove Programs.
- Klikataan Add/Remove Components ja valitaan Management and Monitoring tools -> Details.
- Rastitetaan Simple Network Management Protocol -> OK.
- Klikataan Next ja valitaan polku, mistä asennetaan tarvittavat tiedostot.
- SNMP-palvelu käynnistyy automaattisesti, kun asennus on suoritettu. Tämän jälkeen konfiguroidaan SNMP.

- Klikataan Start -> Settings -> Control panel.
- Valitaan Administrative Tools -> klikataan Services.
- Valitaan SNMP Service -> klikataan hiiren oikealla ja valitaan Properties.
- Valitaan General -välilehti -> valitse Automatic for Startup Type.
- Valitaan Security -välilehti -> klikataan Add kohdassa Accepted community names.
- Valitaan kohtaan Community Rights ”Read Only” -oikeudet.
- Keksitään Community Name -kohtaan nimi, jolla on lukuoikeudet kyseiseen palvelimeen.
- Valitaan miltä koneilta hyväksytään paketit -> valitaan esimerkiksi Accept SNMP packets from any host.

Kun SNMP -palvelu on asennettu Windows – palvelimelle, se voidaan testata Nagios -palvelimen komentorivillä käyttäen Windows – käyttöjärjestelmälle tarkoitettuja plugineja.

./check_snmp_storage.pl -H 10.0.1.30 -C TCPIP -m ^C: -w 80 -c 90
C:\ Label: Serial Number b8b995eb: 64%used(12442MB/19454MB) (<80%) : OK

Tämä plugin tarkastaa Windows – palvelimen käytetyn tilan. Tarkastukseen on lisätty myös määritykset, jos käytetty tila ylittää 80 prosenttia, tulostuu näytölle Warning, ja jos tilavuus ylittää 90 prosenttia, ilmoitus on Critical. Tulosteesta selviää, että kovalevyn tilavuudesta on käytetty 64 prosenttia.

./check_snmp_win.pl -H 10.0.1.30 -C TCPIP -n dns
1 services active (matching "dns") : OK

Tämä plugin tarkastaa onko Windows – palvelimen DNS – palvelu päällä.

4.5.8 SNMP -verkon aktiivilaitteet

Ainakin Ciscon ja HP:n kytkimistä saadaan selville monenlaista tietoa käyttämällä SNMP:tä. Kytkimistä voidaan tarkkailla esimerkiksi verkkoliikenteen määrää, porttien tiloja, tuulettimen toimivuutta, prosessorin kuormitusta, muistin määrää ja sen kulutusta. Tarkastuksissa voidaan käyttää valmiita plugineita tai laitteiden yksilöllisten osien OID -tunnusta. Valmiita plugineita on verkkolaitteille olemassa rajallinen määrä, joten OID -tunnusten ja check_snmp -pluginin käyttö on yleisempää. Esimerkkinä on Cisco Catalyst 3550 multilayer kytkin, jossa lisätään SNMP:n perusasetukset päälle.

Switch(config)# snmp-server community public

Tällä komennolla Nagios -serveriltä on lukuoikeus kytkimen kaikkiin objekteihin, käyttämällä public -salasanaa. Eli toisin sanoen lisäämällä kyseisen komennon, voidaan tarkastukset tehdä joko komentoriviltä tai nagioksen kautta plugineja käyttämällä. Seuraavassa on esimerkki kuinka check_snmp_load.pl -pluginia käyttämällä saadaan Ciscon kytkimen tai reitittimen prosessorin keskimääräinen käyttöprosentti.

./check_snmp_load.pl -H xx.x.x.xx -C public -w 3,3,2 -c 4,4,3 -T cisco

Tämä plugin on tarkoitettu käytettäväksi monissa eri systeemeissä. T-optiolla määritetään, minkä valmistajan laitteessa sitä käytetään. Jos laite olisi esimerkiksi HP:n, T- optio olisi HP. Kuten aiemmin kävi ilmi, tarkastuksia verkkolaitteille voidaan tehdä OID-tunnusten avulla. Yleisenä pluginina toimii check_snmp. Seuraavassa on esimerkki, jossa tarkastetaan vapaana oleva muistin määrä HP:n kytkimessä.

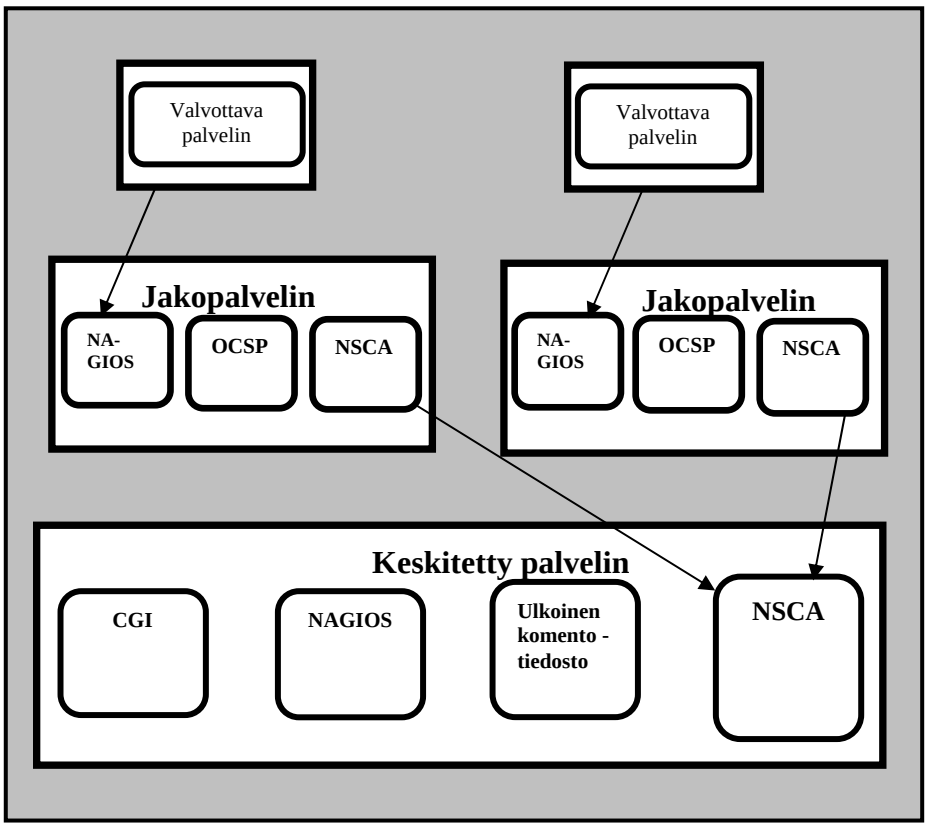
./check_snmp -H xx.x.x.xx -C public -o .1.3.6.1.4.1.11.2.14.11.5.1.1.2.1.1.1.6.1

5 VALVONNAN KEHITTÄMISRATKAISUJA

Tässä luvussa käydään läpi Nagioksen lisäominaisuuksia, jotka osaltaan tukevat valvottujen kohteiden suurta määrää.

5.1 Valvonnan jakaminen

Valvottujen kohteiden valtava määrä voi aiheuttaa Nagios-palvelimelle kuormaa ja sitä myöten palveluiden tarkastus voi tulla epävakaaksi tai jopa katketa kokonaan. Yksi ratkaisu ongelmaan on jaettu valvonta, joka toteutetaan yhden keskitetyn ja kahden jaetun Nagios-palvelimen kautta. Käytännössä yksi keskitetty Nagios -palvelin kerää yhteen kahden tai jopa mahdollisesti useammankin jaetun palvelimen tarkastustulokset (kuva 16).



KUVA 16. Valvonnan jakoprosessi

Kun jaettu Nagios -palvelin on tehnyt tarkastuksen, se suorittaa niin sanotun OCSP – käskyn, jonka seurauksena tarkistukset menevät NSCA -prosessiin, joka puolestaan siirtää tarkastukset keskitetylle Nagios -palvelimelle. Jaetut palvelimet ainoastaan tekevät tarkistuksen sekä lähettävät ne eteenpäin keskitetylle palvelimelle. Ne eivät lähetä esimerkiksi ilmoituksia käyttäjälle eikä niihin ole asennettu web-

käyttöliittymään, jotta tarkistukset näkyisivät graafisessa muodossa. Keskitetylle palvelimelle tulee lisätä myös jaettujen palvelinten tarkastuskonfiguraatiot, jotta passiivinen tarkastusprosessi on mahdollinen. (Nagios: Distributed monitoring.)

5.2 Redundantti valvonta ja valvonnan siirto vikatilanteessa

Valvonnan vikasietoisuus lisääntyy huomattavasti, jos käytetään niin sanottua redundanttia valvontaa. Tämä tarkoittaa sitä, että lisätään redundantti (ylimääräinen) Nagios-palvelin valvontajärjestelmään. Jos Nagios -palvelin hajoaa tai sen verkkoyhteys katkeaa, valvontaa voidaan jatkaa toisella palvelimella. Näin käytössä on kaksi palvelinta, jotka ovat niin sanotusti ”master” ja ”slave”. Molemmat ovat tarkastavia palvelimia, mutta kun master -palvelin on toiminnassa, orjapalvelimen ilmoitusten lähettäminen on poistettu ja master -palvelimen ulkoisten komentojen vastaanotto on sallittu. Toisin sanoen NSCA:n mukaisesti. Tämän lisäksi slave- palvelin tarkkailee Master -palvelinta sekä sen Nagios prosessin tilaa. Master- palvelimen ei tarvitse olla tietoinen slave- palvelimen olemassaolosta. Näille nimenomaisille tarkastuksille on määritetty tapahtumankäsittelijät (eventhandlers), jotka kertovat slave –palvelimelle ottaako se master -palvelimen paikan, jos se huomaa esimerkiksi master-palvelimen nagios -prosessin kadonneen. Kun master-palvelin on taas toimintakunnossa, slave-palvelin huomaa, että sen nagios-prosessi on päällä. Tapahtumankäsittelijään on määritetty, että jos master-palvelimen nagios-prosessi on taas toiminnassa, slave-palvelin lakkaa lähettämästä ilmoituksia.

Redundantin valvonnan huonoina puolina ovat verkkoliikenteen kasvu, koska molemmat palvelimet tekevät tarkistuksia samoille koneille. Jos slave-palvelin ottaa master-palvelimen paikan, vie se kauan aikaa. Tarkastuksia ei tänä aikana pystytä tekemään. Aikaa pystytään kuitenkin lyhentämään pienentämällä tarkastusten väliä slave- palvelimelta master- palvelimeen.

Valvonnan siirto nimensä mukaisesti siirtää vikatilanteen tullessa valvontavastuun toisaalle. Tämä tarkoittaa sitä, että slave -palvelin ei tee tarkastuksia, jos master-palvelin on toiminnassa. Slave -palvelimelta poistetaan käytöstä tarkastusten suorittaminen sekä ilmoitusten lähettäminen. Se on niin sanotussa odotustilassa siihen asti, kunnes master-palvelin joutuu vikatilaan. Slave -palvelimelle asetetaan cron -palvelu päälle, jossa se pyörittää skriptiä, joka tutkii master -palvelimen Nagios -prosessin tilaa NRPE -pluginin avulla. Jos se havaitsee että Master -palvelimen Nagios -prosessi ei ole päällä, se aktivoi slave -palvelimen niin että se ottaa käyttöön tarkastusten suorittamisen sekä ilmoitusten lähettämisen. Jotta slave -palvelin olisi ajan tasalla alkaessaan tekemään tarkistuksia, master -palvelimelta voi esimerkiksi sallia tarkastus lähettämisen aina slave- serverille käyttäen OCSP-käskyä.

Tämä siirtää tarkistukset NSCA -lisän avulla slave-palvelimelle. (Nagios: Redundant and Failover Network Monitoring.)

5.3 Palvelutarkastusten ryhmittely

Selkeys on avainsana palveluiden ja palvelinten tarkastuksissa. Jos palvelimia ja palveluita ei ole jollakin tavalla järjestetty sekä niiden hallinta että valvonta on vaivalloista. Palvelimet voidaan jaotella esimerkiksi niiden käyttöjärjestelmien mukaan, jolloin Linux, Windows ja Solaris palvelimet ovat kaikki erillisinä palvelinryhminä (host-groups). Jokaiselle ryhmälle määritellään oma niin sanottu ”perusvalvontasetti”, joissa olisi perustarkistuksia kuten check_ping, check_disk, check_prosess ym. Tämän lisäksi lisätään käyttöjärjestelmään mukaan niitä tarkastuksia, mitä kyseinen käyttöjärjestelmä vaatii. Samaa jaotteluperustetta voidaan pitää verkon aktiivilaitteita tarkastettaessa. Jaotellaan kytkimet ja reitittimet omiksi ryhmikseen sekä tarvittaessa niiden merkkien mukaan esimerkiksi HP -kytkimet, Cisco – reitittimet. Toinen jaotteluratkaisu on palvelimen prioriteetin mukaan, eli jaotellaan palvelimet niiden tärkeysasteeseen perustuen. Prioriteetti 1 -ryhmään kuuluvat ne palvelimet, jotka ovat liiketoiminnan kannalta kaikista tärkeimmät, esimerkiksi web-sovelluspalvelimet. Prioriteetti 2 -ryhmää kuuluvat tiedostopalvelimet, DHCP -palvelimet ym. Prioriteetti 3 -ryhmään kuuluvat vähiten merkitykselliset palvelimet yrityksen kannalta. Näitä voisi olla esimerkiksi testipalvelimet. Palveluita voidaan myös ryhmitellä palvelimien tapaan. Palveluryhmät tulivat mahdolliseksi vasta kun versio kaksi ilmestyi Nagiokselta.

5.4 Tarkastusten käyttöönotto

Kun palvelut ja palvelimet on ryhmitelty sekä tarkastuksille välttämättömät peruskonfiguraatiot kuten kontaktit, kontaktiryhmät, aikaperiodit, palveluryhmät, palvelinryhmät määritelty Nagios-serverille, on palveluiden ja palvelinten lisääminen tarpeen vaatiessa vaivaton toimenpide. Tällöin tarvitsee lisätä vain tarkastettava kohde (palvelin tai palvelu) sekä tarkastuskomento -tiedostot Nagioksen konfiguraatio -tiedostoihin. Ei pakollisia, mutta kannattavia lisäyksiä ovat kohteiden lisääminen ryhmiin, jotta valvonta selkeytyisi. Uuden tarkastettavan palvelimen lisäyksessä tulee myös näiden edellä mainittujen konfiguraatioiden lisäksi asentaa tarkastettavalle palvelimelle palvelu (SNMP), jotta sitä pystytään tarkastamaan.

6 POHDINTAA

Opinnäytetyön tavoite oli selvittää, mitä tarkastusmenetelmiä Nagioksessa voidaan käyttää, lähinnä keskittyen SNMP -protokollaa hyväksi käyttäen. Eri tarkastuskohteiden ja niiden nopea valmiiksi asettaminen oli myös tavoitteena tässä työssä. Itse päätyökalu Nagios on hämmästyttävän vakaa ja monipuolinen sovellus, joka alkuhämmennyksen jälkeen osoittautui hyvin selkeäksi käyttää. Sen sijaan konfigurointi tuotti varsinkin alussa melkoista päänvaivaa mielestäni melko sekavan dokumentaation takia. Pluginien saatavuus riippui siitä mitä ja millä työkalulla tarkastuksia tehtiin. Esimerkiksi SNMP –pluginien saatavuus oli mielestäni hyvin heikkoa, varsikin silloin, kun pluginin piti tukea versio kolmea. Kuitenkin kun puhutaan avoimen lähdekoodin tuotteista, joita pluginitkin ovat ja joiden kirjoittaminen ja muuttaminen on vapaata, löytyy aina jokin sivusto, johon joku ystävällinen sielu on kirjoittanut tai muuttanut juuri sopivan tarkastuksen.

Opinnäytetyön ehkä haastavin osa oli oikean konfiguraatio -tiedon löytyminen. Vaikka Nagioksella ja SNMP:llä oli omat dokumentaatiosivut, niistä sai harvinaisen vähän irti. Näissä tilanteissa oli paras keino turvautua kyseisiä tuotteita käsittelevien keskustelupalstojen lukemiseen, joista sai loistavia vinkkejä kokeneilta käyttäjiltä.

Vaikka käsittelin Nagiosta mielestäni melko kattavasti, oli aihetta rajattava kuitenkin, niin ettei se paisuisi loputtomiin. Esimerkiksi SNMP:n trap- ominaisuus oli pakko jättää työstä pois ajanpuutteen vuoksi. Tavoite kuitenkin toteutui, koska sain eri tarkastusmenetelmillä olevat tarkastukset toimimaan. Olen todella tyytyväinen saadessani tehdä opinnäytetyöni tästä aiheesta, koska tämä antoi itselleni täysin uuden näkökulman valvontasovelluksiin sekä niiden mahdollisuuksiin.

7 KÄSITTEET

SNMP	(*Simple Network Management Protocol*) on TCP/IP-verkkojen hallinnassa käytettävä tietoliikenneprotokolla.
TCP/IP	*(Transmission Control Protocol / Internet Protocol)* on usean tietoverkkoprotokollan yhdistelmä, jota käytetään Internet-liikennöinnissä.
NMS	*(Network Management Station)* on manageriohjelmisto verkkoa hallitsevassa asemassa.
MNE	*(Managed Network Entity)* on agenttiohjelmisto, joka sijaitsee hallittavassa laitteessa.
MIB	*(Management Information base)* on SNMP:n määrittämä hallintatietokanta joukolle objekteja.
SMI	*(Structure of Management Information)* sisältää tiedot kuinka määritellään ja rakennetaan MIB:t.
RMON	*(Remote Network Monitoring)* on ohjelmisto tai erillinen laite, joka on sijoitettu verkon laitteeseen keräämään tietoa verkkoliikenteestä ja tallentaa keräämänsä tiedot tiedostoon (MIB).
WBEM	*(Web-based Enterprise Managemet)* on hallittavan laitteen hallintaa selaimen avulla.
GPL	*(General Public License)* tarjoittaa sitä, että lisenssin alainen ohjelmisto on kenen tahansa levitettävissä sekä muokattavissa.
CGI	(*Common Gateway Interface)* tarkoittaa rajapintaa, jonka kautta ohjelmat voivat tuottaa www-sivuja yhteistyössä palvelimen kanssa.
Plugin	Nagiosen tekemissä tarkistuksissa käytettävä lisäosa.
NRPE	*(Nagios Remote Plugin Executor)* on Nagiokseen kuuluva aktiivinen lisäosa, joka suorittaa tarkastukset valvottavilla koneilla.
xinetd	Yhteinen kuuntelijaprosessi, joka herättää pyydetyn palvelun vain tarvittaessa.

NSCA	*(Nagios Service Check Acceptor)* on Nagiokseen kuuluva lisäosa, jonka tehtävä on ottaa vastaan tarkastusten tulokset.
OCSP	*(Obsessive Compulsive Service Processor)* on käsky joka siirtää valvottavien palvelimien tarkastukset NSCA -prosessiin, joka puolestaan siirtää tarkastukset keskitetylle Nagios -palvelimelle.
OID	(Object Identifier) on numeerinen arvo MIB:stä, eli tarkastettavan kohteen yksilöivä tunnus.

LÄHTEET

Nagios: About nagios [online] [viitattu 29.6.2006]
http://www.nagios.org/about/

Nagios: Distributed Monitoring [online] [viitattu 22.8.2006]
http://nagios.sourceforge.net/docs/2_0/distributed.html

Nagios: Redundant and Failover Network Monitoring [online] [viitattu 22.8.2006]
http://nagios.sourceforge.net/docs/2_0/redundancy.html

Nagios: State Types [online] [viitattu 22.8.2006]
http://nagios.sourceforge.net/docs/2_0/statetypes.html

Ogletree, Terry 2001. Inside Verkot
Helsinki: It-press

Hautaniemi, Mika 1994. Diplomityö: TKK/Atk-keskuksen TCP/IP-verkon valvonta ja hallinta [online] [viitattu 25.6.2006]
http://www.netlab.tkk.fi/julkaisut/tyot/diplomityot/611/verkonhallinta.html

Haikonen, Jarno, Hlinovsk, Jan & Paju, Antti 2000. Harjoitustyö: Tietoverkkolaboratorio – TKK Teletekniikan perusteet [online] [viitattu 10.7.2006]
http://www.netlab.tkk.fi/opetus/s38118/s00/tyot/47/snmp.shtml

Jaakohuhta, Hannu 2005. Lähiverkot
Helsinki: It-press

Puska, Matti 2000. Lähiverkkojen tekniikka. 2 uudistettu painos.
Helsinki: satku.fi

Comer, Douglass 2002. TCP/IP
Helsinki: It-press

Casad, Joe & Wilsey, Bob 1999. TCP/IP-trainer. Suom. Juha Salmela.
Helsinki: It-press

Wtcs.org [online] [viitattu 10.10.2006]
http://www.wtcs.org/snmp4tpc/snmp4nw.htm

Leinwald, Allan & Fang, Conroy, Karen 1996. Network Management A Practical Perspective. Addison Wesley Longman Inc.

Turunen, Jukka & Leppälahti, Jarkko 2000. Harjoitustyö: Tietoverkkolaboratorio – TKK Teletekniikan perusteet [online] [viitattu 27.10.2006]
http://www.netlab.tkk.fi/opetus/s38118/s00/tyot/35/cmip.shtml

Stallings, W 1993. SNMP, SNMPv2 and CMIP: The Practical Guide to Network-Management Standards. Addison-Wesley Publishing Company Inc.

Stallings, W 1999. SNMP, SNMPv2 and RMON 1 and 2: Addison-Wesley Publishing Company Inc.

Pöllönen, Mikko 2006. Linux työ: SNMP (Simple Network Management Protocol) – Lähiverkot –erikoistyökurssi [online] [viitattu 27.10.2006]
http://www.it.lut.fi/kurssit/05-06/Ti5316800/Linux-tyot/SNMP-Mikko_Pollonen-dokumentti.pdf